



**DISEÑO MECÁNICO ÁGIL;
UN NUEVO MÉTODO ENFOCADO A PROYECTOS DE
DESARROLLO TECNOLÓGICO**

TESIS

PARA OBTENER EL GRADO DE

DOCTOR EN MANUFACTURA AVANZADA

PRESENTA

**M. C. JORGE ARMANDO GUTIÉRREZ BRAVO
DR. ISAAC HERNÁNDEZ ARRIAGA**

SANTIAGO DE QUERÉTARO, QRO, NOVIEMBRE 2019.

CARTA DE LIBERACIÓN DEL ASESOR



26 de junio del 2019

Mtro. Geovany González Carlos
Coordinador Académico
CIATEQ, A.C.

El abajo firmante, miembro del Comité Tutorial del alumno Maestro Jorge Armando Gutiérrez Bravo, una vez revisada la Tesis titulada: "**DISEÑO MECÁNICO ÁGIL; UN NUEVO MÉTODO ENFOCADO A PROYECTOS DE DESARROLLO TECNOLÓGICO**", autorizo que el citado trabajo sea presentado por el alumno para la revisión del mismo con el fin de alcanzar el grado de Doctor en Manufactura Avanzada durante el Examen de Titulación correspondiente.

Y para que así conste se firma la presente a los 26 días del mes de junio del año 2019.

Dr. Isaaq Hernández Arriaga

Grado y nombre completo

Asesor Académico

CARTA DE LIBERACIÓN DEL REVISOR



**GOBIERNO DE
MÉXICO**



CONACYT
Consejo Nacional de Ciencia y Tecnología



Querétaro, Qro., a 29 de octubre del 2019.

DRA. MARÍA GUADALUPE NAVARRO ROJERO
Directora de Posgrado
CIATEQ, A.C.

Por medio de la presente me dirijo a usted en calidad de Revisor del proyecto terminal del alumno JORGE ARMANDO GUTIERREZ BRAVO, cuyo título es:

**DISEÑO MECÁNICO ÁGIL;
UN NUEVO MÉTODO ENFOCADO A PROYECTOS DE DESARROLLO TECNOLÓGICO.**

Después de haberlo leído, corregido e intercambiado información con el alumno, y realizado los cambios que le fueron sugeridos, puede ser autorizada su impresión, a fin de que se inicien los trámites correspondientes para su defensa.

Sin otro particular por el momento, y en espera de que mis sugerencias sean tomadas en cuenta en beneficio del estudiante y la Institución, agradezco la atención prestada.

Atentamente,

Dr. Agustín Escamilla Martínez
Director de Sistemas Mecánicos
CIATEQ, A.C.

"2019, Año del Caudillo del Sur, Emiliano Zapata"

El Marqués, Qro.

+52 (442) 196 1500

| 01 800 800 3798

| mkf@ciateq.mx

www.ciateq.mx

CARTA DE LIBERACIÓN DEL REVISOR



San Luis Potosí, SLP, 8 de noviembre del 2019.

Dra. María Guadalupe Navarro Rojero
Directora de Posgrado
CIATEQ, A.C.

Por medio de la presente me dirijo a usted en calidad de Revisor del proyecto terminal del alumno **Jorge Armando Gutiérrez Bravo**, cuyo título es:

"Diseño mecánico ágil; un nuevo método enfocado a proyectos de desarrollo tecnológico"

Después de haberlo leído, corregido e intercambiado información con el alumno, y realizado los cambios que le fueron sugeridos, puede ser autorizada su impresión, a fin de que se inicien los trámites correspondientes para su defensa.

Sin otro particular por el momento, y en espera de que mis sugerencias sean tomadas en cuenta en beneficio del estudiante y la Institución, agradezco la atención prestada.

Atentamente,

A handwritten signature in black ink, appearing to read "José", written over a horizontal line. The signature is enclosed within a large, hand-drawn, irregular scribble that extends upwards and to the right.

Dr. José Guadalupe Octavio Cabrera Lazarini
Profesor Investigador

AGRADECIMIENTOS

Agradezco a dios, a mi esposa Amalia y a mis hijos; Diego, Julia, Bruno y Zoe, por ser mi motor durante este proyecto. También agradezco a Paul Jaques y Guillermo Gómora, por las facilidades y el apoyo de exponer esta propuesta en la empresa Industrial Automation México. También al M. C. Juan Noé Reyes Elías del CIDESI, por su apoyo durante el desarrollo de la simulación en realidad aumentada con el uso de los Hololens de Microsoft. Finalmente, agradezco a mis asesores; Isaac Hernández y Octavio Cabrera por sus recomendaciones y revisiones durante este proyecto.

Gracias a CONACYT por el apoyo durante estos tres años.

RESUMEN

En la actualidad, los productos están sometidos a un proceso de evolución continuo, motivando a las empresas a buscar nuevos métodos que atiendan la dinámica de dichos cambios. Metodologías ágiles han presentado alternativas para atender esta necesidad. Sin embargo, en el campo del diseño mecánico, aún no se ha resuelto la contradicción para obtener productos ágiles y técnicamente fundamentados. Con base en lo expuesto, esta tesis propone un nuevo método para atender dicha contradicción, basado en el acoplamiento de los principios ágiles, con métodos, técnicas y herramientas del diseño mecánico. Logrando a través de las herramientas virtuales, tanto del modelado 3D como la realidad aumentada, obtener incrementos en la configuración del producto en cada iteración, sin la necesidad de interactuar con un producto físico. El principio se basa en encontrar una solución temprana e intentar mejorarla, en lugar de abordar el problema y posponer la solución. La base metodológica de esta tesis es una investigación aplicada exploratoria, con el objetivo de conocer, definir y esquematizar detalladamente cada una de las etapas del nuevo método del diseño mecánico, aplicada en los proyectos de desarrollo tecnológico. Como fundamento teórico, se exponen los elementos principales de esta investigación; tanto de los métodos de diseño tradicionales, como de los métodos ágiles. Presentando los puntos coincidentes, así como sus contradicciones, basados en la revisión del estado del arte de los métodos ágiles aplicados más allá del "software". Finalmente se presentan y discuten los resultados relevantes de la aplicación de la metodología en un proyecto real, logrando una reducción del 30% en el tiempo de entrega del dispositivo, respecto a lo planeado, así como una satisfacción total del cliente en la encuesta realizada al finalizar el proyecto, generada principalmente por el respaldo técnico y el corto tiempo de desarrollo.

Palabras clave: Metodología de diseño, PDT, Diseño mecánico, Métodos ágiles.

ABSTRACT

Currently, products are subject to a process of continuous evolution, motivating companies to look for new methods that address the dynamics of these changes. Agile methodologies have presented alternatives to meet this need. However, in the field of mechanical design, the contradiction to obtain agile and technically grounded products has not yet been resolved. Based on the above, this thesis proposes a new method to address this contradiction, based on the coupling of agile principles, with methods, techniques and tools of mechanical design. Achieving through virtual tools, both 3D modeling and augmented reality, obtain increases in product configuration in each iteration, without the need to interact with a physical product. The principle is based on finding an early solution and trying to improve it, rather than addressing the problem and postponing the solution. The methodological basis of this thesis is an exploratory applied research, with the objective of knowing, defining and schematically detailing each of the stages of the new method of mechanical design, applied in technological development projects. As a theoretical basis, the main elements of this research are presented; both from traditional design methods, as from agile methods. Presenting the coincident points, as well as their contradictions, based on the review of the state of the art of agile methods applied beyond "software". Finally, the relevant results of the application of the methodology in a real project are presented and discussed, achieving a 30% reduction in the delivery time of the device, compared to the planned, as well as a total customer satisfaction in the survey conducted. Finalize the project, generated mainly by technical support and short development time.

Keywords: Design methodology, TDP, Mechanical design, Agile methods.

ÍNDICE

CARTA DE LIBERACIÓN DEL ASESOR.....	I
CARTA DE LIBERACIÓN DEL REVISOR.....	II
CARTA DE LIBERACIÓN DEL REVISOR.....	III
AGRADECIMIENTOS.....	IV
RESUMEN	V
ABSTRACT.....	VI
ÍNDICE DE FIGURAS	X
ÍNDICE DE TABLAS.....	XII
GLOSARIO	XIII
CAPÍTULO 1. INTRODUCCIÓN	1
1.1 ANTECEDENTES	1
1.2 DEFINICIÓN DEL PROBLEMA.....	3
1.3 JUSTIFICACIÓN.....	4
1.4 DISEÑO METODOLÓGICO DE LA INVESTIGACIÓN.....	8
1.4 OBJETIVOS	8
1.4.1 Objetivo general:.....	8
1.4.2 Objetivos específicos:.....	8
1.5 HIPÓTESIS	9
CAPÍTULO 2. MARCO TEÓRICO.....	10
2.1 TEORÍA DE LOS ELEMENTOS PRINCIPALES DE LA INVESTIGACIÓN.....	10
2.1.1 Producto.....	10
2.1.2 Proyectos de Desarrollo Tecnológico (PDT).....	13
2.1.3 Ciclo de Vida de los PDT.....	14
2.1.4 Teoría del diseño	18
2.1.5 Métodos	22
2.2 MÉTODOS EN EL DISEÑO.....	26
2.2.1 Diseño metodológico.....	26
2.2.3 Diseño en Ingeniería.....	29
2.2.4 Metodologías tradicionales del diseño	32
2.3 MÉTODOS ÁGILES	38
2.3.1 Manifiesto Ágil	38
2.3.2 Ecosistemas de desarrollo de "software" ágil	39
2.3.3 Características de los enfoques ágiles	40

2.3.5	Diseño de una metodología ágil.....	57
2.3.6	Kanban.....	65
2.3.7	Scrumban.....	68
2.4	CONTRASTE ENTRE MÉTODOS ÁGILES Y MÉTODOS DE DISEÑO.....	71
2.4.1	Investigar las iteraciones del proceso de diseño de ingeniería.....	71
2.4.2	Mejora del proceso de diseño con enfoques basados en modelos.....	73
2.4.3	Coordinación en desarrollo de productos complejos.....	75
2.4.4	Modelado de diseño y modelado funcional en todas las disciplinas.....	76
2.4.5	Etapas en el ciclo de vida del producto en una metodología.....	77
2.5	MÉTODOS ÁGILES APLICADOS MÁS ALLÁ DEL "SOFTWARE".....	78
2.5.1	Idoneidad del proyecto para metodologías ágiles.....	79
2.5.2	Ingeniería de sistemas ágiles versus sistemas ágiles de Ingeniería.....	80
2.5.3	Una base empírica para la flexibilidad del producto.....	82
2.5.4	Aplicación de "Scrum" a un proyecto para prototipo de controles de vehículos.....	83
2.5.5	"Scrum" en desarrollo de productos mecánicos.....	83
2.5.6	Recursos informales en línea.....	85
CAPÍTULO 3. PROPUESTA METODOLÓGICA.....		93
3.1	NUEVO MÉTODO PARA DISEÑO MECÁNICO.....	95
3.2	REVISIÓN DE CONDICIONES DE ÉXITO.....	96
3.2	DEFINICIÓN DEL PRODUCTO.....	98
3.3	PLANEACIÓN DEL PRODUCTO.....	104
3.4	PLANEACIÓN DE LAS ITERACIONES.....	106
3.5	ITERACIONES.....	108
3.5.1	Diseño conceptual.....	109
3.5.2	Revisión del concepto.....	114
3.5.3	Diseño óptimo.....	114
3.6	VALIDACIÓN DEL CLIENTE.....	122
3.7	RETROSPECTIVA.....	122
3.8	ENTREGABLES.....	122
CAPITULO 4. RESULTADOS.....		123
4.1	CASO DE ESTUDIO.....	123
4.1.1	Revisión de condiciones de éxito.....	123
4.1.2	Expectativas del cliente.....	125
4.1.3	Planeación general.....	128
4.1.3	Planeación de las iteraciones.....	129

4.1.4 Iteraciones	130
4.1.5 Revisión con el cliente.....	134
4.1.6 Retrospectiva.....	134
4.1.7 Entregables	134
CONCLUSIONES.....	135
APORTACION DE LA TESIS	136
RECOMENDACIONES.....	137
BIBLIOGRAFÍA	138
ANEXO	139

ÍNDICE DE FIGURAS

Figura 1. Inversión Total PEI a nivel nacional, periodo 2009 al 2017.	4
Figura 2. PEI, número de proyectos beneficiados y número de apoyo recibido por tamaño de empresa. 2009 al 2017.	5
Figura 3. Número de proyectos beneficiados y monto de apoyo recibido, 2009-2017.	5
Figura 4. Son 33 las áreas industriales a las que pertenecen los proyectos apoyados a nivel nacional, durante las convocatorias 2009-2017.	6
Figura 5. IES/CPI con mayor participación en el PEI (2009-2017).	6
Figura 6. Metodología de la tesis.	8
Figura 7. Estructura de la revisión del estado del arte.	10
Figura 8. Clasificación de tipos de productos.	13
Figura 9. Etapas del ciclo de vida típico de los proyectos de desarrollo tecnológico.	15
Figura 10. Niveles de madurez de los proyectos.	16
Figura 11. Tipos de diseño.	18
Figura 12. Características deseadas en las metodologías del diseño.	22
Figura 13. Modelo de diseño, Diamante-Doble.	24
Figura 14. Proceso de diseño de ingeniería según Ertas y Jones [48].	33
Figura 15. Modelo del proceso básico de diseño.	34
Figura 16. Modelo ajustado con retroalimentación básica.	35
Figura 17. Comparación del método de diseño con el método científico.	35
Figura 18. Pasos del proceso de diseño identificados por Haik y Shahin [50].	36
Figura 19. Pasos en el proceso de planificación y diseño.	37
Figura 20. Método "Scrum" de gestión de proyectos.	44
Figura 21. Método de desarrollo de sistemas dinámicos.	45
Figura 22. Metodología ágil cristal.	47
Figura 23. Fases del ciclo de vida de desarrollo de "software" adaptativo.	54
Figura 24. Ejemplo de placa Kanban.	67
Figura 25. Enfoques de diseño incremental (izquierda) y progresiva (derecha).	73
Figura 26. Capas de corte de Brand en edificios.	90
Figura 27. Adaptación de capas de corte a sistemas de "software"	90
Figura 28. Marco de Cynefin.	94
Figura 29. "SCRUM"	95
Figura 30. Nuevo método de diseño mecánico ágil.	95
Figura 31. Balance entre funciones y requerimientos.	98
Figura 32. Diagrama básico de funciones.	101
Figura 33. Esquema del modelo de KANO.	104
Figura 34. Equipo de diseño.	105
Figura 35. Desglose de tareas en cada función.	106
Figura 36. Cartera de Iteraciones.	107
Figura 37. Iteraciones.	108
Figura 38. Objetivos de los incrementos en cada iteración.	109
Figura 39. Proceso del diseño mecánico conceptual.	110
Figura 40. Niveles de búsqueda de soluciones.	110
Figura 41. Divergencia y convergencia de ideas.	112
Figura 42. Estructura del diseño mecánico ágil.	115
Figura 43. Estructura de la mecánica clásica y ondulatoria.	116
Figura 44. Diseño para la excelencia.	117
Figura 45. Tipos de AMEF.	118

Figura 46. Proceso estándar para desarrollo del AMEF de diseño.....	119
Figura 47. "software"s más importantes para modelado 3D.....	120
Figura 48. Demostración de una aplicación de los Hololens de Microsoft.....	121
Figura 49. Realidad aumentada en simulación de fluidos.....	121
Figura 50. Representación de un controlador de motores.....	123
Figura 51. Plan general, caso de estudio.....	129
Figura 52. Diseño final de la primera iteración en un ambiente virtual.....	133
Figura 53. Diseño final de la primera iteración en un ambiente virtual.....	2
Figura 54. Diseño final de la primera iteración usando Realidad aumentada.....	3
Figura 55. Diseño final de la primera iteración usando Realidad aumentada.....	3
Figura 56. Diseño final de la primera iteración usando Realidad aumentada.....	4
Figura 57. Diseño final, plano de ensamble.....	4

ÍNDICE DE TABLAS

Tabla 1. Etapas de maduración tecnológica.....	12
Tabla 2. Características de una metodología.....	23
Tabla 3. Clasificación de modelos de diseño.....	25
Tabla 4. Lista de verificación de las condiciones de éxito.....	97
Tabla 5. Matriz de funciones.....	102
Tabla 6. Matriz de requisitos.....	103
Tabla 7. Elementos básicos de una planeación.....	106
Tabla 8. Plan de iteraciones.....	107
Tabla 9. Matriz de análisis y selección del diseño conceptual.....	113
Tabla 10. Análisis de condiciones de éxito, caso de estudio.....	124
Tabla 11. Matriz de funciones, caso de estudio.....	125
Tabla 12. Matriz de funciones completa, caso de estudio.....	126
Tabla 13. Matriz de requerimientos, caso de estudio.....	127
Tabla 14. Plan general, caso de estudio.....	128
Tabla 15. Lista de funciones, caso de estudio.....	129
Tabla 16. Requerimientos de la función de inspeccionar dimensiones.....	130
Tabla 17. Análisis y selección del concepto.....	131
Tabla 18. AMEF de diseño.....	133
Tabla 19. Severidad AMEF de diseño.....	1
Tabla 20. Ocurrencia AMEF de diseño.....	1
Tabla 21. Detección AMEF de diseño.....	1

GLOSARIO

AMEF - Análisis del Modo y Efecto de Fallas
ASD - Adaptive Software Development
CAD - Computer Aided Design
CAEM - Conocidos como modos de cambio y efectos análisis
CAM - Cambridge Advanced Modeler
CE - Conformité Européenne
CPI - Centros públicos de investigación
CU - Casos de uso
DSDM - Método de Desarrollo Dinámico de Sistemas
FDD - Desarrollo impulsado por características
IES - Instituciones de educación superior
IFM - Modelado Funcional Integrado
IMPI - Instituto Mexicano de propiedad intelectual
JAD - Diseño de Aplicación Conjunta
LRC - Linear responsibility chart
LT - Lead time
NASA - National Aeronautics and Space Administration
OMPI - Organización mundial de la propiedad intelectual
PCM - Percentage of completion method
PDT - Proyectos de desarrollo Tecnológico
PEI - Programa de estímulo a la innovación
QFD - Quality function deployment
TRIZ - Teoriya Riesheniya Izobrietatielskij Zadach
WBS - Work breakdown structure
XP - Xtreme Programming

CAPÍTULO 1. INTRODUCCIÓN

Investigaciones recientes han demostrado que es factible el uso de metodologías ágiles en el campo del diseño mecánico, como lo presentan en sus investigaciones Reynisdottir (1), Also (2) y Dwyer (3), los cuales concluyen que un método ágil ayuda a una mejor comprensión de un diseño complejo. Esto es para tratar con las posibles interdependencias entre las características, dónde desarrollar una característica por completo puede tener impactos negativos en otras, o la integración de nuevas características puede necesitar un cambio en una característica existente. Buscando que las características y subsistemas sean diseñados para ser tan modulares y desacoplados como prácticos, lo que reduce los impactos de los cambios en una característica en otra. Sin embargo, dichas investigaciones señalan también la existencia de una brecha técnica significativa para lograr la implementación de métodos ágiles en proyectos reales de diseño mecánico. Con base en lo anterior, este trabajo de tesis presenta una nueva alternativa metodológica aplicada a proyectos de desarrollo tecnológico (PDT), la cual acopla los fundamentos técnicos del diseño con metodologías ágiles, dichos PDT son definidos en la norma "MNX-GT-002-IMNC-2008".

1.1 ANTECEDENTES

El éxito de las empresas se basa principalmente en la creación de nuevos productos, debido a que un nuevo producto eleva directamente el ingreso de la empresa en el mediano plazo y genera un mayor posicionamiento en el mercado. Sin embargo, la rápida evolución en los productos presenta nuevos retos en las empresas para mantenerse en el mercado global. En el mundo, las inversiones en los proyectos suman billones de dólares anuales según datos mostrados en CHAOS (4). Los informes de CHAOS (4) han identificado el estado actual de las tasas de éxito de los proyectos en todas las organizaciones, observando que a pesar de mayor seguimiento e importancia colocada en cada proyecto, el rendimiento en las tasas de fracaso se ha mantenido altas y relativamente estables a lo largo de más de una década. Además, ejemplos específicos de fallas típicas en los proyectos que desarrollan nuevos productos han arrojado a la luz el impacto en las organizaciones. Se puede citar el ejemplo del multimillonario proyecto Iridium de MOTOROLA. Dicho proyectos se podría considerar como un éxito desde el punto de vista técnico, tomando en cuenta que se desarrolló en tiempo, con el presupuesto disponible.

Sin embargo, fue una catastrófica falla comercial debido a que no se consideraron los cambios en el ambiente del negocio. El equipo técnico y de gestión de Motorola fallaron porque nunca observaron los cambios durante el desarrollo del producto, la expansión de redes de los celulares socavarían el negocio de telefonía por satélite del modelo Iridium. En el mundo hay muchas empresas, instituciones y centros de investigación que desarrollan nuevos productos. Por ejemplo, en México existe un programa de estímulo a la innovación (PEI) que apoya a las empresas que desarrollan nuevos productos. Desde la creación de estos programas PEI en el 2009 hasta el 2017 se tienen registro de que en un promedio del 51% de los proyectos vinculados, se declara que se crearía un nuevo producto como resultado del proyecto. Sin embargo, los resultados no han sido tan exitosos como lo esperaban. En el informe de las cartas de cierre para proyectos de la convocatoria 2016 (5) se reportó que fueron 932 empresas beneficiadas por el PEI, de las cuales solo 447 reportaron que cumplieron con un proyecto exitoso y en el informe de las cartas de cierre para proyectos de la convocatoria 2017 (6) las empresas beneficiadas por el PEI fueron 383 de las cuales solo 69 declararon exitoso el proyecto. Considerando que en promedio un 51 % de estos proyectos crearon un nuevo producto, en 2016, solo 228 de 475 proyectos obtuvieron un nuevo producto innovador y exitoso. Y para el año 2017 es más grave el tema ya que de 195 proyectos solo 35 proyectos obtuvieron nuevos productos innovadores y exitosos. Mostrando una tendencia negativa en el desarrollo de este tipo de proyectos.

Existen varios factores que impactan a esta tendencia negativa en el desarrollo de nuevos productos. Sin embargo, empresas que desarrollan PDT usando metodologías tradicionales coinciden que el éxito o fracaso del proyecto, se define en las primeras fases del proyecto. Debido a que en muchas ocasiones los clientes no expresan claramente sus necesidades o las cambian constantemente, lo cual genera que en muchas ocasiones se entreguen productos que no cumplen con las demandas finales de los clientes, o los productos ya no tengan un impacto de innovación por el demasiado tiempo de desarrollo generado por los re-trabajos significativos durante las etapas de fabricación o entrega, dando un saldo negativo tanto para la empresa como para el cliente. En ocasiones, los proyectos son abandonados porque resulta mejor cancelarlos que seguir invirtiendo en productos que no representarían una ventaja competitiva para la sustentabilidad de la empresa. Entonces, empresas e instituciones afines a los PDT se han enfocado a encontrar alternativas que atiendan esta problemática desde el proceso del diseño del producto. Existen diferentes

tipos de diseño aplicados durante el desarrollo de nuevos productos. Sin embargo, cuando se tiene que crear un nuevo producto electo-mecánico, el primer diseño que se requiere es la parte mecánica del producto, debido a que en el proceso de diseño mecánico se presentará la morfología del producto y posteriormente otros tipos de diseño complementarían al producto, tales como: diseño eléctrico, diseño de control, diseño industrial, entre otros. Por lo tanto, se puede considerar al diseño mecánico como la primera fase en el desarrollo de nuevos productos que puede definir el éxito del proyecto. El diseño mecánico, en un sentido amplio y contemplado metodológicamente, supone llevar a cabo un conjunto de actividades interrelacionadas que incluyen, la definición de las necesidades del cliente, la ingeniería del producto, prototipos del proceso de fabricación, los aspectos de seguridad, ecodiseño, ergonómicos, de calidad, de protección del producto, de gestión de los recursos humanos, de gestión económica y de recursos, pruebas, ensayos, validación, entre otras. Por lo tanto, el proceso del diseño mecánico puede resultar complejo, en función del alcance y dimensión del proyecto a llevar a cabo, sin tener ninguna metodología de apoyo.

1.2 DEFINICIÓN DEL PROBLEMA

Empresas buscan nuevas alternativas para desarrollar nuevos productos para obtener proyectos exitosos a partir de entender y adaptarse a nuevos requerimientos del cliente durante la ejecución del proyecto. Un producto que no se adapta a los cambios del mercado se puede quedar obsoleto incluso antes de concluirse. El problema actual radica en que los PDT tiene como su principal característica la incertidumbre en los resultados que se obtendrán, presentando cambios durante todo el proceso de su desarrollo. Por lo tanto las metodologías tradicionales del diseño mecánico son insuficientes y requieren ser reestructuradas, para adaptarse a las nuevas demandas globales. Encontrando que los principios ágiles ofrecen una alternativa para adaptarse a los cambios en el producto, sin embargo, actualmente no se ha logrado una implementación contundente en proyectos de nuevos productos físicos, donde se obtengan productos ágiles y técnicamente fundamentados.

1.3 JUSTIFICACIÓN

En Mexico muchas empresas e instituciones (universidades y centros de investigación) crean productos de desarrollo tecnológico, tanto en sectores públicos como privados, manejando montos de millones de pesos invertidos en el desarrollo de nuevos productos. Solo como referencia de un ejemplo real y específico para PDT. En un reporte del CONACYT en el periodo 2009-2017 los proyectos apoyados por el PEI han detonado una inversión total en innovación por 52,360 millones de pesos en el país (47% de recursos públicos) un total de 5,987 proyectos como se muestra en la figura 1. Por cada peso público destinado a este Programa, los empresarios han invertido 1.13 pesos.

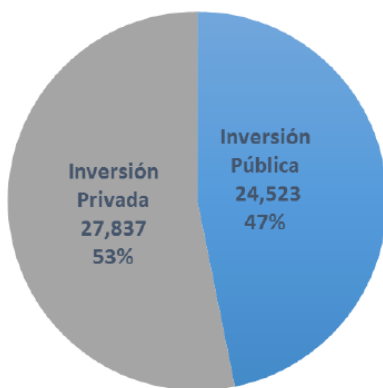


Figura 1. Inversión Total PEI a nivel nacional, periodo 2009 al 2017.
(Fuente: CONACYT, Dirección de Innovación 2017).

Las empresas pequeñas han recibido el mayor apoyo del Programa, en términos de proyectos apoyados como en recursos percibidos, el 37% y 39% respectivamente, seguidas de las empresas grandes, cuyo número de proyectos y monto apoyado representa los 30% del total mostrados en la figura 2.

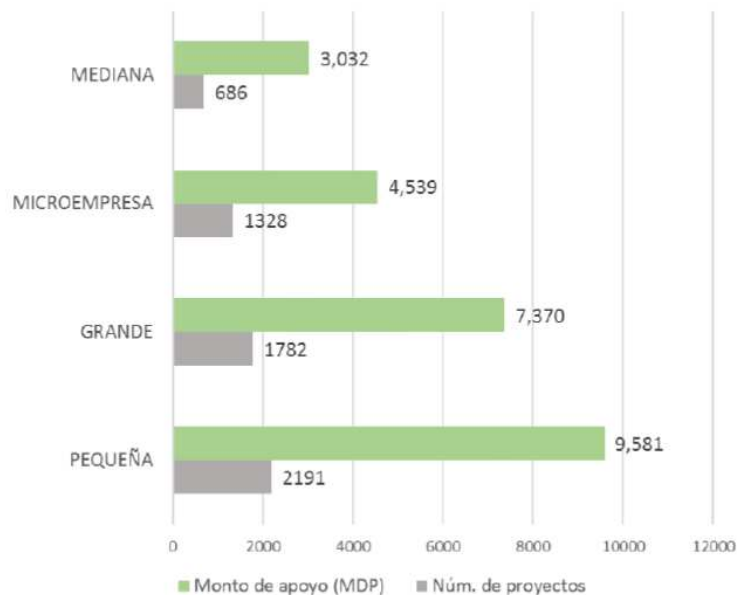


Figura 2. PEI, número de proyectos beneficiados y número de apoyo recibido por tamaño de empresa. 2009 al 2017.
(Fuente: CONACYT, Dirección de Innovación 2017).

El monto de apoyo total para todas las convocatorias del programa asciende a 24,523 millones de pesos, con los que se han beneficiado a 5,987 proyectos como se muestra en la figura 3.

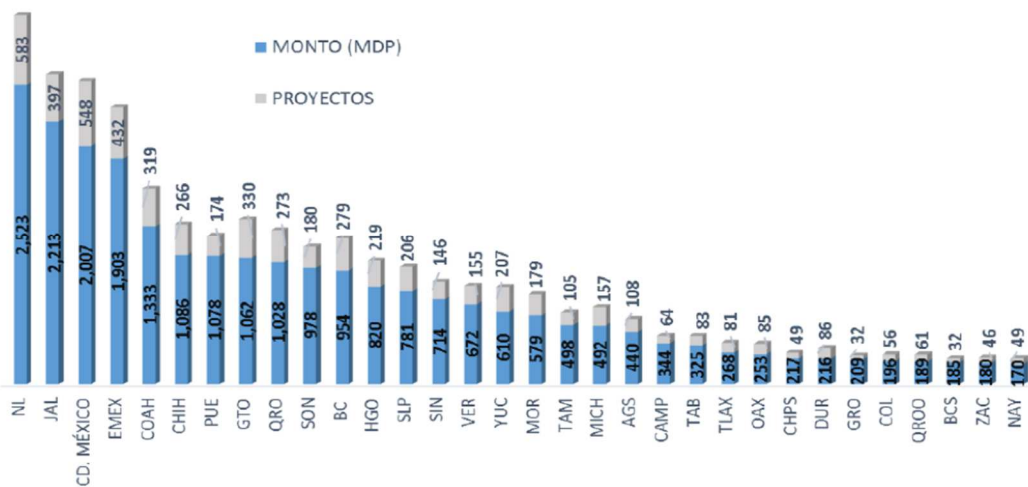


Figura 3. Número de proyectos beneficiados y monto de apoyo recibido, 2009-2017.
(Fuente: CONACYT, Dirección de Innovación 2017).

Sobresalen, tanto por el número de proyectos beneficiados, como por monto de apoyo: Tecnologías de la información, Automotriz, Alimentos, Agroindustrial, Química y Biotecnología, como se muestra en la figura 4 (en conjunto representan alrededor del 51% del total del número de proyectos apoyados y monto de apoyo del periodo).

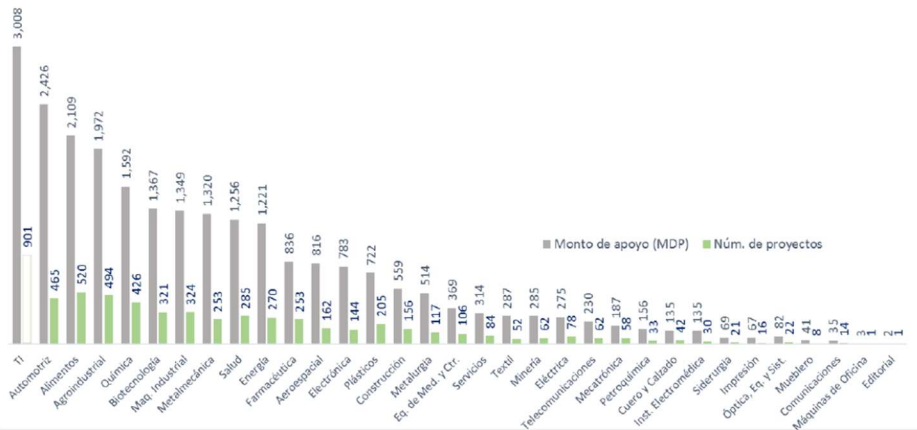


Figura 4. Son 33 las áreas industriales a las que pertenecen los proyectos apoyados a nivel nacional, durante las convocatorias 2009-2017. (Fuente: CONACYT, Dirección de Innovación 2017)

Empresas privadas, públicas, centros de investigación y universidades han atendido con al menos 50 proyectos en el periodo de 2009 al 2017, como se muestra en la figura 5. Por parte de las Instituciones de educación superior (IES), los investigadores se preocupan por cumplir con los compromisos establecidos en tiempo y forma para: mantener una buena relación con las empresas; Mantenerse actualizados en los problemas a resolver; la expectativa de mayores ingresos con futuros proyectos; y por mantener una buena reputación con el CONACYT, que es el principal financiador de los proyectos.

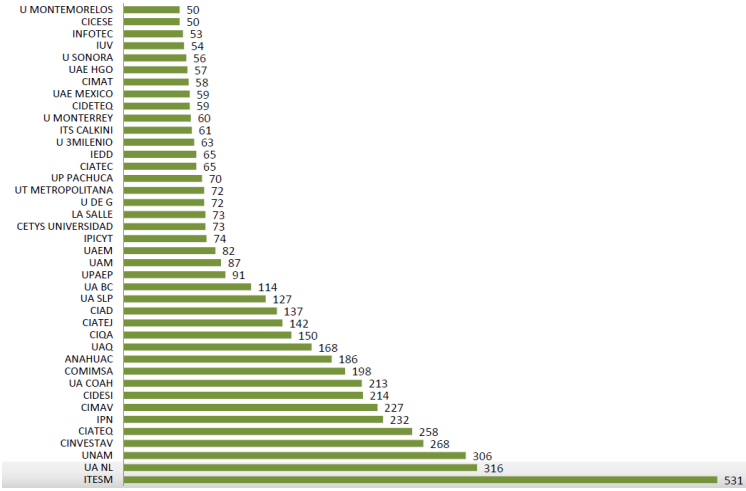


Figura 5. IES/CPI con mayor participación en el PEI (2009-2017). (Fuente: CONACYT, Dirección de Innovación 2017)

En los últimos años, el PEI ha aumentado el control y seguimiento de los proyectos, a través de una plataforma que administra la información de todos los involucrados, desde la aplicación del proyecto hasta su conclusión, con calendarios de compromisos. Donde ha detectado que muchas empresas no han repetido proyectos en más de una convocatoria,

ya que solo el 38% de las empresas han participado en más de una ocasión. Aquí se gesta la primera pregunta; ¿Por qué, el resto no ha participado de nuevo en el programa? Otra observación de campo indica que se desarrollan innovaciones sin patente o innovaciones con patentes vencidas para productos destinados a nuevos mercados, debido a que muchas de las empresas, IES y CPI no cuentan con metodologías sistemáticas para el desarrollo de estos productos, buscando innovaciones donde no existe o ya existen desarrollos tecnológicos que simplemente puede utilizarse en el producto que se está desarrollando. O el caso contrario, están innovando pero no se dan cuenta debido a que no tienen formas sistemáticas para detectar posibles innovaciones. La última observación relevante es que en las empresas hay emprendedores científicos y en las IES científicos emprendedores con un distinto concepto de abordar el PDT.

Tendencias similares a las presentadas en programa PEI, son registradas en el sector privado. Así que solo en Mexico, son muchas las empresas, IES y CPI que están interesadas en mejorar el desarrollo de los PDT. A través de nuevas alternativas metodológicas alineadas con la dinámica actual del mercado. Generando una fuerte relación con los clientes a largo plazo, donde el beneficio sea mutuo.

Bajo el contexto de la problemática expuesta durante el desarrollo de nuevos productos, esta investigación presentará una nueva alternativa metodológica que apoyará a resolver algunas de las causas del problema y retos de los PDT, donde, a través de una nueva forma de desarrollar el diseño mecánico, se apoyarán a lograr productos innovadores, logrando la satisfacción de los clientes, buscando sean múltiples los proyectos realizados con el mismo cliente. Además de tener una forma sistemática de identificar las innovaciones de los productos y lograr unificar los principios de métodos técnicos de diseño con métodos ágiles, obteniendo una forma sistemática para lograr proyectos exitosos tanto para ingenieros como científicos.

Este nuevo método podrá ser una alternativa para ser implementado en las empresas privadas, IES y CPI. En el caso de IES, los estudiantes con poca experiencia a través de este proceso metodológico podrán lograr productos igualmente innovadores en un corto periodo de desarrollo. Utilizando herramientas, técnicas y métodos del diseño mecánico adaptados a los principios ágiles.

1.4 DISEÑO METODOLÓGICO DE LA INVESTIGACIÓN

Este trabajo de tesis presenta una investigación aplicada exploratoria, con el objetivo de conocer, definir y esquematizar detalladamente las etapas de una nueva propuesta metodológica para el diseño mecánico aplicada en PDT, donde las principales etapas que se desarrollarán están mostradas en la figura 6.



Figura 6. Metodología de la tesis.
(Fuente: Elaboración propia)

1.4 OBJETIVOS

1.4.1 Objetivo general:

Desarrollar un nuevo método para el diseño mecánico, enfocado a los PDT.

1.4.2 Objetivos específicos:

- 1 Desarrollar una metodología que acople los métodos del diseño mecánico con los principios de metodologías ágiles.
- 2 Desarrollar una metodología que se adapte a los cambios en los productos físicos.
- 3 Aplicar el nuevo método a un proyecto real de diseño mecánico.
- 4 Aplicar las herramientas de realidad aumentada para lograr incrementos en el producto desde la etapa del diseño.

1.5 HIPÓTESIS

- 1 Si se acoplan los principios ágiles con metodologías, métodos y herramientas específicas del diseño mecánico, se pueden obtener un nuevo método sistemático que le permita a diseñadores con poca experiencia obtener productos innovadores y que se adapten a los cambios de los clientes. logrando la optimización del tiempo de desarrollo del producto así como el aumento de la satisfacción de los clientes.

CAPÍTULO 2. MARCO TEÓRICO

El fundamento de esta propuesta metodológica para el diseño mecánico se basa en los temas relacionados con el producto, los PDT, ciclo de vida del producto, métodos para el diseño mecánico y los métodos ágiles. Por lo anterior, a continuación se presentan algunos aspectos relevantes de cada tema.

2.1 TEORÍA DE LOS ELEMENTOS PRINCIPALES DE LA INVESTIGACIÓN

En la figura 7 se muestra la estructura de los elementos principales de esta investigación.



Figura 7. Estructura de la revisión del estado del arte.
(Fuente: Elaboración propia)

2.1.1 Producto

Existen varias definiciones del producto, entre las más relevantes definen al producto como un bien, idea, método, información, objeto o servicio creado como resultado de un proceso y sirve para satisfacer una necesidad o un requerimiento. Éste tiene una combinación de atributos tangibles e intangibles que lo definen como beneficios, características, funciones, usos, etc.

Ésta tesis enfatiza en la definición del producto basado en su nivel de madurez expuesto en una clasificación de la NASA. En la tabla 1 se presentan las características de tipos de productos dentro de la etapa de su madurez, desde que nace de una investigación científica básica hasta lograr su producción en serie. Se muestra dentro de esa clasificación

que los PDT, describen a los productos en dos esfuerzos de negocio: diseño del producto y desarrollo de prototipo comercial. Donde, el diseño del producto se presenta cuando los componentes básicos están integrados, estableciendo que funcionarán en conjunto. Las etapas del 4 al 6 representan el puente de la investigación científica a la ingeniería. Este nivel es el primero para determinar si los componentes individuales trabajarán juntos como un sistema. Un sistema de laboratorio muy probablemente será una mezcla de equipo manual y componentes que requieran manejo especial, calibración o alineación para que funcionen en conjunto.

En el desarrollo de un prototipo comercial, los componentes tecnológicos básicos son integrados a manera de que la configuración del sistema sea similar a una aplicación final en casi todas sus características. Se dan pruebas a escala en laboratorio y un sistema operativo condicionado. La diferencia mayor entre el nivel 4 y 5 es el incremento en la fidelidad del sistema y su ambiente hacia la aplicación final. El sistema probado es casi un prototipo funcional.

Aunque el diseño del producto y el prototipo comercial, se desarrollan aun en ambiente laboratorio, se espera que sus componentes ya sean integrados a manera que la configuración del sistema sea de aplicación final. Lo cual genera productos que probablemente solo se realizarán en una ocasión, ya que combinan aspectos de investigación con la exigencia de obtener un producto funcional. Con lo anterior se puede clasificar a los tipos de productos en tres grupos como se muestra en la figura 8: El primer grupo se refiere a los productos de investigación, el segundo grupo se refiere a los PDT y el tercero como productos que se manufacturan en serie, los cuales ya alcanzaron el nivel 9 de madurez y están listos para lanzarse al mercado masivamente.

Tabla 1. Etapas de maduración tecnológica.
(Fuente: NASA "Technology Readiness Level")

Nivel TRL	Nivel relativo de la tecnología	Definición	Descripción Concepto industrial	Esfuerzo de Negocio
0	Investigación científica básica			
1	Investigación aplicada - tecnológica básica	Observación y reporte de principios básicos	Comienza la investigación científica básica. Se comienza la transición a investigación aplicada	No hay ninguna actividad de negocios
2	Validación conceptual	Concepto de la tecnología o formulación de la aplicación	Las aplicaciones son aún especulativas y puede aún no haber pruebas o análisis detallados que confirmen dichas suposiciones.	Se comienza a formular posibles usos o aplicaciones de la tecnología.
3		Prueba de concepto II	Las actividades que se llevan a cabo son fuertemente de investigación y desarrollo, que incluyen estudios analíticos y estudios a escala laboratorio para validar físicamente las predicciones de los elementos separados de la tecnología.	Inicio de la validación de la idea de aplicación - posible producto - posible mercado
4	Desarrollo tecnológico	Validación de componentes o sistema en un ambiente de laboratorio.	Los componentes básicos están integrados, estableciendo que funcionarán en conjunto. Las etapas del 4 al 6 representan el puente de la investigación científica a la ingeniería. Este nivel es el primero para determinar si los componentes individuales trabajarán juntos como un sistema. Un sistema de laboratorio muy probablemente será una mezcla de equipo manual y componentes que requieran manejo especial, calibración o alineación para que funcionen en conjunto.	Diseño de producto
5		Componentes integrados a manera que la configuración del sistema sea similar a su aplicación final. Su operatividad es aún a nivel laboratorio.	Los componentes tecnológicos básicos son integrados a manera de que la configuración del sistema sea similar a una aplicación final en casi todas sus características. Se dan pruebas a escala en laboratorio y un sistema operativo condicionado. La diferencia mayor entre el nivel 4 y 5 es el incremento en la fidelidad del sistema y su ambiente hacia la aplicación final. El sistema probado es casi prototipo.	Desarrollo de prototipo comercial
6	Demostración tecnológica	Sistema de ingeniería en validación en ambiente en condiciones relevantes a las reales operativas. Aún a nivel prototipo.	Prototipo piloto con ingeniería con condiciones de escalamiento que le permitirán a la tecnología llegar a un sistema operativo. El prototipo debe ser capaz de desarrollar todas las funciones requeridas por un sistema operativo.	Demostración de mercado - early adopters - Pruebas "Beta"
7	Comisionamiento de sistemas	Prototipo completo demostrado en ambiente relevante.	Prototipo final con sistema operativo funcional.	Primer corrida piloto y pruebas finales reales
8		Sistema final completo y evaluado a través de pruebas y demostraciones	La tecnología ha sido probada en su forma final y bajo condiciones supuestas. En muchos casos significa el final del desarrollo del sistema.	
9	Operación del sistema	Operación de Sistemas	La tecnología se encuentra en su forma final y operable en un sin número de condiciones operativas.	Entrega de producto para producción en serie y comercialización.



Figura 8. Clasificación de tipos de productos.
(Fuente: Elaboración propia)

2.1.2 Proyectos de Desarrollo Tecnológico (PDT).

Los PDT tienen como objetivo principal obtener un resultado dentro de un plazo de tiempo limitado, con un principio y un fin que determinan el alcance y los recursos. Para ello se estructura en función de actividades, que discurren de forma secuencial o paralela en los distintos tipos de proyectos.

Los PDT se sustentan en actividades de aplicación sistemática del conocimiento y que están dirigidos a hacer un uso más eficiente de los recursos disponibles, son el componente fundamental de toda política de innovación, tanto a nivel empresarial como a nivel nacional. A nivel empresarial, los proyectos tecnológicos contribuyen a situar a las organizaciones que los emprenden en una posición adecuada para enfrentar los desafíos que surgen en un mercado cada vez más competitivo y globalizado. A nivel nacional, estos proyectos contribuyen a construir una base sólida para el planteamiento estratégico de actividades tecnológicas en todos los niveles de la economía.

Los PDT se distinguen de otro tipo de proyectos esencialmente en sus resultados. Surgen cuando una organización invierte para obtener el valor potencial de un recurso con fines

de innovación. Es decir, que tanto los objetivos como el resultado de todo proyecto tecnológico es el desarrollo y aplicación de una tecnología.

Las principales características de un PDT, que establece la norma NMX-GT-002-IMNC-2008, son:

- Emanan de estrategias competitivas de la organización.
- Se orientan a la obtención y/o aplicación de conocimientos, a la creación de un prototipo o a la innovación de productos, procesos o servicios.
- Sus actividades comprenden la asimilación, investigación, desarrollo, innovación y explotación de tecnologías.
- Presentan un alto grado de incertidumbre y puede derivar en un resultado no esperado.
- Utilizan un control contable específico.

Los PDT requieren un tratamiento especial ya que basado en sus características no se pueden clasificar como productos que serán manufacturados en serie ni tampoco como productos que serán solo en investigación. Ese punto medio que define a este tipo de proyectos complica el tratamiento bajo una metodología estándar que actualmente existe. Sin embargo, los PDT requieren entregables medibles y bien definidos para lograr la satisfacción del cliente.

2.1.3 Ciclo de Vida de los PDT

Las fases del desarrollo de los PDT, se diferencia de otros tipos de proyectos en función de la madurez del producto, así como de todas las fases que pasará el producto definidas por los objetivos de cada organización que se dedica a la ejecución de este tipo de proyectos. Sin embargo, la mayoría de estas organizaciones dedicadas al desarrollo de nuevos productos tienen fases comunes que pueden presentarlas como genéricas. Frecuentemente usan las 7 fases de los procesos estándares para la ejecución de los PDT que se muestran en la figura 9.

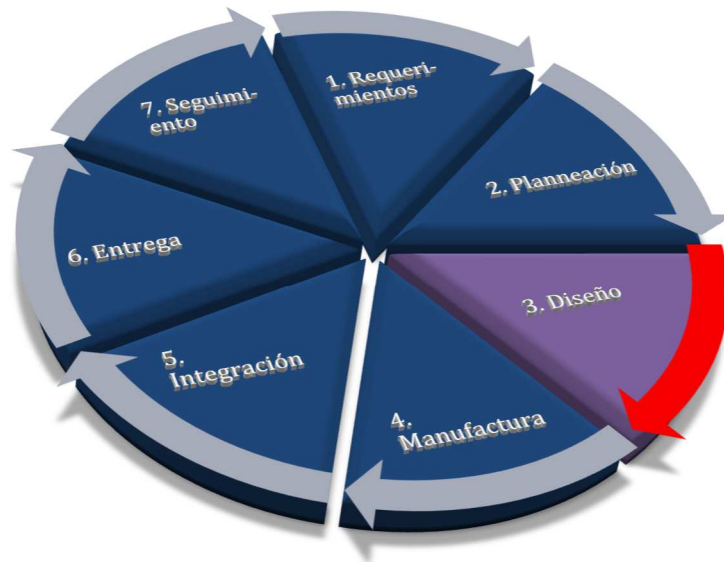


Figura 9. Fases del ciclo de vida típico de los PDT.
(Fuente: Elaboración propia)

Cada una de las fases tiene objetivos establecidos y expuestos en múltiples bibliografías, de las cuales se puede definir brevemente para contextualizar la etapa de diseño que será el enfoque principal en esta investigación.

2.1.3.1 Fase de requerimientos

Entender el problema es la base para definir el proyecto, la tarea principal de esta etapa es trasladar las necesidades subjetivas del cliente en requerimientos. Está claro que encontrar el problema correcto desde esta etapa será gran avance en la solución del proyecto.

2.1.3.2 Fase de planeación

Se trata de la definición de todos los alcances del proyecto. Desde identificar y conocer todas las partes interesadas, fijar y priorizar objetivos, definir los entregables, crear un programa de proyecto, así como identificar riesgos para que el proyecto sea aprobado por todas las partes interesadas. La aprobación de todos los alcances del proyecto se

establecerán como los objetivos que el cliente espera al término del proyecto. Aquí es donde presenta una contradicción con el planteamiento para los PDT. Ya que se establece que una de sus características es el alto nivel de incertidumbre y se pudieran obtener resultados diferentes a los esperados por los interesados en el proyecto. Por lo anterior es fundamental entender el nivel de desarrollo del producto en cuestión para definir claramente los alcances del proyecto como se muestra en la figura 10. Es muy común en los PDT, ser demasiado ambiciosos y querer saltar niveles de tecnología.

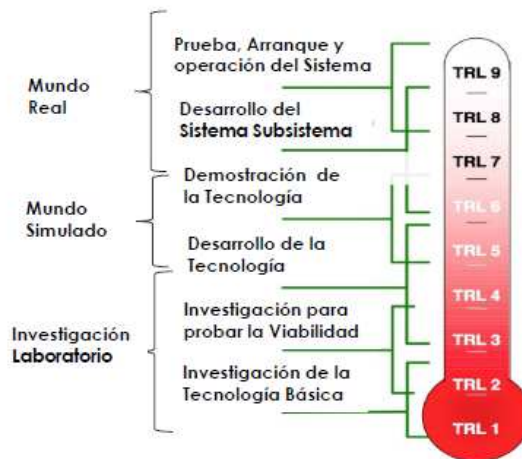


Figura 10. Niveles de madurez de los proyectos.
(Fuente NASA "Technology Readiness Level")

2.1.3.3 Fase de diseño

Una vez acordados y declarados los alcances del proyecto en un documento contractual se proceden al diseño del producto. Esta etapa se considera la plataforma del proyecto ya que la expectativa del proceso de diseño es trasladar todas las necesidades y requerimientos en soluciones físicas, que declararán el cumplimiento o desviación de los alcances acordados y declarados en el documento contractual.

2.1.3.4 Fase de manufactura

En esta etapa se materializan todos los elementos resultantes del diseño del producto. En la mayoría de los casos serán necesarias una multitud de operaciones de manera que, dependiente del tipo de producto será el conjunto de procesos a utilizar para la materialización del producto.

2.1.3.5 Fase de integración

Rara ocasión un producto es aislado de otros elementos. Entonces con todos los elementos manufacturados en la etapa anterior, se procede al ensamble de estos para lograr sistemas que cumplan con las funciones establecidas durante la etapa de diseño. Parte importante de esta etapa es la puesta a punto la cual representa el mayor reto porque dependerá de este ajuste final para el correcto funcionamiento del producto como conjunto de elementos.

2.1.3.6 Fase de entrega

Finalmente el producto se someterá a la evaluación del cliente para determinar si este producto cumple con todos los entregables solicitados. Se formaliza una entrega recepción posterior a la confirmación del cumplimiento de requisitos declarados en el documento contractual. Se realizará la transferencia del equipo y por ende la responsabilidad del correcto funcionamiento.

2.1.3.7 Fase de seguimiento

En algunas empresas se le conoce a esta etapa como la garantía del producto, en el cual se le dará seguimiento durante su operación, para atender alguna posible falla que sea cubierta por la garantía, o en ocasiones se requieren actualizaciones o modificaciones, las cuales salen de los alcances de la garantía pero se requieren realizar por requerimientos adicionales del cliente.

2.1.4 Teoría del diseño

En el campo del diseño existen muchas divisiones tales como: diseño eléctrico, diseño de la programación, diseño de control, diseño mecánico, entre otras, como se muestra en la figura 11. Entonces la naturaleza del PDT determinará el tipo de diseño que utilizaremos o tal vez será el conjunto de tipos de diseño para desarrollar un producto integral. En la mayoría de los PDT partimos de un concepto que represente la física básica del producto. Entonces se puede establecer que el diseño mecánico es la base para el primer acercamiento hacia la solución que se presentará al final del proyecto.

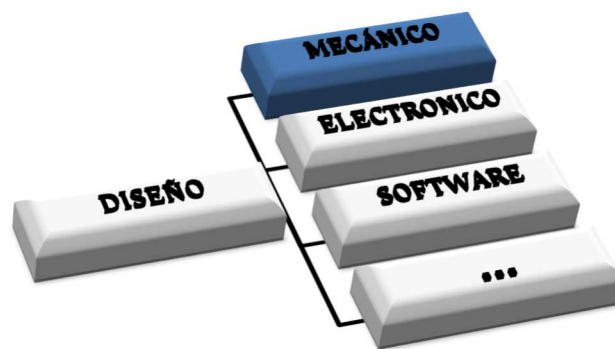


Figura 11. Tipos de diseño.
(Fuente: Elaboración propia)

Aunque existen varios tipos de diseño, las metodologías son genéricas y podrían adaptarse a cualquier rama del diseño. Por lo tanto en este estudio del estado del arte generalizamos la búsqueda de temas relacionados con el diseño mecánico.

El estudio de los procesos de diseño, teorías del diseño y metodologías han sido una preocupación de los ingenieros, diseñadores e investigadores en las últimas 6 décadas. Por más de 40 años, varias propuestas del diseño han sido expuestas por investigadores, diseñadores e ingenieros, con aplicación tanto para la academia como para la industria, exponiendo sobre cómo el diseño debe realizarse. Dichas propuestas han sido consideradas como filosofías modelos y métodos del diseño como lo expone en su investigación Evbuomwan (7).

2.1.4.1 El diseño como Filosofía

Varias universidades, investigadores y diseñadores han presentado propuestas de cómo es el diseño y cómo debería ser realizado, lo cual ha generado varias controversias. Tres escuelas británicas afines al diseño e inmiscuidas en la temática del diseño han presentado alternativas del diseño, como lo presenta Broadbent (8) en el libro "*Design: science: method*". El primer grupo cree que el diseño debe ser caótico y creativo, el segundo grupo considera que el diseño debería ser organizado y disciplinado, mientras que el tercer grupo cree que nunca se le debería imponer un proceso a un diseñador. Soportado por el primer punto de vista frecuentemente se considera al diseño como un arte considerando que los diseñadores nacen y no se hacen (9). Como soporte del segundo punto de vista los métodos sistemáticos son usados cuando algunas de las siguientes condiciones se presentan en el diseño; cuando la consecuencia de un error será grave: cuando la probabilidad de fallar es alta y/o cuando el número de variables que interactúan en el diseño es alta, de tal manera que conviene realizar un desglose de tareas (10). Como soporte del tercer punto de vista, Lawson's (11) realizó un experimento, planteando un mismo problema a diseñadores y a científicos. Los científicos realizaron una estrategia sistemática explorando el problema, buscando encontrar bajo las leyes la solución correcta u óptima. En contraste, los diseñadores sugirieron varias posibles soluciones hasta que encontraron una solución buena o satisfactoria. La evidencia de este experimento sugiere que los científicos resuelven el problema basado en el análisis, mientras que los diseñadores resuelven el problema por síntesis. Los científicos se concentran en estrategias para resolver el problema, mientras que los diseñadores se centran en estrategias para resolver la solución.

2.1.4.2 El diseño como modelo

Los modelos del diseño son representaciones de filosofías o estrategias propuestas para mostrar cómo es el diseño, o cómo se debería realizar. Frecuentemente, son mostradas en diagramas de flujo, mostrando un proceso iterativo con retroalimentación entre etapas. En el pasado, los modelos de diseño surgieron de varios puntos de vista filosóficos que pueden clasificarse principalmente en dos tipos de modelos, como: descriptivos y prescriptivos, declarado por Evbuomwan (7). Los modelos prescriptivos son asociados con propuestas académicas que muestran el diseño como un proceso global. Cubriendo varios

pasos para desarrollar el diseño teórico. Los modelos descriptivos muestran las acciones de los diseñadores y actividades durante el proceso de diseño. Recientemente, otro grupo de modelos son presentados que enfatizan el uso de técnicas computacionales.

2.1.4.3 El diseño como método

Durante las diferentes fases de un proyecto de diseño, herramientas y técnicas son utilizadas como apoyo para el desarrollo del diseño, con la finalidad de obtener productos funcionales. El uso de herramientas y técnicas en el diseño son consideradas como métodos del diseño. Los métodos del diseño generalmente ayudan a formalizar y sistematizar actividades dentro de los procesos, así como externar el pensamiento de diseño, esto es, los métodos tratan de extraer las ideas y pensamientos del diseñador y exponerlos en formatos y/o diagramas.

Hubka (12) define un método de diseño, como un sistema de reglas y directrices metodológicas, las cuales ayudan al diseñador a determinar la manera de proceder para un comportamiento particular en una actividad de diseño, y regular la colaboración con las técnicas disponibles. Los métodos del diseño también son estructurados para exhibir ciertas características en términos de su uso, tales como los objetivos del servicio del método, su aplicabilidad general, condiciones bajo las cuales pueden ser utilizados, si están direccionados para un diseñador o para un grupo, sus orígenes, así como sus funciones y el tiempo demandado por el método. Tomando en cuenta las características mencionadas, los métodos del diseño pueden ser clasificados como:

- Métodos para dar mejoras básicas en el trabajo de diseño.
- Métodos que consideran la creatividad del diseñador.
- Métodos que describen y caracterizan un problema bajo un proceso lógico y matemático.
- Métodos que establecen leyes y regulaciones que aumenta la probabilidad de éxito del diseño.
- Métodos basados en conocimiento específico de un producto.
- Métodos que apuestan por el uso de técnicas, principios y procedimientos, buscando una automatización del proceso de diseño.

- Combinación de los métodos ya mencionados, enfocados a un caso específico.

Otros métodos de diseño se han presentado recientemente, tales como: Diseño axiomático, diseño para manufactura, entre otros.

2.1.4.4 El diseño como sistema

Recientemente, varios problemas han sido resueltos gracias al uso de sistemas de cómputo (CAD), donde los sistemas pueden ser modelados y simulados en condiciones muy cercanas a la realidad. Logrando estimar vida útil del producto, extraer dibujos automáticos así como lista de partes en segundos. Al principio el único enfoque de los sistemas CAD era solo para representar la geometría del diseño, sin embargo, en la actualidad se organiza el diseño desde la planeación hasta la entrega. Con el acoplamiento de algunos modelos de diseño, herramientas o técnica se pueden lograr diseños innovadores.

El diseño mecánico ha requerido conocimientos de solo partes y ensamblajes mecánicos. Sin embargo, en la actualidad la mayoría de los dispositivos mecánicos contienen componentes electrónicos. Desde la segunda guerra mundial en 1940 se tuvo el auge de la mezcla de componentes de control electrónico para los sistemas mecánicos. Desde esa época los ingenieros mecánicos tienen que combinar los componentes electrónicos con los sistemas mecánicos. Estos sistemas electrónicos han venido revolucionando la industria con la incorporación de sistemas computacionales y de lógica compleja. Muchos de los electrodomésticos en la actualidad incluyen microprocesadores, como cámaras, celulares, autos, entre otros. Entonces esa combinación de campos del conocimiento en la actualidad se le conoce como mecatrónica, la cual incluye la combinación de la mecánica, la electrónica y la programación. Lo cual hace al diseño más complejo por el empalme de estas disciplinas durante el desarrollo del diseño. Pero es muy marcado que ningún dispositivo electrónico ni de programación es útil, si no se tiene un producto para implementarse, lo cual enfatiza la importancia del diseño mecánico del producto como premisa en los diferentes tipos de diseño.

Entonces, la creación de un producto requiere un análisis extenso para lograr un diseño que cumpla todas las características y atributos esperados con el cliente, así que todas las

metodologías actuales de diseño ponen mucho empeño y cuidado en esta etapa lo cual conlleva un tiempo significativo en el desarrollo del diseño mecánico.

2.1.5 Métodos

2.1.5.1 Características de las metodologías de diseño

Hay un gran número de características (o propiedades) que podrían ser propuestas por cada metodología de diseño. Como ejemplo, se proponen los 8 procesos técnicos de las etapas de diseño que expone Evbuomwan (7). Los cuales manejan 10 características, mostradas en la figura 12, que deben soportar a las metodologías de diseño.



Figura 12. Características deseadas en las metodologías del diseño.
(Fuente: Modificación, Evbuomwan (7))

Todas las características de la tabla 2 representan facetas únicas que la metodología del diseño debe usar con la finalidad de ejecutar efectiva y eficazmente los procesos técnicos del diseño, durante las etapas del ciclo de vida de los productos. Las primeras letras de cada una de las características pueden ser combinadas para crear un acrónimo

ERICOITIMI (explorar, racional, investigación, creativa, oportunista, incremental, toma de decisión, iterativa, multidisciplinaria, interactiva).

Tabla 2. Características de una metodología.
(Fuente: Modification, Evbuomwan (7))

Característica	Descripción
1. Explorar	El diseño es un esfuerzo profesional y formal que requiere conocimientos, capacidades y habilidades específicas.
2. Racional	Diseño en un proceso de razonamiento lógico, análisis matemático, simulaciones computacionales, prototipos y pruebas, etc.
3. Investigación	Diseño requiere la investigación de todos los requerimientos de los interesados del proyecto, búsqueda de técnicas de diseño disponibles, búsqueda de soluciones ya desarrolladas, lecciones aprendidas y sus modos de falla que la originaron, etc.
4. Creativa	Diseño requiere saber cómo hacerlo, ingenio, memoria, reconocimiento de patrones, pensamiento lateral, etc.
5. Oportunista	El equipo de diseño debe ser capaz de definir propuestas de ingeniería inversa o secuencial dependiente de la situación presentada
6. Incremental	Las mejoras o refinamiento deben ser usadas en el diseño con la finalidad de cada vez tener diseños más robustos.
7. Toma de Decisión	Diseño requiere evaluaciones con datos. La selección de soluciones es basada en la experiencia y los criterios dados por los interesados en el proyecto.
8. Iterativa	El diseño es iterativo, el producto es analizado respecto a los requerimientos funcionales y no funcionales, restricciones, y costos. Basados en revisiones o retroalimentación es como se refina el diseño hasta logra el diseño final.
9. Multidisciplinaria	El diseño requiere de un equipo multidisciplinario
10. Interactiva	El diseño es interactivo. El equipo de diseño es una parte integral de diseño actual.

Los equipos de diseño invocan diferentes modelos de pensamiento durante la ejecución de una metodología de diseño. El tipo particular de pensamiento es una función de un punto de ejecución en una metodología y será el único proceso utilizado. Dos principales modos de pensamiento existen en las metodologías; DIVERGENCIA y CONVERGENCIA. Las cuales son interrelacionados y complementarios. La secuencia de la divergencia y convergencia se representa por el "DOUBLE- DIAMOND MODEL of DESIGN (NORMAN (13))" presentadas en la figura 13.

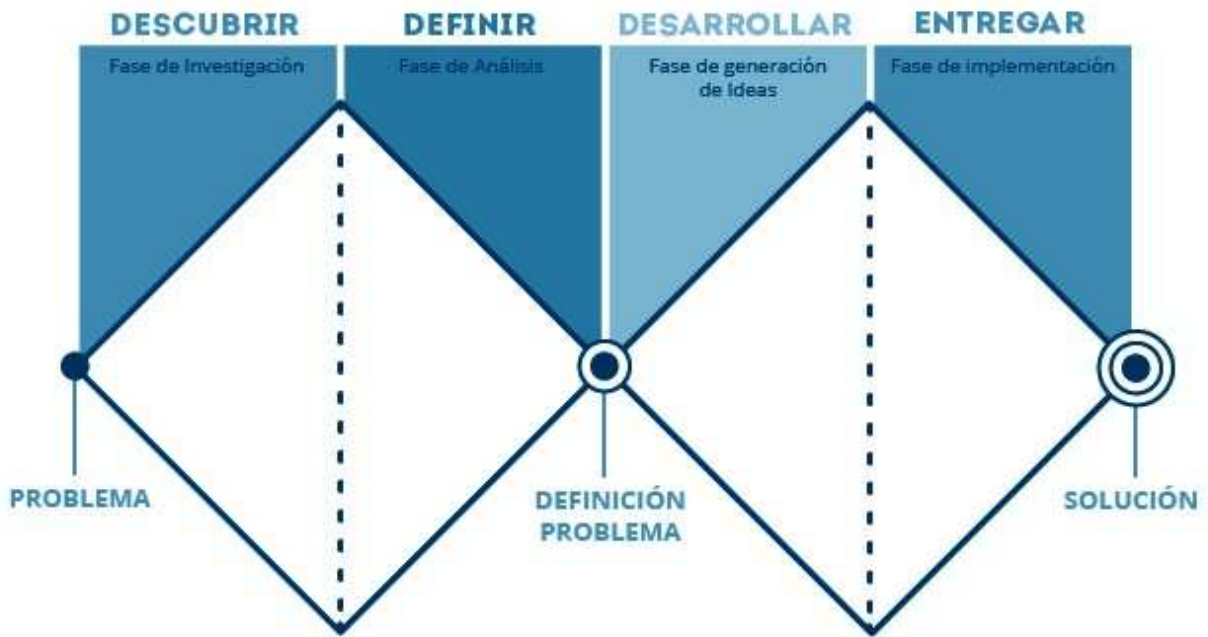


Figura 13. Modelo de diseño, Diamante-Doble.
(Fuente: Modificación, NORMAN (13))

La idea detrás de Diamante-Doble es que cuando una idea de diseño es concebida, la primera acción es expandirla al equipo de diseño (divergencia) permitiendo que el equipo explore todos los problemas relacionado con esa idea de diseño. Una vez que todas las ideas relacionadas a la idea del diseño son revisadas, el equipo puede enfocarse en, ¿Cómo podría usarse en el diseño? (convergencia). Después que el equipo ha decidido ¿Qué debe contener el diseño?, el equipo debe otra vez expandir el pensamiento para revisar todas las posibles alternativas del diseño (divergencia). Finalmente, una vez que se ha identificado y revisado una alternativa de solución, el equipo se enfoca en seleccionar la mejor y única opción (convergencia).

2.1.5.2. Modelos metodológicos para el diseño

Tendencias de cómo abordar el proceso de diseño, son expuestas en investigaciones recientes de Adams (14) y Tomiyama, Lutters, Kind Ch., & Kimura F. (15). Para contextualizar la gama de propuestas metodológicas en el diseño se pueden clasificar los tipos de diseño en la tabla 3, basados en las recomendaciones de Cross N. (10), Takeda H. (16) y Finger & Dixon (17). Aunque muchos de estos modelos muestran tendencias hacia el uso de

herramientas más complejas, procesos y métodos basados en algoritmos, tales como el despliegue de funciones de calidad (QFD), teoría de resolución de problemas inventiva (TRIZ), diseño axiomático, entre otros. Sus principios se mantienen bajo un proceso en cascada, con la filosofía de encontrar la mejor solución posible, aunque en algunos casos, las soluciones no son necesariamente realistas o no están alineadas con las expectativas del cliente. Lo que abre la oportunidad de innovar con una nueva alternativa metodológica que muestre tendencias hacia una evolución del diseño, con productos robustos técnicamente fundamentados y ágiles.

Tabla 3. Clasificación de modelos de diseño.
(Fuente: Elaboración propia)

Tipo de modelo	Breve descripción y autores de los modelos
Descriptivo	Muestra la secuencia de actividades que ocurren en diseño (heurístico), dentro de los cuales se puede mencionar modelos de: March (18), Mattchet & Briggs (19), Gero & Hybs (20), entre otros.
Prescriptivo	Prescriben un patrón de actividades de diseño, dentro de los cuales se puede mencionar modelos de: Jones & Thornley (21), Asimow (22), Pahl & Beitz (23), VID (24), Watts (25), Marples (26), Archer (9), Krick (27), Cross (10), Hubka (12), French (28), Harris (29), Pugh (30), BS-7000 (31), Suh (32), Taguchi (33), Ullman (34), entre otros.
Cognoscitivo	Explica el comportamiento del diseñador, dentro de los cuales se puede mencionar modelos de: Huysentruyt (35), Yoshikawa (36), Gero (37), Tomiyama (38), Takeda (16), Lhote (39), Visser (40), entre otros.
Computacional	Expresan la forma en que una computadora podría desarrollar la tarea de diseño: como lo exponen: Agogino (41), Neville (42), Mostow (43), Dixon (44), Gero (37), Fitzhorn (45), Cagan & Agogino (46), entre otros.

Debido a la relevancia de esta propuesta orientada hacia un modelo prescriptivo, a continuación se mencionan algunos de los modelos de diseño más relevantes en el campo prescriptivo.

2.2 MÉTODOS EN EL DISEÑO

2.2.1 Diseño metodológico

El proceso de diseño propuesto por Dym y Little (47) se basa en un estilo de cascada lineal. Usan un modelo de proceso de diseño descriptivo y prescriptivo, intentando ambos describir los elementos (o etapas) del proceso de diseño, al mismo tiempo que se prescriben las tareas a realizar durante el proceso de diseño. Una característica interesante de modelo es que los autores identifican las etapas de pre y post-procesamiento como la definición del problema y fases de comunicación de diseño, respectivamente.

Dym y Little (47) sugieren y describen numerosas herramientas y métodos útiles en diversas etapas del proceso de diseño. Señalan que el público objetivo para la aplicación de las herramientas descritas, debe ser un equipo de diseño pequeño. Dicho modelo alienta esencialmente las tareas en cada paso para llevarse a cabo en orden. También sugieren algunas estrategias, para motivar a los diseñadores a buscar más de una solución, y que no comprometan solo un concepto o configuración particular, sin que éstos sean reforzados con información adicional o soluciones alternativas.

El enfoque Dym y Little (47), es la descomposición de un problema más grande en sub-problemas (ideas o entidades). También sugieren numerosos métodos formales para apoyar el proceso en diferentes etapas, especialmente durante la definición del problema y diseño conceptual. La técnica de los árboles de objetivos puede ayudar para aclarar y comprender las declaraciones del proyecto. Mientras que los gráficos de comparación por pares, ayudan a priorizar los objetivos desde el principio, así como establecer métricas formales para ayudar a evaluar un diseño en comparación con los objetivos.

El método de especificación de rendimiento expuesto por Dym y Little (47), ayuda a crear criterios de evaluación mediante la identificación de atributos y especificaciones de rendimiento. Una matriz morfológica ayuda explorar alternativas de diseño que proporcionen los medios para una función específica. Diferentes estilos de las matrices de evaluación ayudan a seleccionar la solución de diseño. La adquisición de información es crucial para comprender el problema, las restricciones y los objetivos. Dym y Little (47) sugieren una serie de métodos que incluyen: revisiones de literatura, encuestas de usuarios y cuestionarios para estudios de mercado, grupos focales, entrevistas informales, entrevistas

estructuradas, intercambio de ideas, sináptica o analogías, “benchmarking” de productos competitivos e ingeniería inversa.

Más adelante en el proceso de diseño, Dym y Little (47) sugieren una serie de métodos para analizar la información y los resultados de las pruebas. Evaluando si los objetivos del diseño se lograron. El rendimiento funcional debe describirse en términos de requisitos y especificaciones. Los experimentos de campo o laboratorio proporcionan información útil, mientras se definen si los resultados son suficientes o apropiados para ayudar a determinar si un concepto debe ser aceptado o rechazado durante la prueba del diseño conceptual.

En caso de tener incertidumbre de la funcionalidad del diseño, un prototipo rápido puede proporcionar información útil. Incluso si tiene una capacidad funcional limitada, dicho prototipo pueden reducir costos y tiempo en el proyecto. Si el desarrollo del prototipo es difícil debido al costo, tamaño o peligro, la simulación con un modelo analítico, informático o físico puede ser posible. Existen métodos que atienden a detalle la creación de modelos matemáticos, así como selección de materiales y técnicas para fabricación del prototipo. Obtener comentarios sobre el concepto durante la etapa del diseño es importante para garantizar que cumple con los requisitos y para averiguar formas en que puede ser mejorado. Algunos métodos para obtener retroalimentación del diseño incluyen: reuniones regulares con clientes y usuarios, revisiones de diseño formales con clientes, usuarios y partes interesadas. En algunos proyectos, se pueden requerir audiencias públicas. Dym y Little (47) sugiere un modelo lineal similar al modelo de proceso de diseño, resumido en la siguiente lista de puntos:

1. Definición del proyecto (Alcance)

- (a) Estudio de viabilidad
- (b) Reunión de inicio del proyecto
- (c) Definición del Alcance; límites de presupuesto y horario establecidos
- (d) Acta constitutiva

2. Estructura del proyecto

- (a) Equipo del proyecto
- (b) Tareas a desarrollar: estructura de desglose del trabajo establecido

3. Programación del proyecto

- (a) Asignación de tareas: Carta de Responsabilidad Lineal establecida
- (b) Recursos y tareas programadas
- (c) Presupuesto definido

4. Seguimiento, evaluación y control del proyecto

- (a) Trabajo, tiempo y costo
- (b) Trabajo real y planificado
- (c) Tendencias analizadas
- (d) Revisión de planes según sea necesario

Desde una perspectiva abstracta, Dym y Little (47) afirman que la gestión de proyectos debe estar equilibrada en términos del alcance, gasto y cronograma de un proyecto. Señalan que los proyectos de diseño son más difíciles que otros tipos de proyectos, tales como los de la industria de la construcción, debido a las muchas incertidumbres y ambigüedades en estas tres categorías.

Herramientas sugeridas para la gestión de proyectos deben incluir; actas constitutivas del proyecto, estructuras de desglose del trabajo (WBS), tablas de responsabilidad lineal (LRC), equipo calendarios, diagramas de Gantt, redes de actividades, presupuestos y matrices de porcentajes completos (PCM). El proceso de diseño y los modelos de gestión de proyectos presentados por los autores son muy lineales en naturaleza, con un énfasis limitado en la iteración y la retroalimentación. Los procesos y métodos tradicionales son muy claros, casi algorítmicos por naturaleza, y favorecen a realizar las cosas una sola vez y en orden. Sin embargo, dichos métodos argumentan que la siguiente idea apuntalan al éxito de los proyectos de diseño: primera, la buena comunicación es crítica, especialmente entre diseñador, cliente y usuario. Esto podría ampliarse para incluir a todos los interesados. Además, se debe encontrar un "lenguaje" apropiado para los diferentes tipos de comunicación, por ejemplo: palabras, imágenes, números, reglas, propiedades, dibujos, etc., especialmente al aclarar al cliente los objetivos. En segundo lugar, hacer preguntas cada vez más detalladas a lo largo del proceso ayuda a conducir cada paso, y refinar continuamente el problema y la solución. Esto es indicativo de la importancia de iteración. Otro punto es que dichos modelos de gestión de proyectos puede ser en realidad

probabilísticos con la finalización abierta del diseño. Finalmente, los autores solo mencionan brevemente que la teoría concurrente entre la ingeniería y el diseño para la fabricación puede ser importante para el éxito, sin detallar metodológicamente la interacción de los interesados. Introduciendo únicamente al diseño concurrente para trabajar activamente con el equipo de partes interesadas en el proyecto, para diseñar colectivamente un producto, así como considerar aspectos de la fabricación y otras fases que se desarrollarán en el ciclo de vida del producto.

2.2.3 Diseño en Ingeniería

Dieter y Schmidt (48) presentan varias ideas y herramientas para el diseño en el campo de la ingeniería. El enfoque principal está dirigido al desarrollo de productos físicos. Hay una tendencia hacia más complejos y procesos o métodos basados en algoritmos, como el despliegue de funciones de calidad (QFD), la teoría de resolución de problemas inventiva (TRIZ / TIPS) y diseño axiomático. El proceso de diseño es agrupados en tres fases principales: diseño conceptual, diseño de la realización y diseño de detalle. Pasos y algunas tareas sugeridas se resumen de la siguiente manera:

- Diseño conceptual
 - Definir problema
 - Planteamiento del problema
 - “Benchmarking”
 - QFD
 - Planificación de proyectos
 - Recopilar información
 - Internet
 - Patentes
 - Comercio
 - Literatura
 - Generación de conceptos
 - Lluvia de ideas
 - Descomposición funcional
 - Tabla morfológica
 - Evaluación de conceptos
 - Selección del concepto de Pugh

- Matrices de decisión
- Diseño de la realización
 - Arquitectura de producto
 - Definición del elemento físico para llevar a cabo la función
 - Diseño de la configuración
 - Selección preliminar de materiales y fabricación
 - Modelado y dimensionamiento de piezas
 - Diseño paramétrico
 - Diseño robusto
 - Tolerancias
 - Dimensiones finales
 - Diseño para fabricación
- Diseño de detalle
 - Dibujos detallados y especificaciones

Esto sugiere un proceso muy lineal, similar a un proceso de cascada. Por otro lado, Dieter y Schmidt (48) describen un modelo básico del proceso de diseño, similar al modelo propuesto por Asimow (22). Este modelo ilustra un ciclo de retroalimentación con las siguientes fases: información específica y general → Operación de diseño → Resultado → Evaluación. Donde la evaluación exitosa indica una transición al siguiente paso, de lo contrario, el ciclo de retroalimentación fluye de regreso al principio del proceso. Esto indica una clara noción de que la iteración proporciona una base para la mayoría de las actividades de diseño, y esto debe aplicarse en fases y pasos sugeridos en el modelo. Más allá del diseño, las fases adicionales del proyecto pueden incluir fabricación, distribución, uso y retiro. Como base para el modelo del proceso de diseño, Dieter y Schmidt (48) presentan un modelo básico de resolución de problemas: definición del problema, recopilación de información, generación de soluciones alternativas, evaluación de alternativas y toma de decisiones, y finalmente, comunicación de los resultados. Además, ellos comparan el método de diseño con el método científico. Se sugiere que el diseño se divida en diferentes categorías: diseño original / innovador / disruptivo, diseño adaptativo, rediseño o diseño de variantes, diseño de selección y diseño industrial.

Diseño original: la originalidad del diseño es rara y a menudo incluye la invención. Tiene el potencial de perturbar significativamente los mercados.

Diseño adaptativo: es más común y aprovecha la síntesis de una solución existente adaptándola a una nueva aplicación.

El rediseño: ofrece una variante o mejora de un diseño existente y es una actividad común.

El diseño de selección: se refiere al proceso de identificación de componentes estándares que cumplen con las especificaciones del producto.

El diseño industrial: es típicamente más artístico, con el objetivo de mejorar interacción humana y atractivo.

Los productos en cada categoría de diseño tendrán diferentes actividades, riesgos y objetivos. Los autores explican que un producto de calidad y rentable comienza en la fase de diseño; donde, malas decisiones durante el diseño, aumentan los costos y los problemas en la fabricación y en las siguientes fases del ciclo de vida del producto. Enfatizando que "El proceso de diseño debe llevarse a cabo para desarrollar productos de calidad y costo competitivo en el menor tiempo posible". También sugieren que muchos proyectos de diseño pueden no desarrollarse debido a la falta de viabilidad económica o técnica. En estos casos, se crea nueva información durante el proceso de diseño del sistema, y esta información representa la experiencia del equipo de trabajo. Un diseño exitoso y de calidad debe ser evaluado contra el logro de los requisitos de desempeño, normativos y sociales (como el medio ambiente). Esta visión holística considera todas las partes interesadas y las limitaciones externas en un proyecto. Los autores argumentan que las revisiones de diseño son importantes para garantizar el éxito en un proyecto de diseño. Un diseño revisado, es un estudio retrospectivo del diseño hasta ese momento. Proporciona una sistemática forma de identificar problemas con el diseño, determinar planes de acción futuros e iniciar acción para corregir cualquier área problemática. "Las revisiones de diseño deben ser atendidas por una variedad de partes interesadas, y cubrir los aspectos técnicos y comerciales del diseño. Brinda una oportunidad para monitoreo y retroalimentación tanto

desde el punto de vista de la aceptabilidad del diseño, como desde el punto de vista de la gestión del proyecto. Esto sugiere que las revisiones formales del diseño se lleven a cabo entre las fases. El diseño es fácil de cambiar antes de finalizar el diseño conceptual, las interfaces son importantes después del diseño de la realización, y el producto debe estar listo para la fabricación después del diseño de detalle.

2.2.4 Metodologías tradicionales del diseño

Eder (12) proporciona una breve discusión de la teoría de Sistemas Técnicos de Hubka (12) que ha estado en desarrollo desde mediados de la década de 1960, la cual se compara con la metodología de Pahl y Beitz (23). Mientras que los métodos de Pahl y Beitz están dirigidos principalmente a la ingeniería mecánica, Hubka (12) trabajó para desarrollar una teoría que es aplicable a todos los dispositivos de ingeniería, o "sistemas técnicos" (aunque típicamente es aplicada todavía a productos físicos). La teoría se basa en el concepto de un "sistema de transformación" general que incluye un proceso de transformación (propósito previsto) e interacción con cinco operadores: entorno, humanos, sistemas técnicos, sistemas de información y sistemas de gestión, con el objetivo de transformar un operando específico (material, energía, información, materia viva) en estructura, forma, ubicación o tiempo. Dentro de la teoría, un sistema técnico se mueve a través de un ciclo de vida compuesto de múltiples sistemas de transformación, donde los operadores o subsistemas también se consideran en la transformación de sistemas de una manera jerárquica. En comparación con Pahl y Beitz (23), la teoría y los métodos de Hubka (12) parecen más completos. Hubka (12) formaliza el ciclo de vida y también un ciclo de resolución de problemas. También se incorporan listas formales que clasifican requisitos, propiedades técnicas del sistema y propiedades del proceso de transformación. Para apoyar la teoría y métodos, numerosos ejemplos de casos también se han desarrollado. Sin embargo, los métodos de Pahl y Beitz han demostrado ser populares (especialmente en Alemania) debido a la adopción por parte de instituciones académicas y organizaciones profesionales, mientras que los métodos de Hubka (12) generalmente no se han presentado a los estudiantes o sean adoptados por la industria. Como resumen de las metodologías tradicionales más representativas del diseño se presentan en las figuras de la 14 a la 19.

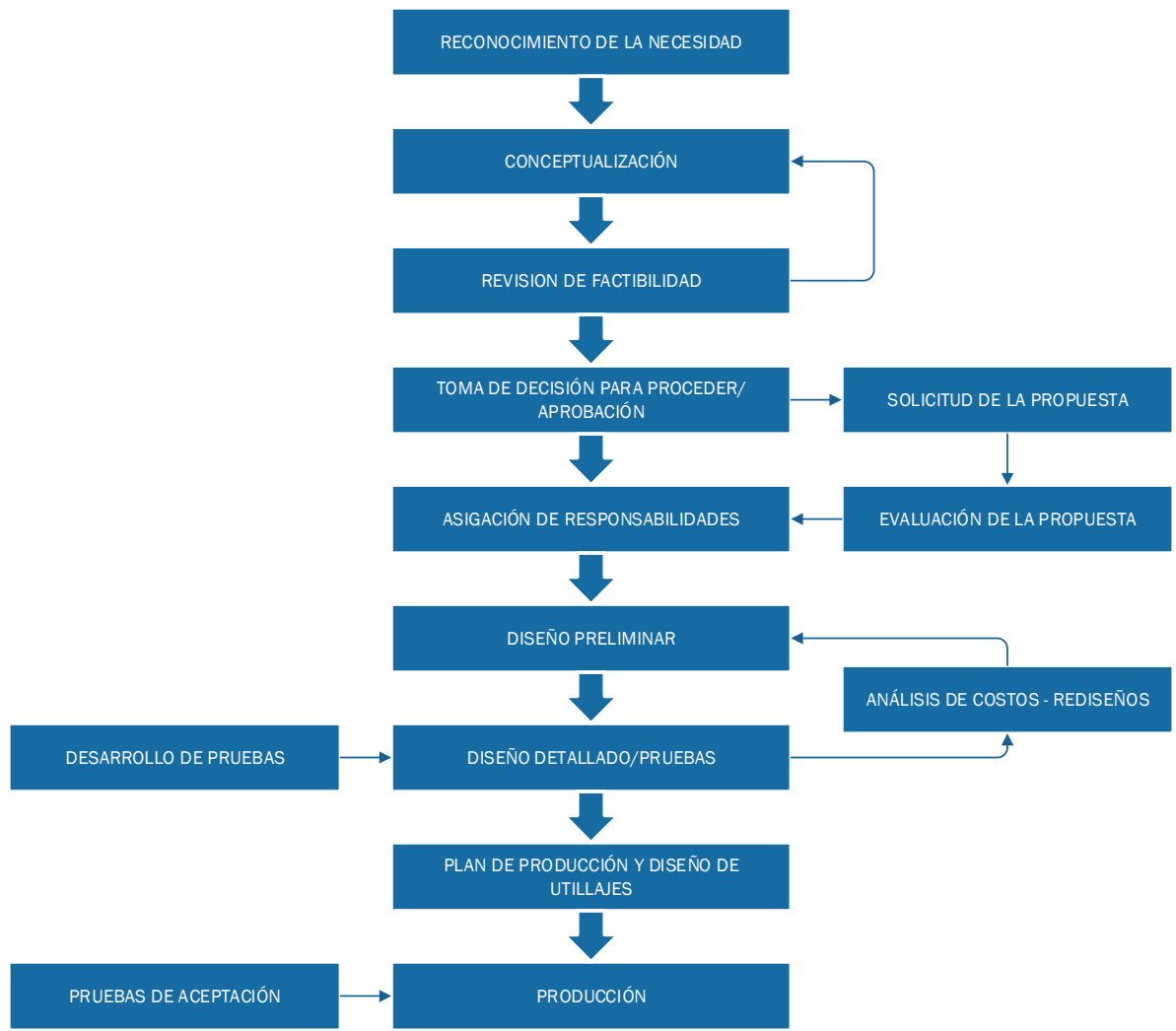


Figura 14. Proceso de diseño de ingeniería según Ertas y Jones (49).
 (Fuente: Modificación, Ertas y Jones (49))

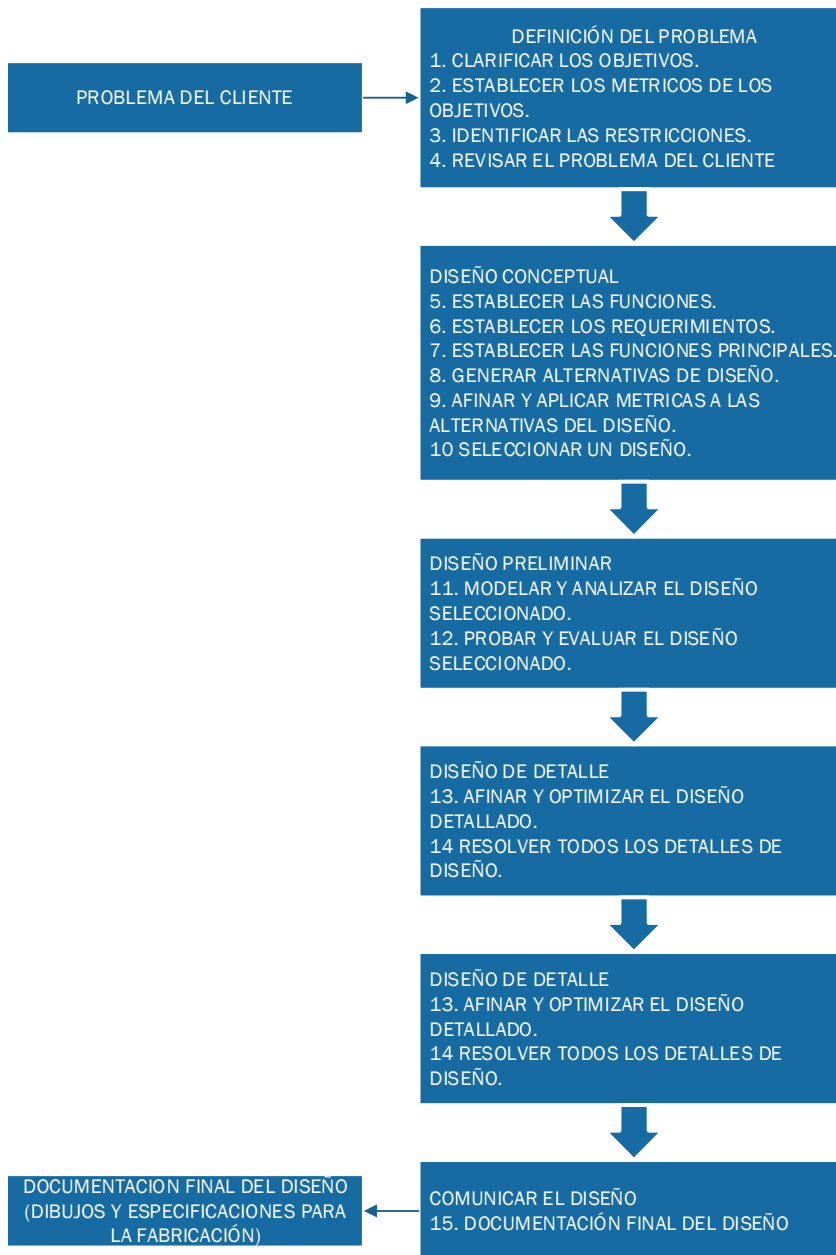


Figura 15. Modelo del proceso básico de diseño.
(Fuente: Dym & Little (47))

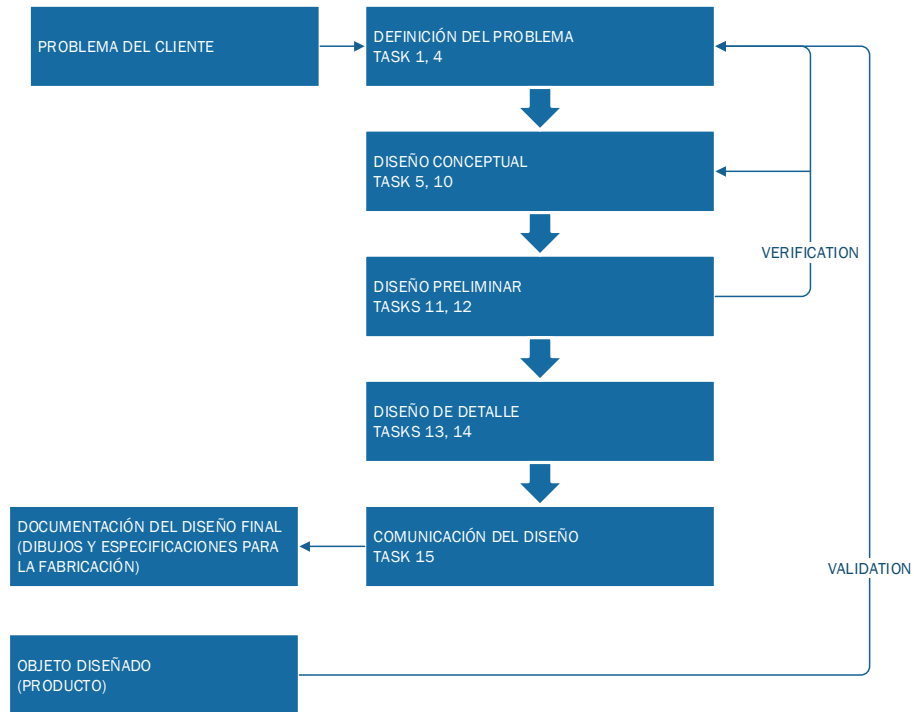


Figura 16. Modelo ajustado con retroalimentación básica.
(Fuente: Modificación Dym & Little (47))

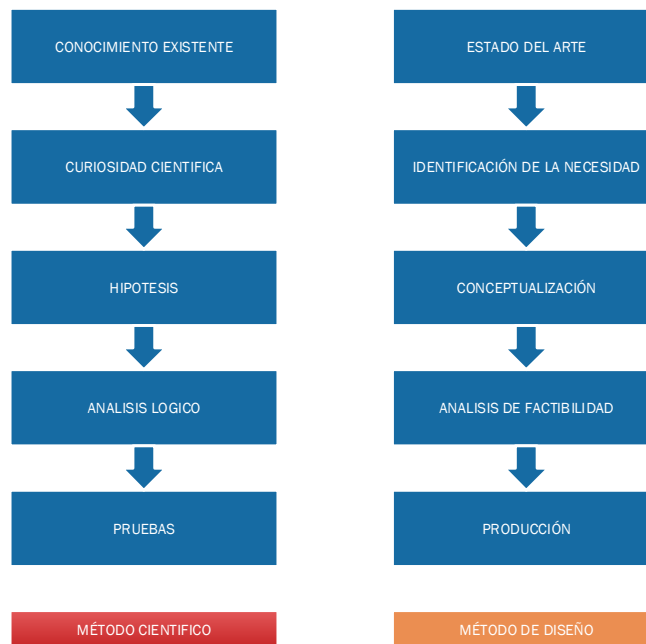


Figura 17. Comparación del método de diseño con el método científico.
(Fuente: Modificación Percy Hill (50))

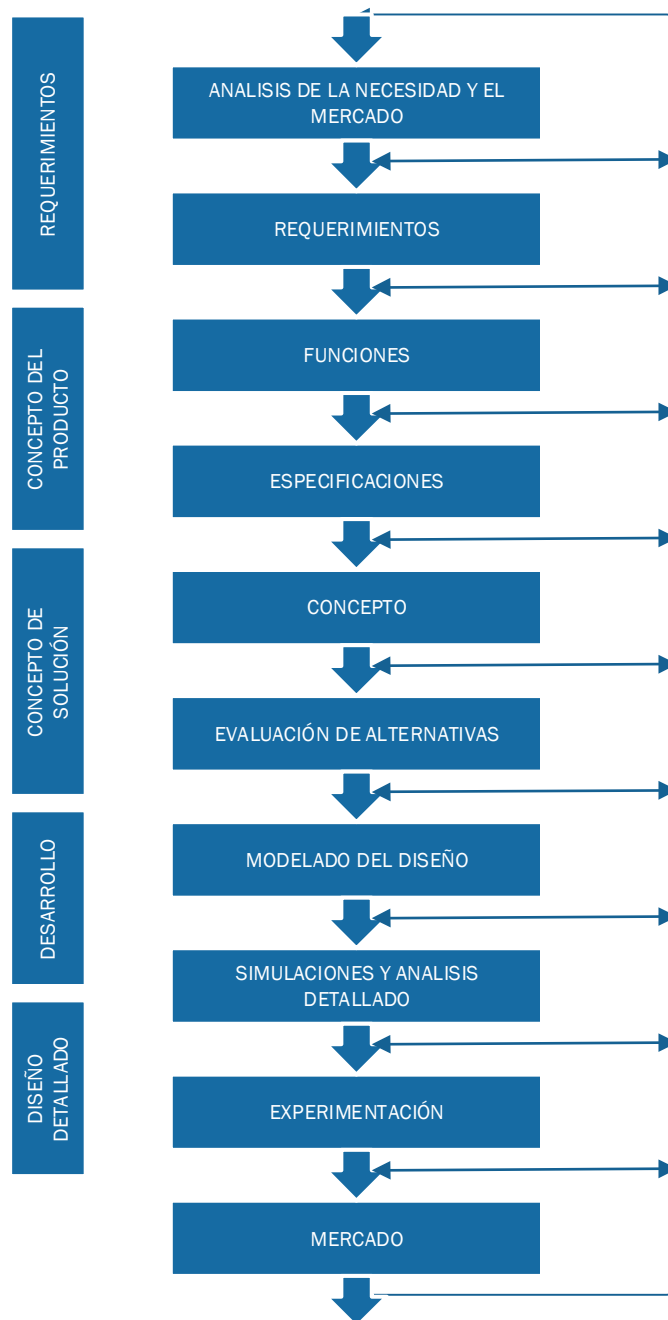


Figura 18. Pasos del proceso de diseño identificados por Haik y Shahin (51).

(Fuente: Modificación, Haik y Shahin (51))

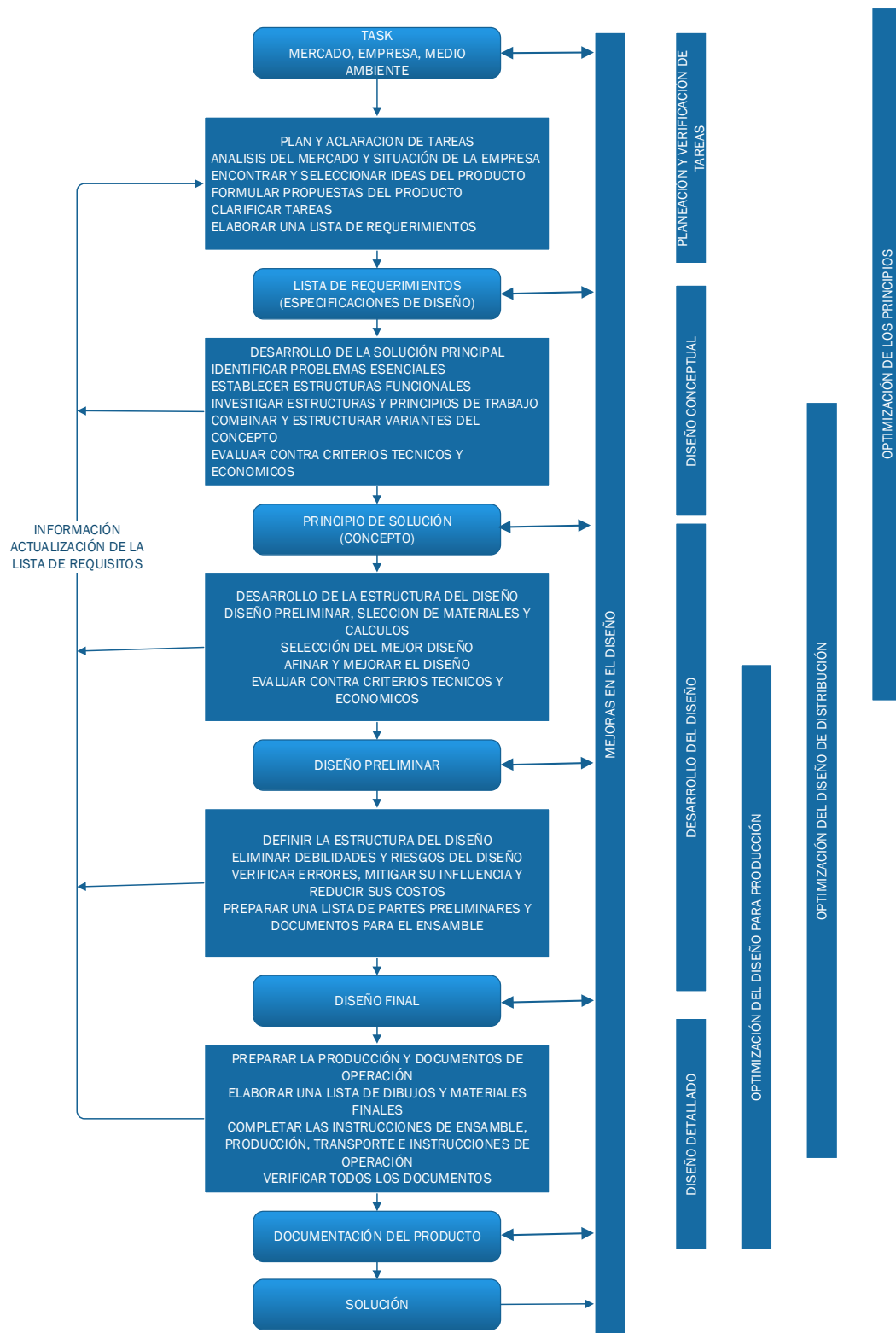


Figura 19. Pasos en el proceso de planificación y diseño.
(Fuente: Modificación Pahl & Beitz (23))

2.3 MÉTODOS ÁGILES

Los métodos de desarrollo ágiles han prevalecido en los proyectos de "software" en las últimas dos décadas como lo presenta Fitzgerald (52). Existe una amplia gama de metodologías que aprovechan los principios ágiles como una alternativa a enfoques tradicionalmente formales y planificados. La gran cantidad de metodologías con diferentes objetivos y con diferentes dominios, y tienen enfoques únicos en las técnicas de desarrollo y gestión de proyectos. Algunos ejemplos de los métodos ágiles incluyen "Scrum", Kanban, "Scrum"ban, XP, Desarrollo impulsado por características, Método de desarrollo de sistemas dinámicos, varios métodos Lean, Métodos cristalinos, desarrollo de "software" adaptable, varios procesos unificados, marco ágil a escala, "Scrum" a gran escala, entrega ágil disciplinada y gestión de proyectos extremos. Típicamente, los métodos ágiles se han dirigido a proyectos en los que el resultado es menos crítico y los requisitos son menos seguros. Sin embargo, empiezan a haber más ejemplos de adaptación de enfoques ágiles para su uso en proyectos críticos que demandan seguridad y calidad, según Fitzgerald (52).

2.3.1 Manifiesto Ágil

El movimiento ágil de desarrollo de "software" se solidificó en 2001 con el lanzamiento del manifiesto ágil (53), el cual es un conjunto de cuatro valores clave y doce principios asociados. Fue creado y apoyado por los principales desarrolladores ágiles de desarrollo de "software" de la época, incluido Jim Highsmith (54), autor de "Ecosistemas ágiles de desarrollo de "software", y Ken Schwaber, Jeff Sutherland y Mike Beedle, que fueron los principales desarrolladores de la metodología ágil de "Scrum". Los cuatro valores se citan aquí:

- "Individuos e interacciones sobre procesos y herramientas"
- ""software" funcional sobre documentación completa"
- "Colaboración con el cliente sobre la negociación del contrato"
- "Respuesta al cambio sobre seguir el plan"

El manifiesto establece que los artículos de la izquierda se consideran más importantes que los artículos de la derecha, según Kenneth (55) , incluso si este último todavía tiene valor en un proyecto. Estos valores se reflejan en los doce principios básicos, citados a continuación:

- "La más alta prioridad es satisfacer al cliente a través de la entrega temprana y continua de "software" valioso".
- "Bienvenidos, los cambios en los requisitos, incluso en fases avanzadas del desarrollo. Procesos ágiles aprovechan el cambio como ventaja competitiva".
- "Entregar "software" funcional con frecuencia, de un par de semanas a un par de meses, con una preferencia a la escala de tiempo más corta".
- "Los empresarios y los desarrolladores deben trabajar juntos a lo largo de todo el proyecto".
- "Desarrollar proyectos en torno a personas motivadas. Brindarles el medio ambiente y apóyalos, y confíe en ellos para hacer el trabajo".
- "El método más eficiente y efectivo para transmitir información hacia dentro de un desarrollo el equipo es una conversación cara a cara".
- "El "software" funcional es la medida principal del progreso".
- "Los procesos ágiles promueven el desarrollo sostenible. Los patrocinadores, desarrolladores y usuarios deberán ser capaces de mantener un ritmo constante indefinidamente".
- "La atención continua a la excelencia técnica y al buen diseño aumenta la agilidad".
- "La simplicidad, el arte de maximizar la cantidad de trabajo no realizado, es esencial".
- "Las mejores arquitecturas, requisitos y diseños surgen de equipos auto organizados".
- "A intervalos regulares, el equipo reflexiona sobre cómo ser más eficaz, luego sintoniza y ajusta su comportamiento en consecuencia". Los valores y principios influyen en el desarrollo de metodologías ágiles.

2.3.2 Ecosistemas de desarrollo de "software" ágil

Highsmith (54) describe los conceptos y filosofías de agilidad aplicadas al desarrollo de proyectos de "software". Esto es seguido por la introducción de las ideas e influencias de varios otros Firmantes del Manifiesto Ágil. Una breve reseña de las metodologías de desarrollo de "software" ágil que fueron populares en el momento de escribir, conduce a una descripción de cómo uno puede acercarse a la creación o adopción un ecosistema ágil de desarrollo de "software", apropiado para un proyecto y organización. Los autores utilizan el término ecosistema para referirse a la colección de cultura organizacional y

personas. Diversas metodologías y otros elementos que deben trabajar juntos en lugar de considerar una sola metodología para guiar todos los proyectos de diseño. Se señala que si bien el enfoque del trabajo es una aplicación para el "software", los principios ágiles subyacentes se aplican fácilmente a otros aspectos del negocio desarrollo y gestión de proyectos.

2.3.3 Características de los enfoques ágiles

El autor describe la agilidad como "la capacidad de crear y responder al cambio para obtener ganancias en un entorno de negocios turbulento". El movimiento de desarrollo de "software" ágil comenzó como una alternativa de enfoque en el desarrollo de "software" en la década de 1990, impulsado por el aparente fracaso de los métodos rigurosos del desarrollo de "software" en grandes proyectos de "software". Highsmith (54) afirma que los enfoques ágiles son caracterizados por un tratamiento diferente a la gestión de proyectos tradicional, así como al desarrollo de "software" en varias áreas. Primero, hay dos culturas claras o tipos de proyectos, los de *optimización* y los de *exploración*. La optimización se centra en la predicción con planes, programas y luego controlar o reducir desviación de estos, mientras que la exploración, usa planes y programas como una guía para manejar la incertidumbre y hacer concesiones en costo, tiempo y alcance. Los usuarios de métodos ágiles aceptan que los planes están "virtualmente siempre mal", y aun así los planes son utilizados como guías para la toma de decisiones y la colaboración. El aspecto de cultura indica las tendencias de los individuos, así como la organización, mientras que los proyectos también pueden clasificarse de manera similar, en función del nivel de incertidumbre o conocimiento y comprensión en todas las partes del proyecto. Un enfoque ágil es más a menudo adecuado a proyectos exploratorios o proyectos con altos niveles de incertidumbre. Los métodos tradicionales y ágiles deben equilibrar la estructura y la flexibilidad. Los métodos tradicionales tienden a favorecer la estructura rígida, mientras que los métodos ágiles favorecen la flexibilidad, dependiendo de la capacidad para responder a los cambios más que la capacidad de planificar con precisión. Parte de esto es equilibrar la disciplina de adherencia a las reglas con la comprensión de cuándo improvisar o modificar para adaptarse mejor a la situación del proyecto. La mentalidad ágil abarca el cambio y busca adaptarse rápidamente a las condiciones comerciales variables. Esta exige una toma de decisiones rápida, incluso si la decisión no es la mejor o la más óptima, e intenta alcanzar los objetivos con el mínimo esfuerzo. La retroalimentación obtenida de

estos ciclos más cortos ayudará a guiar futuras decisiones y actividades. Highsmith (54) enfatiza que "los proyectos ágiles no están controlados por conformidad con el plan, pero sí de conformidad con el valor comercial". La gestión tradicional se enfoca a los objetivos del proyecto y limitaciones de alto nivel para crear un plan, con un calendario específico y detallado, presupuesto y alcance. Las actividades de gestión intentan monitorear y controlar el proceso para cumplir con el plan, y el éxito se evalúa observando la desviación del plan, que puede ser poco realista o desactualizada. La gestión ágil de proyectos intenta ver el proyecto de manera más integral y evaluar el éxito de un proyecto por el valor comercial que ofrece. La atención se centra en los objetivos y las limitaciones de alto nivel. Mientras esto puede significar que se cumple un plazo específico en algunas circunstancias, el resultado es una opción más flexible en el marco para alcanzar el éxito y ofrecer valor a los clientes y a las partes interesadas. La gestión de proyectos tradicional utiliza un enfoque predictivo y hace hincapié en información más detallada en la planificación anticipada, mientras que los métodos ágiles usan un enfoque adaptativo. Una comparación puede ser figurada para controlar los sistemas. En un sistema físico bien entendido, un modelo detallado y preciso puede ser creado. Este modelo se puede usar como parte de un sistema de control efectivo que aprovecha el mínimo control, monitoreo y retroalimentación para lograr estabilidad y seguimiento del proyecto. Este enfoque predictivo se basa en un nivel de estabilidad y una mínima incertidumbre o ruido en el modelo, tanto en las entradas, salidas, así como en las métricas del proyecto. Al controlar un sistema con alta incertidumbre y ruido, la retroalimentación rápida usando una variedad de medidas es importante. Además, la estrategia de control puede ser necesaria para ajustar o adaptar en un amplio rango de operación con una visión adicional del comportamiento del sistema que puede ser recolectado. Highsmith (54) explica que los enfoques tradicionales tienden a enfatizar los procesos prescriptivos, describiendo en detalle el método de llevar a cabo tareas que contribuyen al éxito del proyecto. Los métodos ágiles parecen moverse un paso atrás y enfocarse en el equipo y las personas que componen el equipo, y cómo trabajan para hacer que el proyecto tenga éxito. Esto es similar al valor frente a la conformidad con el plan, donde el objetivo de varios procesos y políticas prescritos por la administración tradicional es ayudar al equipo o la organización a ser exitosa. Las metodologías ágiles simplemente reconocen esto y tratan de ponerlo en la vanguardia. Los métodos tradicionales ven a las personas como intercambiables, como partes del proceso general en una forma estandarizada. Los métodos ágiles intentan

aprovechar la varianza y las capacidades únicas de las personas en un equipo para ser más efectivo. Los miembros del equipo tienen habilidades, que son fácilmente transferibles a través de proyectos y tareas. Compare esto con las políticas o reglas de un proyecto u organización. Un conjunto de las políticas para comunicarse y coordinarse entre un equipo u organización se pueden considerar como la metodología.

2.3.4 Descripción de metodologías ágiles

Highsmith (54) describe una serie de metodologías ágiles. Éstas incluyen, "Scrum", Dynamic Systems Development Model, "Crystal Methods", Desarrollo basado en características, "Lean Development", Extreme Programming y Adaptive "software" Development. Cada una de estas metodologías se basa en los principios ágiles. La iteración es clave para aplicar comentarios en proyectos complejos e inciertos, considerando que los requisitos del cliente cambiarán a lo largo del proyecto, lo cual dificulta hacer planes a largo plazo. Comunicación constante dentro de un equipo de desarrollo, así como con la administración y los clientes, es importante para facilitar la comprensión y la retroalimentación. El enfoque debería estar en la entrega del sistema más simple que cumpla con los requisitos del cliente, y evitar anticiparse en características del producto que el cliente no espera y tal vez nunca pueda usar. Sin embargo, el diseño y el proceso debe ser fácil de cambiar para cumplir con las prioridades cambiantes de los clientes. Mientras que algunos casos es necesaria una planificación de alto nivel, el diseño debe revisarse y refinarse periódicamente en función de nuevos requerimientos. Todos estos principios buscan reducir continuamente la incertidumbre y el riesgo a medida que el proyecto progresa, lo que tiende a ser demasiado en muchos proyectos de desarrollo de "software". "Scrum" está muy enfocado en la administración de proyectos y tiene pocas características específicas del "software". Utiliza la iteración en el tiempo para monitorear el progreso del proyecto, se ajusta en función de los comentarios de las diferentes partes interesadas del proyecto, y proporcionan un entorno propicio para el éxito del equipo de trabajo. Esta interacción se compone de comunicación, colaboración, coordinación y conocimiento compartido. Una iteración en "Scrum" se llama "sprint" y puede durar de una a ocho semanas, cuando los períodos son más comunes. Una iteración tiene una etapa de planificación al comienzo, y una demostración o etapa de entrega y revisión al final. Una acumulación de productos describe todas las tareas o características que juntas componen un proyecto. En la terminología de "Scrum", estas son "historias de usuarios" o

simplemente "Historias" para reflejar que pueden estar escritas de una manera no técnica, comprensible por ambos, el cliente y el equipo de desarrollo. Requiere revisión regularmente, típicamente en el comienzo de la planificación de "sprints". Un retraso en la publicación es un subconjunto de la acumulación de productos que describe las prioridades para el próximo "sprint" y lanzamiento. La reunión de planificación de "sprints" en el comienzo de la iteración permite al propietario del producto (que representa los intereses del cliente) y al equipo de desarrollo para planificar cooperativamente lo que se completará durante este "sprint" basado en complejidad de tareas y recursos disponibles. Este conjunto de tareas se bloquea hasta el próximo "sprint". Durante el "sprint", el equipo de desarrollo extrae tareas de la acumulación de "sprints" y trabaja en conjunto para completar todas las tareas en la acumulación al final del "sprint". Parte de La fase de planificación implica identificar los criterios para que una tarea se considere completa, lo que generalmente significa que será demostrable al final del "sprint". Durante el "sprint", habrá un día reunión de ""Scrum"" (también conocida como stand up diario). Todos los miembros del equipo de desarrollo deben asistir a esta reunión corta (lo suficientemente breve), junto con el propietario del producto, administración y el "experto del "Scrum"". Los miembros del equipo deben describir lo que han logrado desde la última iteración, lo que pretenden lograr antes de la próxima iteración, y si hay algo impidiendo su progreso. El objetivo de un maestro de "Scrum" es trabajar con el equipo para seguir el proceso, identificar impedimentos (típicamente logísticos u organizacionales) y eliminarlos. Mientras el propietario de la administración puede obtener información de estado útil de la reunión de "Scrum", deben participar activamente en el interrogatorio o guiar al equipo de desarrollo. Al final de un "sprint", Highsmith (54) sugiere la duración de los "sprints" de 4 semanas, pero como "Scrum" ha evolucionado desde que se escribió esta fuente, los "sprints" son más comúnmente realizados en 1-2 semanas basadas en evidencia anecdótica. El equipo de desarrollo demuestra el progreso al propietario del producto y otras partes interesadas. Esta información se usa para revisar la cartera de pedidos del producto, planificar el próximo "sprint" y tomar decisiones de ir / no ir sobre si el proyecto continúa. Además, el equipo de desarrollo debe tener una reunión retrospectiva para reflexionar sobre el progreso realizado y cómo ajustar sus prácticas para tener más éxito o abordar las deficiencias. La metodología de "Scrum" se ilustra en la figura 20. Actualmente, "Scrum" es uno de los métodos ágiles más conocidos y usados en el desarrollo de "software".

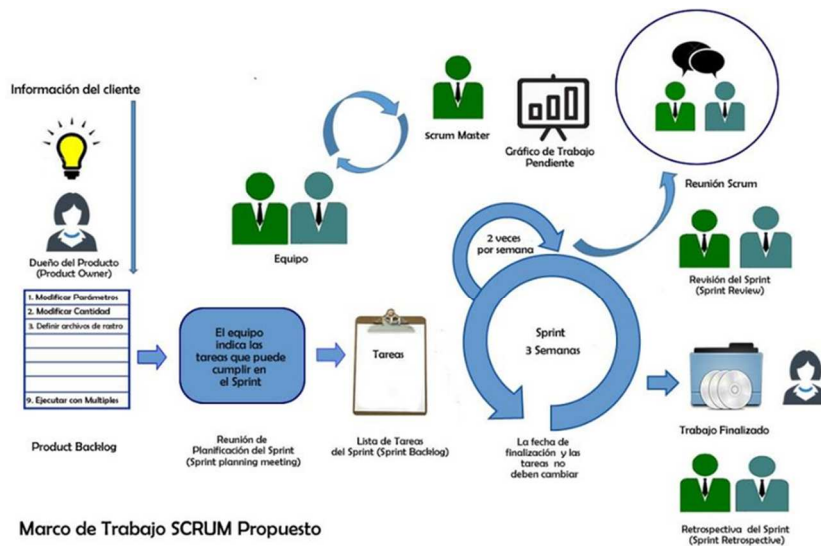


Figura 20. Método “Scrum” de gestión de proyectos.
(Fuente: Modificación Kenneth (55))

2.3.4.1 Métodos de Desarrollo Dinámico de Sistemas

El Método de Desarrollo Dinámico de Sistemas (DSDM) se basó en el desarrollo rápido de aplicaciones y proporciona un marco para el tiempo y la interacción del desarrollo. DSDM tiene mucha más complejidad y formalidad que la mayoría de las otras metodologías ágiles, pero puede ser más adecuado para la transición de una cultura más tradicional. Además, la formalidad se presta bien a la certificación de calidad como ISO 9001, y está respaldado por un consorcio de la industria, CONSORTIUM (56). DSDM tiene tres fases iterativas principales, que se muestran en la Figura 21, y numerosos principios, roles y productos de trabajo. Sin embargo, también está diseñada para adaptarse a las técnicas existentes en la organización. La primera fase es sobre el modelo funcional, que busca reunir y realizar un prototipo de los requisitos funcionales, así como especificar requisitos no funcionales, de manera iterativa. El diseño y la fase de construcción busca cumplir con todos los requisitos funcionales y no funcionales mediante ingeniería del “software” y refinando de los prototipos. Los cuadros de tiempo de estas fases pueden basarse en características, mediante el cual un pequeño conjunto de características pasa por todas las fases durante cada caja de tiempo, y es probable que se superpongan entre estas dos primeras fases. Finalmente, la fase de Implementación consiste en desplegar el sistema para el entorno del usuario. Hay un enfoque significativo en el uso de prototipos como parte

de lo que se entrega en cada etapa. Mientras que el autor advierte contra dejar un prototipo áspero convertirse en el producto final, DSDM ve los prototipos como un elemento básico que puede ser utilizado para demostrar la viabilidad (negocios, usabilidad, desempeño, capacidad) o como base para el sistema.

Los prototipos se utilizan para capturar información para la colaboración como una alternativa a la pesada documentación. Una idea subyacente que impulsa el enfoque en la iteración y los prototipos es la regla de 80-20, según la cual "nada se construye perfectamente la primera vez" y el último 20 por ciento del trabajo es pérdida de tiempo. Además, es difícil para el equipo y el cliente prever todos los requisitos en el principio del proyecto, y este cambio es respaldado por la iteración y la idea de que "el paso actual debe completarse solo lo suficiente para pasar al siguiente paso".

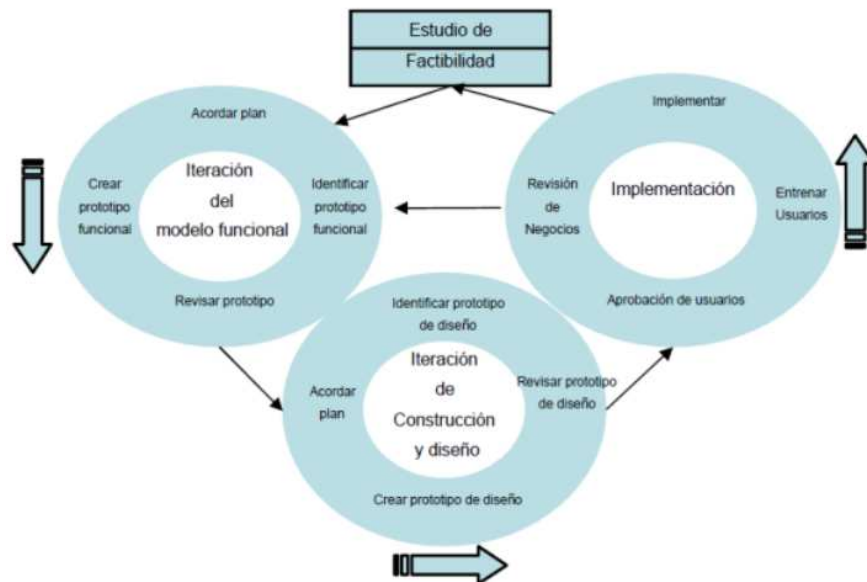


Figura 21. Método de desarrollo de sistemas dinámicos.

(Fuente: Modificación DSDM (56))

2.3.4.2 Metodología ágil Crystal

Desarrollado por Alistair Cockburn (57), el autor crea un marco de metodologías de diferente peso diseñadas para adaptarse a un equipo, proyecto y dominio específico ("Crystal Clear", Crystal Orange, etc.). Los tres factores principales que afectan el diseño de la metodología son las cargas de comunicación, la criticidad del sistema y las prioridades del proyecto. Los canales de comunicaciones crecen exponencialmente con

equipos más grandes, y esto requiere más prácticas y herramientas para complementar o reemplazar la interacción cara a cara. Proyectos donde la criticidad de la falla del "software" es alta (es decir, la vida amenazante) probablemente tendrá demandas regulatorias externas sobre los elementos metodológicos (tales como más políticas y menos tolerancia en las prácticas). Esto puede aumentar el peso de la metodología y agregar restricciones, pero no deberían afectar significativamente el diseño de la metodología. Tenga en cuenta que el peso de la metodología se define como el producto de tamaño (es decir, número de elementos por elemento) y su formalidad (por ejemplo, formalidad y detalles de los documentos).

Finalmente, las diferentes prioridades del proyecto impactan en el diseño de la metodología (por ejemplo, tiempo de salida al mercado, reducción de costos, exploración, responsabilidad legal). Los métodos Crystal proporcionan una matriz para ayudar a seleccionar y adaptar una metodología apropiada con cada uno de estos tres factores principales en un eje (comunicaciones prioridades de carga, criticidad y proyecto), representados en la Figura 22. "Crystal Methods" resalta que las personas, la interacción, la comunidad, las habilidades, los talentos y las comunicaciones tienen el impacto principal en el éxito del proyecto. El proceso se considera secundario y debe solo ser apenas suficiente. Sin embargo, debe adaptarse y evolucionar con el equipo específico (y sus talentos únicos), el proyecto y el dominio para ser efectivo. Considerando personas, interacciones, etc. reduce la necesidad de elementos de metodología pesadas y caras. "Crystal Methods" describe una serie de principios de metodologías de diseño. Mejorando los canales de comunicación entre personas para entregar un producto que funcione con mayor frecuencia y se reduzca el trabajo intermedio requerido. Esto se admite mediante el uso de ciclos incrementales, con un período máximo de cuatro meses. El equipo debe ajustar la metodología para que coincida con cada proyecto, especialmente a medida que evoluciona. Los elementos en una metodología se dividen en trece categorías: roles, habilidades, equipos, técnicas, actividades, procesos, hitos, productos de trabajo, estándares, herramientas, personas, calidad y valores de equipo. Todos estos deben ser considerados al crear y adaptar la metodología. Al seguir los objetivos de ligereza, agilidad y suficiencia absoluta, en cualquier metodología el elemento que no ayuda al equipo a trabajar de manera efectiva debe descartarse. Una metodología puede volverse compleja fácilmente ya que está "arreglada" con un número

creciente de elementos cada vez más complejos. Se observa que un mayor peso de la metodología puede ser apropiado en proyectos con "software" crítico. Cockburn describe su metodología "Crystal Clear" en el libro ""Crystal Clear": "Una metodología impulsada por los humanos para equipos pequeños", según Cockburn (57).



Figura 22. Metodología ágil cristal.
(Fuente: Modificación Cockburn (57))

2.3.4.3 Desarrollo impulsado por características

El desarrollo impulsado por características (FDD) es un conjunto de procesos simples con pautas para el desarrollo ágil basadas en iteraciones cortas de las características. Cinco procesos componen el "sistema" de desarrollo de "software" FDD.

Primero, un modelo general es desarrollado por los miembros principales del equipo para describir la arquitectura de alto nivel y proporcionar una base para diseño de características y modelado. Este modelo también comunica la comprensión sobre el diseño del sistema y proyecto de dominio para todos los miembros del equipo. Se sugiere

que esta fase tome alrededor del diez por ciento del tiempo total del proyecto, con algunas modificaciones adicionales durante todo el proyecto. A continuación, una lista de características pequeñas se desarrolla a través de una descomposición funcional ortogonal del modelo general. Las características deben ser una porción vertical del sistema que se puede implementar en dos semanas o menos y proporcionar valor para el negocio y para el cliente, quien debería participar en el proceso. Características que permanecen a alto nivel deben descomponerse en más pequeñas. Se sugiere que esta descomposición se restrinja al cuatro por ciento del tiempo del proyecto, con algunos trabajos en curso. Después de establecer la lista de características, el equipo debe planificar el programa de proyectos por función en múltiples niveles. Hitos y finalización de la actividad empresarial de alto nivel las fechas se basan en una escala de mes o año, con paquetes de actividades comerciales y características individuales, asignados a sub-equipos y desarrolladores individuales para ser administrados en incrementos de dos semanas.

La planificación debe considerar el orden de las funciones y las dependencias, el riesgo, la complejidad, el equilibrio entre la carga de trabajo y el cliente. Si bien esta planificación es importante, debe revisarse con frecuencia, con tiempo inicial (dos por ciento sugerido) que coincida con el esfuerzo de revisión en curso. Finalmente, el diseño y Las fases de compilación por características abordarán de forma iterativa la lista de características planificadas por sub-equipo para entregar funciones.

Los conjuntos relacionados de características tienen su diseño y modelos refinados para crear paquetes de diseño. Estos paquetes de diseño se implementan, inspeccionan, prueban e integran en el sistema general. Se espera que las fases de diseño y construcción usen más de las tres cuartas partes del tiempo total del proyecto. En comparación con otras metodologías ágiles, FDD muestra una mayor estructura tradicional a través de equipos jerárquicos dirigidos por miembros experimentados, participación del cliente explícita más limitada, y un énfasis en el modelado y la planificación al principio del proceso. Los cambios que surgen son aceptados a través del permiso de re trabajo de las fases de planificación, pero la gerencia debe garantizar que los cambios y el plan siguen siendo compatibles.

El modelado estructural que aborda la arquitectura y el dominio temprano está dirigido a reducir la repetición del diseño y también permite que las tareas se distribuyan a más sub-equipos independientes y potencialmente especializados. Esta estructura se escala más fácilmente a proyectos grandes y equipos con menos adaptaciones. Otro aspecto interesante del énfasis en el inicio y el modelado por características es que se recomiendan los modelos ejecutables, acercando los modelos a prototipos o código de trabajo que los documentos de diseño que quedarán obsoletos.

2.3.4.4 ““Lean Development””

““Lean Development”” considera que la agilidad y la tolerancia al cambio son ventajas competitivas en un mercado turbulento. Se basa en los temas de crear una cultura de arriba hacia abajo de agilidad, y ese cambio no solo se debe tolerar, sino que debe apalancarse en un entorno de alto riesgo en escenarios de "exploración" para generar ganancias. Esto se marca como pasar de la gestión de riesgos, que busca proteger contra la pérdida, para "arriesgar el emprendimiento".

La idea principal se basan en gran medida en aquellos procesos de producción pobre (o artesanales), que se centra en el uso de generalistas altamente calificados y simples y herramientas flexibles para crear lo que el cliente quiere en lugar del enfoque de producción en masa de usar trabajadores no calificados y equipos de un solo propósito para replicar un diseño creado por especialistas. Ahí son cuatro factores críticos de éxito que impulsan el Desarrollo Lean.

El objetivo principal de esta metodología es proporcionar valor para el cliente que se logra con trabajo en equipo. Satisfacer al cliente que exige una interacción cercana y colaboración con el cliente. El valor puede mejorarse aún más al crear tolerancia al cambio de "software". Esto incluye considerar cuidadosamente lo que puede cambiar y reducir el costo de hacer probables cambios en el futuro durante el diseño y desarrollo.

Extendiendo esto, una organización debería buscar crear soluciones de dominio que sean aplicables o adaptables en múltiples proyectos. El equipo también debe crear solo lo que sea necesario en un esfuerzo por entregar valor al cliente más con rapidez. Esto puede ser ayudado comprando en lugar de construir algunos componentes para mejorar la velocidad del desarrollo. Los clientes generalmente prefieren recibir el ochenta por ciento

ahora en vez del cien por ciento en el futuro, que puede ser difícil de lograr en mercados que cambian rápidamente. El minimalismo es otra clave para la agilidad, que es ayudado por la reducción de papeleo y la reducción del esfuerzo para comunicarse. Por ejemplo, la documentación a menudo agrega poco valor, pero utiliza recursos y tiempo significativos, aunque sea pequeña, los equipos cubricados pueden comunicarse fácilmente. Al igual que con otras metodologías ágiles, "Lean Development" es muy adecuado para dominios problemáticos específicos, y no será apropiado en todos los proyectos.

El entorno de "Lean Development" se centra en tres partes: la base técnica del "software" y arquitectura, las políticas y directrices para gestionar el esfuerzo, y los procesos, métodos y herramientas usadas. Desde el punto de vista del proceso, "Lean Development" generalmente se divide en tres fases.

La fase de inicio busca evaluar la viabilidad del proyecto (técnica y comercial) y comprender qué el cliente valora en un esfuerzo por reducir el riesgo. La fase de estado estacionario usa iteraciones encuadradas en el tiempo para un análisis completo, diseño, prueba, integración e implementación. Estos períodos de tiempo deben ser menos de noventa días y preferiblemente menos de sesenta días, con una evaluación cuidadosa entre iteraciones. La fase de estado estable finaliza con la entrega y aceptación del producto. Finalmente, en La fase de transición y renovación aborda la evolución continua del producto, así como las actividades de transferencia del conocimiento como documentación y capacitación.

2.3.4.5 "Extreme Programming"

"Extreme Programming" (XP) esboza un conjunto concreto de prácticas dirigidas en pequeños equipos de desarrollo de "software", respaldados por valores y principios fundamentales. Ser exitoso, la cultura de la organización debe estar alineada con los cinco valores y principios clave: comunicación, simplicidad, retroalimentación, valor y trabajo de calidad. XP busca crear un ambiente adecuado para la creatividad y la comunicación a través del uso disciplinado de estos recursos complementarios. Las prácticas están diseñadas para fomentar la interacción entre los desarrolladores y con el cliente, priorizando la comunicación cara a cara sobre la documentación para mejorar la comprensión del dominio del problema. El diseño debe mantenerse simple pero tolerante

al cambio, continuo y se desea una respuesta rápida. Apoye a la idea de que el equipo y sus miembros deberían esforzarse por hacer lo que es correcto y seguir siendo disciplinado incluso bajo presiones competitivas (como los plazos). La calidad del trabajo es el producto de la actitud y cultura fundamentales del equipo, combinado con una definición de calidad, ya sea basada en el valor o cuantificable (como ningún defecto). Programación extrema describe doce prácticas complementarias que trabajan juntas para hacer un proyecto exitoso, resumido de la siguiente manera:

- El juego de planificación: XP se dirige a iteraciones de tres semanas, usando historias como descripciones de características creadas a través de la colaboración entre el cliente y el equipo de desarrollo, para delinear un plan de lanzamiento que proporciona una indicación de alto nivel del alcance, el costo y el cronograma general. Clientes impulsan el valor al priorizar las historias, y el plan general debe actualizarse continuamente".
- "Pequeñas Versiones: el "software" debe lanzarse con frecuencia en pequeños incrementos que contengan requisitos comerciales más valiosos. Cuando una versión completa es costosa debido a la instalación, capacitación y migraciones, considerar que el proyecto puede ser liberado como alternativa razonable".
- "Metáfora: se usa una descripción o visión de alto nivel para mantener el proyecto y sus características en rastrear y mejorar la comprensión general para desarrolladores y clientes".
- "Diseño simple: el diseño del sistema debe ser el mejor y más simple para cumplir solo con la funcionalidad, limitando el diseño anticipatorio tanto como sea posible".
- "Refactorización: XP fomenta la refactorización continua del código existente a medida que se agregan características o modificaciones para garantizar que el diseño siga siendo simple y robusto, y para reducir la degradación de código existente (por ejemplo, a través de soluciones rápidas)".
- "Pruebas: el uso extenso y frecuente de pruebas unitarias y funcionales ayuda a reducir defectos, especialmente con refactorización. El desarrollo basado en pruebas (TDD) se adapta bien al desarrollo de XP. Las pruebas deben ser tan automatizadas como sea posible".
- "Programación de pares: esta es una práctica de programación intensa en la que dos desarrolladores trabajan juntos en la misma terminal y en el mismo código, con

el objetivo de codificación simultánea, código inspección y aprendizaje o transferencia de conocimiento. Es similar a un código y luego a un ciclo de inspección, maximizando la frecuencia de retroalimentación".

- "Propiedad colectiva: un complemento para la programación de pares, la idea es que todo el código en el proyecto pertenece al equipo en lugar de a un individuo, y que cualquier desarrollador puede modificar y mejorar cualquiera de los códigos, en lugar de asignar partes específicas del código a las personas. El objetivo también es evitar la culpa, reducir el ego y fomentar el aprendizaje a través de mayor (pero apropiada) toma de riesgos".
- "Integración continua: junto con las prácticas de prueba, el aumento de la frecuencia de construir y probar todo el código en el proyecto ayuda a obtener comentarios y encontrar defectos (especialmente problemas relacionados con la integración) tan pronto como sea posible".
- "Semana de 40 horas: productividad y calidad sostenibles es posible si los desarrolladores son entusiastas y comprometidos por su cuenta, en lugar de ser empujados al límite y obligados a trabajar (más de una semana de) horas extras por horarios poco realistas.
- "Cliente en el sitio: interacción continua y colaboración entre el cliente y el equipo de desarrollo debe ser fácil, lo que se logra teniendo al cliente en el sitio para permitir comunicación cara a cara y retroalimentación rápida".
- "Estándares de codificación: la naturaleza intensamente colaborativa de la programación de pares y el colectivo de propiedad requiere que el equipo use estándares de codificación para la coherencia".

La programación extrema se centra en la colaboración, el minimalismo, la simplicidad, y se debe tener cuidado al eliminar, agregar o sustituir este conjunto complementario de prácticas si la metodología permanece efectiva.

2.3.4.6 "Adaptive "software" Development"

"Adaptive "software" Development" (ASD) fue desarrollado por Highsmith (54). Toma influencia de la teoría de sistemas adaptativos complejos. Está enfocado en el cambio y adaptación continua utilizando la base de prácticas y personas ágiles. ASD tiene muchas

similitudes a otras metodologías ágiles, y considera usar el producto de la mejor manera para aprender cómo debe evolucionar, con ciclos rápidos de desarrollo iterativo, mejorando los requisitos y el diseño "suficientemente buenos".

La metodología sugiere un ciclo de vida dinámico e iterativo de especular, colaborar, aprender en proyectos de alta incertidumbre y cambios, en lugar del plan lineal, estático, como el ciclo de compilación que generalmente se encuentra en un enfoque de cascada. Especular alienta la exploración de problemas complejos en el entendimiento de que la incertidumbre exige que los planes se actualicen. Colaborar alude al hecho de que el intercambio de información en proyectos complejos requiere un esfuerzo de equipo. Aprendizaje, enfatiza que el conocimiento nunca es completo y debe ser aumentado con comentarios regulares de los clientes y del equipo en sí. Este ciclo de vida tiene seis características, siendo enfocado en la misión, basado en características, iteraciones, periodos de tiempos, administración de riesgos tolerantes al cambio. Un enfoque en la misión sobre los requisitos detallados ayuda a guiar la dirección del proyecto y las decisiones de compensación, incluso cuando los requisitos son confusos y es probable que cambien.

ASD prioriza las funciones orientadas al cliente como entregables sobre la finalización general de las tareas. Reparar la entrega de proyectos e iteraciones por las interacciones en periodos de tiempos son importantes para enfocar y forzar compensaciones, y no deberían causar horas extra. Las iteraciones deben ser adaptativas y planificadas en parte a través del análisis y gestión del riesgo en el proyecto. El ciclo de vida ASD se ilustra en la figura 23. Tenga en cuenta que hay algunos aspectos explícitos previos y posteriores a las fases de especulación y aprendizaje. La iniciación del proyecto se usa para establecer la misión y los objetivos, comprender las limitaciones, establecer la organización del proyecto, identificar y describir los requisitos, hacer estimaciones iniciales de tamaño y alcance, e identificar riesgos clave del proyecto. Se recomienda el uso de sesiones de Diseño de Aplicación Conjunta (JAD) para ayudar a reunir requisitos. Es importante proporcionar una estimación inicial de la programación (cuadro de tiempo total del proyecto), así como también la cantidad y el tamaño de las iteraciones individuales. Se observa que estas tareas deben revisarse regularmente en cada fase de planificación del ciclo adaptativo y revisado según corresponda. Esta segunda fase es dominada por el desarrollo de un tema

u objetivo general para la próxima iteración, la asignación de un conjunto de características que se pueden entregar dentro de la caja de tiempo.

Durante la asignación de funciones, el equipo trabaja para evaluar el esfuerzo, riesgos y las dependencias, mientras que el cliente debe decidir la prioridad. Se puede completar un plan preliminar y aproximado de algunas o todas las iteraciones, no solo en la actual iteración, para mejorar continuamente el plan.

El desarrollo concurrente de características permite que el equipo técnico pueda entregar las características de una manera colaborativa y concurrente, con gerentes que faciliten comunicación e interacción rápidas. Las herramientas más allá de la discusión cara a cara pueden ser necesarias ya que un equipo crece o se distribuye. Esta interacción puede incluir otras partes interesadas y cubrir no solo desafíos técnicos, sino también requisitos comerciales y toma de decisiones. La revisión alienta el aprendizaje desde varias fuentes: la visión del cliente de la calidad, la calidad desde un punto de vista técnico, la efectividad del equipo y las prácticas en uso, así como el estado del proyecto en relación con el plan.

Grupos de enfoque a clientes se sugieren para obtener retroalimentación estructurada de los clientes. La revisión regularmente de la arquitectura y las revisiones continuas del código proporcionan información sobre la calidad técnica. El autor sugiere que los procesos cascada hacen que el aprendizaje y la retroalimentación sean más difíciles, como la filosofía "hazlo bien a la primera", la cual desalienta la experimentación y el retroceso. Además, los miembros del equipo pueden beneficiarse al compartir trabajo en progreso, para ayudar a encontrar errores o pequeños problemas que puedan corregirse antes de que se vuelvan más grandes.



Figura 23. Fases del ciclo de vida de desarrollo de "software" adaptativo.
(Fuente: Elaboración propia)

En apoyo de Adaptive "software" Development, Highsmith (54) describe específicamente la importancia de un cambio en el estilo de gestión. La gestión jerárquica tradicional de comando y control se enfoca en la optimización para lograr un proceso más eficiente y predecible a través del rigor y el control. Él sugirió que los proyectos ágiles se benefician enormemente de un estilo de gestión de Liderazgo-Colaboración, con un enfoque en la adaptabilidad sobre la optimización. Tal enfoque busca equilibrar el rigor y estabilidad con flexibilidad e improvisación, descrita como "apenas suficiente". Requiere una cultura de cambio, por el cual los gerentes son líderes que proporcionan dirección y guía, facilitan la conexión de personas y equipos, así como ayudar a crear entornos para la innovación, la creatividad y la toma de decisiones efectiva. Se sugiere que prácticas como la gestión de riesgos y el control de la configuración ayuden a respaldar toma de decisiones en colaboración.

Establecer un ecosistema de desarrollo ágil de "software" con base a los esfuerzos previos para comprender los valores y principios ágiles. Highsmith (54), describe los pasos para crear un ecosistema de desarrollo de "software" ágil y describe los elementos que deben considerarse al desarrollar uno.

El primer paso es comprender la oportunidad y el dominio del problema, la cultura organizacional, y utilizarlos como base para seleccionar o diseñar una metodología. Un proceso ágil o enfoques rigurosos pueden ser apropiados dependiendo del dominio y la cultura, y el autor propone metodologías en tres amplias categorías:

Adhoc, ágil o riguroso. Highsmith (54) sugiere que el dominio y la cultura se evalúen de la manera sugerida por Geoffrey Moore (58). Moore describe cuatro tipos de mercado diferentes: "Early Market, Bowling Alley, Tornado y Mainstreet".

Los mercados de "Mainstreet" son tradicionales y estables, mientras que los otros generalmente trabajan para traer un proyecto a esta zona estable. Los proyectos en los diferentes tipos de mercado se caracterizan por diferencias en objetivos, incertidumbre, perfiles de riesgo, criticidad y restricciones. Del mismo modo, hay cuatro diferentes culturas organizacionales: cultura, competencia, colaboración y control. El núcleo subyacente de los valores asociados con la cultura de una organización debe coincidir con la metodología y el dominio del proyecto. Comprender la cultura es importante, ya que tiene un impacto significativo en la toma de las decisiones comerciales. La cultura puede variar

a través de una organización o etapas de un proyecto. Los primeros mercados se caracterizan por la investigación y el desarrollo temprano, que es poco probable que se beneficien de cualquier metodología significativa (una metodología adhoc o "light agile" pueden ser apropiadas).

"Bowling Alley" y los mercados de tornados pueden ser volátiles y demandar una metodología adecuada para cambios rápidos, así como apoyar una estrecha colaboración con clientes para obtener retroalimentación y adaptarse rápidamente, lo cual encaja bien con metodologías ágiles. La cultura se centra en los individuos y tiende a organizarse a sí misma, aunque puede ser difícil de escalar debido a una resistencia contra la estructura y el proceso, por lo que es más adecuado para una metodología adhoc o tal vez una metodología lean o ágil. Una cultura de competencia se centra en las habilidades y la excelencia técnica para conducir un equipo de alto rendimiento que puede no tener mucha tolerancia para el proceso, encajando bien con métodos ágiles.

Las culturas de colaboración a menudo se basan en equipos inter funcionales que se benefician de liderazgo fuerte e interacción mejorada con capacidad limitada para aceptar el rigor, en este caso, metodologías ágiles se acercan a una buena opción. Una cultura de control enfatiza la predictibilidad y la planificación, en lo cual es mejor acomodar metodologías rigurosas. En los casos donde la cultura y la oportunidad no están bien combinadas (por ejemplo, Mainstreet), la única solución puede ser dividir la organización de tal manera que más de una cultura puede desarrollarse independientemente (por ejemplo, una empresa derivada o una nueva división).

Finalmente, Highsmith (54) describe las tareas que guían para la selección de una metodología.

Primero, la cultura del grupo debe articularse en términos de valores y principios, un ejercicio que en sí mismo puede ser difícil e introspectivo. También puede ser de interés el evaluar la compatibilidad de esta cultura en comparación con otros grupos y la organización en general. Siguiendo, los dominios problemáticos, proyectos u oportunidades de negocios que comprenderán el trabajo del grupo, deben ser identificados y caracterizados. Si la cultura y el dominio no parecen compatibles, puede ser necesaria una nueva estructura organizacional o un cambio cultural.

Basado en la cultura predominante, se debe seleccionar una metodología básica, que con suerte será aplicable a algunos proyectos. Finalmente, para cada proyecto que el grupo aborda, la aplicabilidad de la cultura y la base metodológica o ecosistema debe ser evaluado continuamente. La metodología básica puede luego adaptarse al proyecto específico. Es importante que el ecosistema de desarrollo se adapte primero a la cultura, y el proyecto en segundo lugar, ya que es probable que sea más exitoso que tratar de cambiar la cultura utilizando una metodología para un proyecto específico.

Es cierto que las características del proyecto variarán, y puede ser beneficioso tener una cartera de ecosistemas que se adaptan tanto a diferentes culturas de equipo como a diferentes proyectos.

2.3.5 Diseño de una metodología ágil

Highsmith (54) proporciona una base detallada para diseñar una metodología ágil. Ante todo, el autor reitera que una metodología efectiva debe adaptarse a influencias, así como el equipo que lo usa. La metodología debe equilibrar la flexibilidad deseada para ayudar a los proyectos exploratorios a sobresalir con la estabilidad y la previsibilidad necesarias para un negocio efectivo. Si bien una sola metodología no será efectiva para todos los proyectos, una metodología cuidadosa fomentará la reutilización y la adaptación eficientemente. Al manejar la variación entre proyectos, una metodología bien diseñada proporcionará consistencia y mejorará la previsibilidad, incluso si el proceso es no repetible y el proyecto tiene una incertidumbre considerable.

2.3.5.1 Comprender partes de una metodología

Una metodología se compone de elementos. Elementos son el "qué" de la metodología, descrito como elementos, componentes o rasgos del equipo y la metodología propia, la cual puede ser escrita, descrita, aislada, etc. El concepto de elementos puede extenderse para describir el ambiente del equipo que también compone el ecosistema (el ecosistema abarca todo, potencialmente incluye múltiples metodologías y la cultura). Las metodologías se caracterizan por el tamaño (la cantidad de elementos), ceremonia (detalle y formalidad de los elementos), peso (el producto de tamaño y ceremonia) y la tolerancia (qué tan cerca debe estar un equipo de la metodología, la capacidad de variación y flexibilidad).

Las prácticas son el "cómo": describen los pasos en un proceso o comportamientos del equipo o actividades llevadas a cabo, para usar o crear elementos. Estas prácticas se dividen en tres categorías.

2.3.5.1.1 Colaboración

Las prácticas describen la interacción entre las personas dentro de un equipo, así como las de afuera. Las prácticas de gestión de proyectos describen cómo se gestiona y supervisa el proyecto, describiendo la visión y ayudando a atar el trabajo. Las prácticas de desarrollo de "software" guían cómo es el "software" diseñado, construido y probado.

Algunas prácticas pueden pertenecer a más de una categoría y un conjunto de prácticas en una metodología, deben complementarse entre sí como un sistema. Diferentes metodologías ágiles pueden centrarse más en algunas categorías que otras, dejando una mayor flexibilidad o adaptabilidad en algunas áreas. Por ejemplo, mientras "Scrum" y XP incorporan prácticas de colaboración, "Scrum" tiende a centrarse más en las prácticas de gestión de proyectos, mientras XP prioriza las practicas del desarrollo de "software".

Los principios y valores subyacentes de la organización deben guiar el desarrollo de prácticas. Las prácticas pueden ajustarse fácilmente para adaptarse a diferentes culturas, y la implementación de las prácticas siempre debe considerar útil la intención de los principios y la cultura subyacentes. El autor promueve la simplicidad en una metodología, siguiendo temas mínimos del proceso y en su lugar confiando en la colaboración del equipo, respaldado por un entorno creativo para generar valor. "Cuando una metodología se convierte en algo para eludir a fin de obtener el trabajo hecho, impide el progreso en lugar de ayudarlo". Se sugiere que hay seis áreas clave donde se necesitan prácticas para crear una metodología ágil efectiva:

- La elaboración de la misión ayuda a definir una visión que respalda los objetivos comerciales para dirigir el proyecto y mantener el foco.
- Iniciación del proyecto busca crear una estimación inicial breve para el alcance, el costo y el cronograma. utilizado como una guía por el equipo y el cliente.
- Los ciclos iterativos fomentan la retroalimentación regular y deben ser impulsados por las características para proporcionar al cliente valor en periodos cortos de tiempo.

- Comentarios constantes permite la comparación con la visión y la evolución de los planes ayudando a mantener el proyecto enfocado
- Relación con el cliente, fomenta la estrecha colaboración entre el equipo de desarrollo y el cliente para comprender los valores comerciales y priorizar en consecuencia.
- La excelencia técnica y un producto de calidad, proporcionarán una base para crear valor en el presente y el futuro.

Highsmith (54) se centra específicamente en proyectos de "software", pero esta categoría aísla las prácticas específicas de disciplina y ayuda a modularizar una metodología haciendo que las prácticas de colaboración y gestión de proyectos sean más generales en el campo interdisciplinario de proyectos.

Una nota final sobre la tolerancia de la metodología, es que puede cuantificarse por el número de políticas. Un mayor número de políticas generalmente indica una tolerancia menor. Las políticas son parte de una metodología, las cuales son mandatorias por la organización.

2.3.5.2 Principios de diseño metodológico

Highsmith (54) proporciona nueve principios para guiar el diseño de la metodología, citados a continuación:

- "Principio 1: la comunicación cara a cara interactiva es el canal más económico y rápido para el intercambio de la información"
- "Principio 2: los equipos más grandes necesitan una metodología más pesada"
- "Principio 3: el exceso de peso de la metodología es costoso"
- "Principio 4: la ceremonia mayor es apropiada para proyectos con mayor criticidad"
- "Principio 5: la formalidad no es disciplina, el proceso no es habilidad y la documentación no es comprensión"
- "Principio 6: aumentar los comentarios y la comunicación reduce la necesidad de entregables intermedios"
- "Principio 7: la eficiencia es prescindible en actividades sin cuello de botella"
- "Principio 8: pensar en flujo, no en lote"

- "Principio 9: Se puede requerir una mayor aplicación en la metodología para cumplir con los requisitos reglamentarios, de seguridad, y consideraciones legales"

Los equipos más grandes o los equipos distribuidos pueden comprometer el principio 1 y también pueden afectar el principio 6, lo que requiere una metodología más pesada, especialmente con respecto a las prácticas de colaboración. Principio 3 impulsa los objetivos de la simplicidad y la suficiencia absoluta. Los principios 4 y 9 son muy similares y abordan gestionar el riesgo y cumplir con los requisitos impuestos externamente. El Principio 5 sugiere que el anterior enfatice a métodos rigurosos y apoye a este último, que es el enfoque de los métodos ágiles (por ejemplo, la formalidad apoya la disciplina). El Principio 7 alienta a los miembros del equipo a concentrar sus esfuerzos en lo más necesario, mientras que el Principio 8 busca agilizar las actividades y procesos para reducir cualquier retraso cuando sea posible.

2.3.5.2.1 Marcos, plantillas y escenarios

En un esfuerzo por dejar atrás una metodología lineal y cambiar el enfoque de los elementos prescriptivos a las pautas, Highsmith (54) sugiere un ciclo de vida de fase y revisiones compatibles con plantillas de dominios de problemas y escenarios de desarrollo. Estos deberían ser personalizables para que coincida con la organización, las unidades dentro de la organización y los equipos de proyectos.

El uso de un marco de ciclo de vida de fase y las revisiones ayudan a delinear el flujo de alto nivel e importantes hitos. El autor enfatiza que la representación visual de un ciclo de vida iterativo puede ser difícil y parecen lineales. Los hitos brindan oportunidades importantes para un monitoreo de alto nivel y toma de decisiones, a la vez que permite sincronizar proyectos para la integración de múltiples subsistemas. Algunos ejemplos de hitos clave incluyen la aprobación del proyecto, el lanzamiento del proyecto, la característica iterativa de las terminaciones y el lanzamiento del producto. El autor afirma que si bien los criterios de decisión ciertamente cambian, la mayoría de los proyectos encajan en un conjunto básico de las siguientes fases:

- Gestión del producto
- Viabilidad del proyecto

- Planificación
- Desarrollo iterativo de productos
- Estabilización e implementación

Las plantillas de dominio de problemas se crean combinando elementos de la metodología dirigidos a un entorno específico de dominio del problema, abordando la variación de los dominios problemáticos que una organización puede enfrentar. Los equipos del proyecto pueden seleccionar una plantilla apropiada para el proyecto en cuestión, y deben personalizar la plantilla para ser más efectivo. Esto se fomenta con una descripción ligera en la práctica (es decir, el autor sugiere una o dos páginas). Es importante garantizar el dominio del problema al que se aplica la plantilla y debe estar claramente identificada.

La serie de metodologías Crystal mencionada anteriormente se puede comparar a un conjunto de plantillas de dominio de problemas. Los escenarios se esfuerzan por describir cómo las personas trabajan juntas como un equipo para producir resultados con la narrativa en lugar de prescribir una lista secuencial de tareas para individuos. Ellos enfatizan la interacción y colaboración tanto dentro del escenario como entre escenarios y dentro de la plantilla o fase (especialmente de una manera no lineal).

Los escenarios pueden compartir información de entrada de fase o contribuir juntos para generar salidas de fase. Entre los escenarios, la interacción puede reflejar dependencias, iteración o cooperación. Los proyectos pequeños se pueden describir en un puñado de escenarios mientras proyectos grandes requerirán más. Los escenarios se centran en la interacción, pero pueden describir actividades y hacer referencia a otros elementos de una metodología, como técnicas, roles, patrones y productos de trabajo. Eso puede ser posible para reutilizar escenarios individuales en diferentes plantillas y marcos de ciclo de vida.

La metodología aplicada a proyectos específicos se adapta fácilmente sustituyendo o modificando escenarios para acomodar características únicas del proyecto. Pasos de diseño de metodología de colaboración. Como se mencionó anteriormente, los primeros pasos en el diseño, una metodología incluye evaluar la cultura del equipo de la organización y del proyecto, los valores y principios, y luego comprender el dominio del problema. Objetivos y características del proyecto también debe ser determinado. Los

objetivos y las características se pueden evaluar en un sentido general con consideración de los proyectos típicos asumidos por la organización o el equipo, y en un proyecto base al personalizar la metodología para su uso en un proyecto específico. Solo entonces puede un ciclo de vida de marco, plantillas y escenarios se desarrollan con éxito. Al considerar los objetivos del proyecto, se deben considerar el valor comercial y las prioridades de los compromisos.

Comprender el valor del negocio así cómo el proyecto se puede ajustar la estrategia a largo plazo articulando la prioridad y el equilibrio del costo del proyecto, el cronograma y el alcance, lo que impulsará decisiones de gestión de proyectos potencialmente difíciles a lo largo del proyecto. Características del proyecto deben entenderse para aplicar correctamente una metodología que incluya el tamaño del equipo, la criticidad, el riesgo e incertidumbre y alcance de la actividad. El alcance de la actividad describe qué partes de una vida de desarrollo será abordado por el proyecto. Por ejemplo, un proyecto en su inicio de investigación y desarrollo es improbable que incluya actividades de estabilización o despliegue de productos. Finalmente, la metodología puede ser descrita. Definir explícitamente los principios proporciona una base para entender las otras partes de la metodología. También conduce a una breve descripción de la intención de la metodología, y una breve descripción de su alcance. El ciclo de vida de fase y revisiones debe ser descrito, después de lo cual se pueden identificar plantillas de dominio específicas. Una parte clave de la plantilla de dominio es el diseño de los escenarios que componen la plantilla (junto con otros elementos). Entonces, se puede presentar una descripción de cada elemento en las plantillas de metodología. El autor sugiere limitando la descripción de la metodología a un máximo de 25 - 50 páginas, para mantener la simplicidad.

Al crear la metodología, es importante evitar caer en una descripción de metodología tradicional: centrarse en cómo trabajar juntos y colaborar, lo que puede hacer referencia a las habilidades y el conocimiento, pero no explica las habilidades ni transmite conocimiento a través de recetas detalladas. Asegurando el proyecto y el equipo tiene las habilidades y el conocimiento requerido para una actividad de negocios y administración separada de diseño y aplicación de la metodología.

2.3.5.2.2 Personalice las plantillas para el equipo

Una vez que se haya establecido una metodología, con una base en el marco del ciclo de vida, plantillas dirigidas a dominios específicos deben ser construidos a partir de una colección de escenarios y otros elementos, para ser utilizado por los equipos de proyecto. Como se indicó anteriormente, la metodología debe ser personalizada para que coincida con el equipo y el proyecto en cuestión. Involucrar al equipo en la personalización del proceso asegura que todos entiendan y puedan contribuir al método que usarán, al menos los aspectos más detallados de esto. El autor sugiere numerosas formas en que la metodología puede ser adaptada. Una estrategia es comenzar con una plantilla de dominio de esqueleto desnudo y embellecerla con elementos y prácticas apropiados para el equipo y el proyecto que abordan. Agregue peso, y cada adición debe considerarse cuidadosamente en comparación con los principios básicos para asegurarse que contribuirá al rendimiento del equipo. Las prácticas deben seguir siendo simples para evitar la degeneración en procedimientos prescriptivos y listas de tareas.

Otra estrategia es hacer ajustes a una plantilla de dominio basada en variaciones del dominio de destino y diferencias en la composición del equipo. Un ejemplo de variación de dominio es, si un equipo comparte ubicación o se distribuye, lo que sin duda requiere un ajuste de las prácticas de colaboración. La colaboración del equipo también deberá ajustarse para diferencias en talento, habilidades, conocimiento y personalidades. Se puede aplicar una personalización adicional a productos del trabajo. La formalidad se puede aumentar o relajar para que coincidan con las características del proyecto.

Aumentar la trazabilidad aumenta el esfuerzo, pero puede ser necesario para que coincida con la criticidad del proyecto. La importancia de que los productos de trabajo estén actualizados y reflejen el estado actual del proyecto a ser considerado. Algunos productos de trabajo pueden estar vivos y actualizados constantemente, mientras que otros pueden necesitar ser permanentes y estáticos. El autor sostiene que las prácticas de colaboración e interacción deberían ser el centro de personalización en lugar de la formalidad de los productos de trabajo u otros elementos, especialmente cuando un equipo crece.

Después de personalizar la plantilla de la metodología al comienzo de un proyecto, sus escenarios y las prácticas deben ser adaptados regularmente para reflejar mejor el comportamiento y las prácticas reales del equipo. Las retrospectivas de los equipos son una

buena oportunidad para hacerlo, y deben llevarse a cabo a un ritmo en el mismo orden que las iteraciones de desarrollo. Escalado un tema principal en la aplicabilidad de métodos ágiles para proyectos y organizaciones que la mayoría de las metodologías están dirigidas a equipos pequeños. Por lo general, a medida que los proyectos y equipos se vuelven más grandes, se agregan elementos adicionales más grandes y se incrementa la formalidad en la metodología, con un cambio hacia la optimización y una metodología más pesada.

Es importante equilibrar el aumento en el proceso con mejores prácticas de adaptación, apoyando la colaboración e interacción efectiva. Cuando se escalan las prácticas de colaboración, se debe tener cuidado para mantener un mínimo de sobrecarga para estar seguro que continúan agregando valor y apoyando al equipo en lugar de controlar o limitar el equipo. Al escalar la arquitectura del "software" y las prácticas de desarrollo, puede ser necesario dividir un proyecto grande en sub-equipos enfocados en módulos débilmente acoplados, coordinados en parte a través de un equipo de arquitectura de alto nivel que es responsable del diseño e integración de arquitectura continua, con comentarios de integración utilizados para mejorar el diseño. Un equilibrio entre anticipación al diseño adaptativo impulsado por diseño y re trabajo debe hacerse, teniendo en cuenta la velocidad a la que diferentes decisiones de diseño o componentes arquitectónicos pueden cambiar. Colaboración entre los sub equipos y el equipo de arquitectura deben ser firmes y eficientes para lograr una transferencia de conocimiento eficiente en todo el proyecto

2.3.5.3 Resumen de ASDE

En general, Highsmith (54) ofrece una excelente idea de lo que significa ser ágil y cómo acercarse a la adopción de metodologías ágiles, especialmente en lo que se refiere al desarrollo de "software". Ágil se trata de los valores y principios subyacentes contenidos en la cultura de equipos y organizaciones, y el enfoque general para la resolución de problemas que ellos tomaron, no sobre seguir una metodología ágil específica. Con temas de adaptabilidad, colaboración y simplicidad, Highsmith (54) describe la clave que conforma una metodología ágil, y explica en términos generales cómo crear una metodología que se ajuste dentro de los dominios de cultura y problemas de una organización. Aunque el autor describió previamente una serie de metodologías ágiles populares que pueden adoptarse, se hace hincapié en la personalización de la

metodología para adaptarse a proyectos y equipos específicos. Finalmente, los enfoques ágiles no son apropiados para todos los proyectos. Se destacan cuando los proyectos se centran en exploración sobre optimización, caracterizada por alta incertidumbre y cambios rápidos, donde el camino y el resultado son difíciles de predecir, lo que hace que los planes detallados sean tenues en el mejor de los casos.

2.3.6 Kanban

Anderson y Carmichael (59) proporcionan una breve introducción a Kanban y describen sus componentes, y también proporciona información sobre la aplicación de Kanban a una organización. Originalmente, el término Kanban se utilizó para describir las implementaciones de flujo de trabajo desarrolladas en Japón por Toyota en la década de 1960, para limitar el trabajo en progreso en las instalaciones de fabricación ajustada a una demanda basada en la estrategia "jalar". Kanban se ha usado más recientemente para describir una gestión de proyecto basada en flujo metodológico que adapta esta estrategia de "extracción" para mejorar la comprensión del trabajo y el flujo de trabajo, fomentando el cambio y la adaptación. El uso de un tablero Kanban para visualizar el trabajo y el flujo de ayuda en esta comprensión. El dominio objetivo de Kanban son los servicios que proporcionan trabajo de conocimiento intangible de productos a los clientes. El método de Kanban descansa sobre una base de nueve valores, tres agendas y seis principios. Los valores son transparencia, equilibrio, colaboración, enfoque al cliente, flujo, liderazgo, comprensión, acuerdo y respeto.

En un sistema Kanban, el trabajo representa un flujo de valor hacia el objetivo que proporciona valor al cliente. Tres agendas proporcionan un propósito de alto nivel para impulsar cambios en la organización. La agenda de sostenibilidad se enfoca en mejorar la eficiencia y lograr resultados dentro de la organización. La agenda de Orientación al Servicio mira hacia afuera y se esfuerza por entregar y mejorar los servicios que cumplen y exceden las expectativas del cliente.

En la supervivencia la agenda se ocupa de la evolución continua y de abrazar el cambio para seguir siendo competitivo. Los seis principios subyacentes de Kanban se dividen en dos grupos. Principios de gestión del cambio incluye comenzar con lo que la organización hace actualmente, buscando mejorar a través del cambio evolutivo (utilizando el punto

de partida como línea de base para la evaluación) y alentando actos de liderazgo de individuos en todos los niveles dentro de una organización.

Los principios son entender y enfocarse en las necesidades y expectativas del cliente, administrar el trabajo y permitir que las personas se auto organicen a su alrededor y desarrollen políticas para mejorar a sus clientes y empresa. Estos principios intentan garantizar que el enfoque de gestión de la organización permanezca en el trabajo y el flujo de valor para el cliente en lugar de administrar a las personas que hacen el trabajo.

El flujo de un sistema Kanban se visualiza en una placa Kanban, como se ilustra en la figura 24. Partes clave de un sistema Kanban incluyen el uso de señales visuales que limitan el trabajo en progreso (WIP; recopilación y recuento de elementos en el sistema), así como puntos claros de compromiso y entrega. Los compromisos son acuerdos tácitos o explícitos que indican que el cliente quiere y aceptará un artículo y el proveedor del servicio producirá y entregará el artículo. Los puntos de entrega representan la finalización del artículo. Los elementos para el compromiso se pueden seleccionar a través de un proceso de un grupo de ideas o solicitudes (similar a una acumulación de productos de "Scrum"). El tiempo de entrega (LT) de un artículo se refiere a la hora entre el compromiso y la entrega, mientras que el tiempo de entrega del cliente de un artículo se refiere al tiempo entre la solicitud y la entrega, que puede ser diferente si el cliente "empuja" la solicitud en lugar de aceptar exclusivamente "Empujes", o si hay servicios adicionales entre el sistema Kanban de interés, solicitudes de clientes y entrega al cliente. El índice de entrega (DR) puede describirse usando La Ley de Little, donde la tasa de entrega promedio es la media del trabajo en progreso dividido por la media del tiempo de entrega para un sistema de estado estacionario (es decir, entre dos puntos del mismo trabajo en progreso). La misma ley se puede usar para describir un número de otras métricas de rendimiento de interés en un sistema Kanban, donde en lugar de tiempo de entrega, tiempo en proceso, Prueba, Desarrollo, Sistema pueden proporcionar visión. Visualmente, un Diagrama de Flujo Acumulativo ilustra la Ley de Little, como se muestra en la Figura 24, y se puede observar que al limitar el trabajo en progreso se optimizará el tiempo de entrega.

KANBAN: DURANTE TODO EL PROCESO



Figura 24. Ejemplo de placa Kanban.
(Fuente: Modificación, Anderson & Carmichael (60))

Hay seis prácticas generales en la metodología Kanban que representan las principales actividades de gestión. Estas prácticas se citan a continuación:

- "Visualizar"
- "Limitar el trabajo en progreso"
- "Gestionar flujo"
- "Hacer políticas explícitas"
- "Implementar circuitos de retroalimentación"
- "Mejorar colaborativamente, evolucionar experimentalmente"

Visualizar, a menudo se logra con un tablero Kanban que muestra el flujo de trabajo horizontalmente a través de una serie de pasos como se ilustra arriba en la figura 24. Es importante que los elementos de trabajo, el compromiso, se incluyan puntos de entrega y límites de trabajo en proceso. También es bueno visualizar políticas (especialmente con respecto a cómo se mueven los elementos de trabajo entre los pasos) y resaltar las

dependencias de prevenir el progreso como se mencionó anteriormente, limitar trabajo en progreso ayuda a optimizar tiempo de entrega que mejora la capacidad de respuesta del cliente y la capacidad de adaptarse a circunstancias cambiantes.

Gestionar el flujo en un sistema Kanban para maximizar el valor, minimizar el tiempo de entrega y proporcionar cierta predictibilidad puede ser difícil, y se debe prestar atención a los cuellos de botella y las dependencias. El costo de demora y la urgencia de los elementos de trabajo ayudan a la priorización. Diferentes modelos de costo de demora pueden resultar en la aplicación de diferentes clases de servicio (con políticas variables). Las políticas simples de proceso ayudan a restringir y guiar el trabajo dentro del flujo, y siempre deben aplicarse, pero también pueden ser modificables de modo que puedan evolucionar para proporcionar un mayor valor. Ejemplos donde las políticas se pueden aplicar son para límites de trabajo en progreso, definición de hecho para la entrega, asignación de capacidad, clase de selección de servicio y reabastecimiento de artículos de trabajo.

Kanban implementa bucles de retroalimentación con cadencias, refiriéndose a ambas reuniones o revisiones, y el período de tiempo entre revisiones. Un mínimo La implementación de Kanban puede usar reabastecimiento y reuniones Kanban diarias (como un "Scrum" diario), mientras que una organización más grande puede implementar un esquema más complejo. La última práctica refleja que Kanban está diseñado para ayudar a una organización a mejorar, y para Fomentar la evolución y adaptación continua. Una metodología Kanban no puede ser considerada "Completa", más bien debe cambiar constantemente para ayudar a una organización a encajar y sobresalir dentro de su ambiente. Hay varios pasos para introducir un sistema Kanban en una organización, que incluye cómo evaluar la adopción y eficacia del sistema en áreas clave. Herramientas de pronóstico probabilístico y los métodos complementarios de monitoreo pueden mejorar la previsibilidad y los esfuerzos de planificación de la ayuda. Finalmente, los autores explican los enfoques para escalar Kanban en diferentes partes de una organización.

2.3.7 Scrumban

Scrumban (61) originalmente comenzó como un método para la transición de una metodología de "Scrum" a un enfoque Kanban, pero se convirtió en una metodología distinta en sí misma. De hecho, Reddy (62) sugiere que es un buen método para facilitar el

cambio e introducir "Scrum" y Ágile en los equipos. Scrumban se basa en elementos de "Scrum" y Kanban para lograr un enfoque flexible de flujo flexible a un marco de desarrollo más estructurado. Está dirigido principalmente al desarrollo de "software", como otras metodologías ágiles. Comparado con "Scrum", Scrumban difiere de varias maneras.

"Scrum" enfatiza en un equipo de generalistas, pero Scrumban permite y algunas veces fomenta la especialización con equipos y funciones (aunque no hace cumplir esto). Scrumban ayuda a definir cómo trabajar con políticas explícitas, y aplica las leyes de la teoría de flujo y paros. Scrumban también varía desde un enfoque Kanban. Prescribe "Scrum" como base para el "software" y se organiza en torno a los equipos. Mientras que Kanban se esfuerza por lograr un el sistema de flujo basado en "jalar", las iteraciones en el recuadro de tiempo siguen siendo una parte útil de Scrumban cuando se aplica de forma adecuada. Finalmente, las formalidades específicas se utilizan para dar más estructura a la mejora continua de las técnicas.

Reddy sugiere que Scrumban ofrece una serie de oportunidades para divergir del proceso estándar de "Scrum" para abordar deficiencias o desafíos que surgen a menudo cuando se implementa "Scrum". Los ejemplos de estos incluyen abordar el trabajo de emergencia o prioridad a mitad de "sprint", integrando un equipo auto organizado con estructuras de gestión tradicionales, aprovechando mejor las habilidades especializadas de individuos o equipos existentes, y una mejora continua más efectiva a través de un objetivo adicional como medidas de desempeño. A medida que aumenta la cantidad de tareas que un recurso es responsable simultáneamente, la productividad total disminuye a medida que se gasta una mayor cantidad de tiempo en el cambio de contexto (63).

Scrumban sugiere mejorar el flujo de trabajo y la productividad general mediante el uso de límites de trabajo en progreso, apoyados por el trabajo de buffers para mejorar la calidad, enfocar los esfuerzos y restringir la distracción y la sobrecarga. Como resultado, la gestión de riesgo y la priorización deberían mejorar. El principio de enfocar el esfuerzo y priorizar el uso efectivo de los límites de trabajo en progreso, se extiende desde el individuo al equipo y finalmente a la organización como un todo (para limitar el enfoque estratégico y priorizar las oportunidades). Se observa que el objetivo de mejorar el flujo y la productividad de todo el sistema (equipo u organización) puede resultar en una disminución en la utilización o productividad de las personas, lo que es contrario a la medida de la productividad.

Scrumban vincula el riesgo con la prioridad y el impacto de no completar las tareas por un límite de tiempo. Evaluación de riesgo e impacto es una parte importante de priorizar el trabajo en un equipo. Cuatro perfiles de riesgo comunes se sugieren: urgente o de emergencia, costo fijo, costo estándar o costo intangible o de inversión. Un perfil de riesgo urgente sugiere que una tarea es de alta prioridad y no se completará en condiciones normales: un equipo necesita detener otro trabajo y agilizar estos elementos. Un perfil de riesgo de costo fijo es de mediana a alta prioridad, y no cumplir con un plazo es perjudicial. Sin embargo, se espera el equipo tiene tiempo para completar estas tareas dentro de un límite si se planifica. Un costo estándar de un perfil de riesgo indica impacto inmediato pero poco profundo, donde las tareas no tienen una fecha límite explícita, pero aún pueden incurrir en costos debido a la demora. Se sugiere que estas tareas sean atendidas por orden de llegada.

Los perfiles de costos intangibles o de inversión no suelen tener un impacto directo significativo, pero son tareas necesarias para apoyar el progreso del equipo (por ejemplo, abordar la deuda técnica o mejorar la arquitectura, es decir, volver a trabajar). Al implementar Scrumban, el primer paso es crear un tablero Kanban que represente los flujos existentes de trabajo y prácticas, considerando insumos, productos, procesos y enlaces a partes interesadas externas o sistemas. El diseño inicial de la placa debe ser simple y puede evolucionar con el tiempo. El equipo debe comenzar utilizar el tablero para reflejar elementos de trabajo reales, y el flujo puede comenzar a medirse y evaluarse durante reuniones diarias usando métricas y visualizaciones de trabajo en progreso, barreras, tiempo de entrega, y rendimiento. Un diagrama de flujo acumulativo también proporciona una idea. A continuación, introduciendo límites en el trabajo en progreso puede ayudar a estabilizar el sistema, un paso necesario antes de que las mejoras significativas puedan hacerse. Se pueden hacer mejoras adicionales al sistema de flujo de trabajo a través del entendimiento y la gestión del riesgo, de los cuales es especialmente importante abordar el coste del retraso. Con riesgos los métodos de gestión mejoran la entrega de valor al cliente con la reducción de desperdicio ocupaciones. Se pueden obtener más conocimientos y mejoras al cuantificar el valor del trabajo para ser realizado en términos de costo e ingresos, y usar esto como parte de las decisiones de priorización del trabajo. La visualización del valor y las diversas métricas del sistema ayudan a comprender a todo el equipo y debe integrarse con la visualización del flujo del sistema de la placa Kanban.

Finalmente, la mejora continua y la evolución deberían formar parte de la cultura del equipo, ayudada en parte por el uso de retrospectivas de equipo para reflejar e introducir nuevos conceptos.

2.4 CONTRASTE ENTRE MÉTODOS ÁGILES Y MÉTODOS DE DISEÑO

Anteriormente, se ha visto una variedad de metodologías de administración de proyectos de diseño, ambas tradicionales que normalmente se aplican a productos físicos, así como a métodos ágiles que han sido principalmente dirigidos al desarrollo de "software". Cada una de estas metodologías puede ser categorizada por un número de métodos, uno de los más importantes es el papel de la iteración. Las metodologías también se pueden prestar bien a los modelos y simulaciones para ayudar a comprender y mejorar los procesos, especialmente con respecto a la coordinación entre el equipo. Además, las metodologías de diseño encajan en el conjunto del ciclo de vida del producto, y tienden a tener etapas de diseño similares, independientemente de la disciplina o la industria.

2.4.1 Investigar las iteraciones del proceso de diseño de ingeniería

Safoutin (64) investigó empíricamente el uso de la iteración en el proceso de diseño instrumentando un "software" de diseño de ingeniería para investigar el uso de la iteración por estudiantes de ingeniería de una manera simplificada en una clase de diseño. El autor observa que muchos consideran el patrón de diseño de ingeniería ideal sin considerar la iteración (es decir, un modelo de cascada), considerando a la iteración generalmente como una luz negativa y como una acción correctiva inevitable para corregir los errores que causan retrasos.

Safoutin (64) afirma que la iteración es necesaria y puede ser beneficiosa, especialmente como una forma de proporcionar retroalimentación. Un modelo que describe presenta tres tipos diferentes de iteración visto en diseño, tales como: iteración de repetición, iteración de progresión e iteración de retroalimentación. La Figura 25 ilustra los enfoques de iteración descritos. La iteración de repetición se describe como incrementos sucesivos de las mismas actividades que conducen al resultado final sin resultados intermedios. Aplicado al diseño dividido en características, esto representa un diseño descendente, enfoque de diseño en profundidad. La iteración de progresión busca crear resultados intermedios de todo a niveles crecientes de detalle. Este es un enfoque de arriba hacia abajo, de amplitud, que cubre un diseño con resultados progresivamente más detallados. La iteración de

retroalimentación extiende la iteración de progresión a integrar explícitamente los resultados o la información de los resultados en cada nivel de detalle para lograr una mayor precisión en el resultado del diseño final.

Desde el punto de vista del desarrollo mecánico de productos, Safoutin (64) saca a la luz que la iteración es inevitable e incluso beneficioso frente a la mentalidad de cascada tradicional. Safoutin (64) también sugiere que cada enfoque, profundidad o amplitud en primer lugar, proporcionan diferentes ventajas en la comprensión de un diseño complejo. La sugerencia de utilizar un enfoque de iteración con retroalimentación entre los niveles de detalle, aún refleja una metodología de proyecto de diseño de cascada tradicional, siguiendo el concepto de diseño a través de un diseño detallado. Esto es para tratar con las posibles interdependencias entre las características, donde completar una característica por completo puede tener impactos negativos en otros. La recomendación de incluir comentarios es útil, como lo es la idea de los resultados intermedios (es decir, un diseño conceptual o prototipo inicial). Sin embargo, es posible aproximar un enfoque ágil al mismo modelo.

Una metodología basada en características (como es adoptada por la mayoría de los enfoques ágiles) es similar a la iteración de repetición, donde una característica se lleva rápidamente a la finalización del entregable. Abordar los desafíos de la interdependencia e incorporar el conocimiento futuro en una característica existente, el enfoque ágil usa dos enfoques. Primero, se recomienda el re-trabajo. La integración de nuevas características puede necesitar un cambio en una característica existente. El costo de esta repetición se reduce al considerar esta posibilidad en el diseño original, pero también es aceptable para asignar el tiempo y los recursos a esta actividad. En segundo lugar, las características y subsistemas están diseñados para ser tan modulares y desacoplados como prácticos, lo que reduce los impactos de los cambios de una característica en otra. Mientras que algunos proyectos pueden no ser utilizables hasta estar completo, muchos pueden comenzar a mostrar valor con una finalización parcial. Este es especialmente el caso con proyectos de "software", pero también puede estar presente en otras disciplinas.

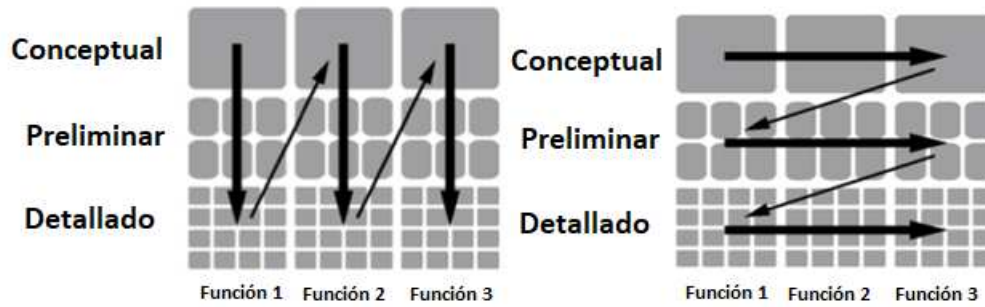


Figura 25. Enfoques de diseño incremental (izquierda) y progresiva (derecha).
 (Fuente: Modificación John Safoutin (64))

2.4.2 Mejora del proceso de diseño con enfoques basados en modelos

Wynn (65) presenta un marco de modelado formal y detallado para el proceso de diseño dirigido a planificación y análisis de desarrollo de productos complejos. Este marco proporciona la base de la costumbre de "software" desarrollado por el autor para ayudar en la planificación y simulación de grandes proyectos con muchas secuencias de tareas jerárquicas, secuenciales, relativamente independientes. Este enfoque basado en modelos intenta abordar las limitaciones de los métodos tradicionales basados en gráficos de Gantt en torno a la ordenación de tareas y asignación de recursos, especialmente frente a la incertidumbre y la iteración.

El autor usó el producto del programa de desarrollo en Rolls-Royce para hacer las observaciones útiles en la creación del marco de modelado y especificando el "software" (ahora conocido como Cambridge Advanced Modeler, o CAM (66)), y más tarde aplicando el "software" a un caso real. El "software" y el enfoque de modelado se consideraron exitosos. Es importante reducir el costo del modelado de procesos, mejorar la comunicación de las partes interesadas con respecto a la gestión del proyecto, y proporcionar información basada en el análisis para mejorar el proceso de gestión. Se observa que el "software" es limitado y no proporciona tareas simultáneas o fuertemente vinculadas o dependen dientes, ni considera la selección adaptativa de tareas basada en la retroalimentación del proyecto actual estado.

Wynn utiliza tres perspectivas para clasificar los modelos de diseño. El primero clasifica el proceso de diseño en cualquier etapa o actividad orientada. Los enfoques basados en etapas son más estructurados y tienden a ser secuenciales y cronológicos con comentarios

limitados. Los modelos de diseño basados en actividades tienden ser cíclico y reelaborar intensivamente. En la segunda perspectiva, los enfoques orientados a problemas típicamente centrarse en gran medida en el análisis de problemas antes de pasar a generar múltiples opciones de solución. Los métodos orientados a soluciones se mueven a una única solución temprana, que luego se analiza y modifica de manera iterativa durante la exploración del espacio y los requisitos del diseño. La tercera perspectiva categoriza los modelos de diseño como abstractos, de procedimiento o analíticos. Los métodos abstractos proporcionan alto nivel, orientación general, pero brindan apoyo limitado para la mejora específica del proceso. Procesar los enfoques proporciona mayor relevancia a situaciones prácticas y aspectos específicos del proceso de diseño, y pueden estar centrados en el diseño o en proyectos. Los enfoques analíticos buscan modelar un proceso de diseño a la instancia en detalle con el análisis de realización de objetivos posible que se puede utilizar para mejorar el proceso. Este método aprovecha un marco de modelado junto con técnicas, procedimientos o herramientas de computadora. El autor sugiere que los enfoques basados en etapas suelen estar orientados a la solución, mientras que los métodos basados en la actividad pueden usar cualquiera. Los métodos abstractos a menudo se basan en actividades, pero pueden ser ya sea orientado a problemas o soluciones. Los enfoques de procedimiento suelen estar orientados a los problemas. Un método analítico puede diseñarse para adaptarse a cualquier categoría. Wynn (65) promueve la categorización por característica de las metodologías de diseño. El deseo de mejorar la planificación del proyecto y la capacidad de retroalimentación más allá de lo que es posible con un diagrama de Gantt, lo cual refleja un cambio mayor al control del proceso.

Sin embargo, el entorno objetivo de Wynn (65) y las limitaciones explícitas en el marco de modelado, así como la implementación de "software" aún sugieren un enfoque de gestión tradicional para productos de trabajo independientes, secuenciales y ligeramente iterativos. El método analítico está dirigido a 90 metodologías de gestión de proyectos prescriptivos y de procedimientos y no proporcionan la capacidad de adaptarse al proyecto o al equipo. Los enfoques ágiles pueden clasificarse generalmente como actividades orientadas a soluciones, pero ciertamente trae elementos de enfoques abstractos, procesales y analíticos. Además, los elementos basados en etapas y orientados a problemas pueden estar presentes o ser útiles en algunos casos, pero son quizás menos

adecuados para los dominios de problemas exploratorios. Los métodos ágiles intentan hacer que los practicantes vean el sistema como un todo, a la vez que proporciona orientación y prácticas para situaciones específicas, así como suficientes análisis y datos posibles para la retroalimentación y la mejora continua.

2.4.3 Coordinación en desarrollo de productos complejos

Suss (67) investiga la coordinación entre equipos involucrados en el desarrollo de grandes proyectos a través del modelado y simulación, en un esfuerzo por mejorar el diseño de metodologías de ingeniería. El enfoque del autor es similar al de Wynn (65), pero amplía el modelado de tareas, considerando la incertidumbre y la generación de conocimiento dentro de una tarea, así como la coordinación e interdependencia entre tareas desde el punto de vista de la información. Esto aborda algunas de las deficiencias del trabajo previo.

El uso del enfoque del modelado estocástico de mayor fidelidad permite investigar las relaciones y los impactos de las estrategias de coordinación, la iteración y el re trabajo e incertidumbre al intentar reducir el tiempo de desarrollo del producto. Los proyectos grandes son desglosados en tareas pequeñas y especializadas conectadas por dependencia compartida (ejecución paralela e independiente) contribuyendo al objetivo final; la dependencia secuencial y la dependencia recíproca. La dependencia agrupada se maneja de manera efectiva con reglas o estándares, especificando el resultado final antes de tiempo. La dependencia secuencial se aborda con la planificación, mientras que la dependencia recíproca se beneficia de la retroalimentación y el ajuste mutuo. Se sugiere que cuanto menos rutinaria sea la tarea, más un enfoque de retroalimentación mejorará el resultado frente a la diversidad. Además, a medida que aumenta la incertidumbre, las estrategias de coordinación deben pasar de las reglas o la planificación a más reuniones y una mayor comunicación horizontal.

Suss (67) modela la incertidumbre epistémica y estocástica (de la información) del proceso de tareas de diseño. El modelo epistémico sigue una curva en S para reflejar una tendencia de ganancia de conocimiento más rápida en el medio de una tarea. Como parte de la serie de una simulación, el autor intenta modelar un estilo de proceso ágil "Scrum" de desarrollo de productos. Basándose en los resultados de la simulación, Suss (67) concluye que siguiendo un proceso de desarrollo similar a "Scrum", con resultados intermedios bien definidos y cortos ciclos de retroalimentación y revisión que ayudan a mejorar la

coordinación y enfocarse dentro del equipo de proyecto, lo que lleva a una reducción del trabajo de diseño e iteraciones, especialmente frente a la incertidumbre.

El autor afirma que un enfoque ágil es beneficioso a todo tipo de proyectos de diseño y desarrollo de productos, y no se limita estrictamente al desarrollo de "software". La coordinación mejorada y la reducción del tiempo de desarrollo están respaldados por un número de mecanismos descritos en el trabajo. La planificación del proyecto, las políticas y los sistemas deberían contribuir a la reducción del riesgo del proyecto mediante la programación temprana de tareas inciertas, que reduce la incertidumbre general cuanto antes. También deberían diseñarse para aumentar la comunicación y el flujo de información dentro de un proyecto para reducir los cuellos de botella que pueden afectar el tiempo de desarrollo de una manera no lineal. En general, los resultados sugieren que los métodos ágiles de gestión de proyectos de diseño ofrecen ventajas sobre enfoques tradicionales.

2.4.4 Modelado de diseño y modelado funcional en todas las disciplinas

Eisenbart (68) Revisó la literatura de múltiples disciplinas para desarrollar un marco que pueda ser utilizado para comparar modelos de diseño o documentos en todas las disciplinas. Los modelos de diseño de varias metodologías se utilizan para capturar información sobre el diseño en un punto específico del progreso del proceso de diseño, es decir, el estado del diseño. Los autores identificaron disciplinas específicas y genéricas a los estados de diseño, que se pueden categorizar como estados problemáticos (generalmente agnósticos a la solución) y estados del producto (información específica de la solución).

Los modelos de diseño se analizan utilizando un modelado morfológico propuesto por Buur y Andreasen (69). El trabajo resalta la similitud en el diseño a través de diseño mecánico, eléctrico, "software" y de construcción. Se sugiere que el método de modelado sea adecuado para su uso en disciplinas, como en el caso de los sistemas mecatrónicos, es necesaria. Sobre la base de esto, Eisenbart (68) revisa una gama de métodos de modelado funcional utilizados en 92 diferentes disciplinas, que mostraron una amplia gama de enfoques y perspectivas. Un análisis más extenso de modelado funcional sugiere una perspectiva de proceso de transformación es fundamental para el modelado a través de múltiples disciplinas, lo que podría proporcionar una base para un marco de modelado

integrador considerando que las perspectivas específicas del proyecto podrían incluirse según sea necesario [33]. Finalmente, Eisenbart (68) Propone un Marco de Modelado Funcional Integrado (IFM) basado en el concepto de matrices multi dominio que se puede usar para una conceptualización eficaz del sistema interdisciplinar, con suficiente modularidad y extensibilidad para adaptarse a diferentes proyectos. Este marco se basa en la transformación de la perspectiva del proceso, pero también incorpora otras seis perspectivas. El marco identifica entidades relacionadas y puntos de vista centrales para apoyar estas perspectivas. El marco IFM continúa extendiéndose y está en continuo desarrollo.

2.4.5 Etapas en el ciclo de vida del producto en una metodología

Qureshi (70) evaluó una amplia gama de metodologías de diseño y gestión dentro del ciclo de vida del producto, para obtener un marco común de etapas de diseño útiles en colaboración y comunicación durante el desarrollo de productos interdisciplinarios. Las etapas de diseño en este marco se dividieron en dos niveles, inspirados en las principales actividades o etapas más típicamente relevantes a los ciclos de vida del producto y el mayor detalle a menudo presentado en las metodologías de diseño, que típicamente solo cubren un subconjunto del ciclo de vida del producto. También fueron influenciados por el equipo multidisciplinario de los estados de diseño sugeridos por Eisenbart (68). Estos ciclos de vida y etapas de diseño son listados a continuación:

Inicio

- Establecer una necesidad
- Análisis de mercado y pronósticos
- Identificación de la necesidad
- Gestión de proyectos
- Especificación de requisitos

Diseño

- Diseño conceptual
- Diseño de la realización
- Diseño detallado

- Desarrollo de sistemas de producción

Implementar / realizar

- Fabricación
- Ensamble
- Integración de sistemas
- Adquisiciones

Uso / Soporte

- Ventas y distribuciones
- Instalación
- Operación
- Servicio y mantenimiento

Fin de la vida útil

- Retirar / Deshacer / Liquidar

Usando este marco, se completó un estudio empírico industrial multidisciplinario para comprender los procesos de diseño utilizados en la industria en comparación con los presentados en la literatura. Este estudio mostró la utilidad del marco común para describir los procesos de diseño y los ciclos de vida de los productos en la industria a través de múltiples disciplinas y contextos o mercados. Los participantes pudieron hacer coincidir los procesos de su organización a las etapas generales del marco bastante bien. Además, allí tendió ser un enfoque de niveles claros para el flujo del proceso con la mayoría de las organizaciones que utilizan un enfoque de compuerta por etapas en un nivel superior, mientras se adopta un proceso más iterativo dentro de las etapas. Se observa un estudio enfocado principalmente en organizaciones grandes (17 de 23 organizaciones) con grandes equipos de diseño (250 a 1000 empleados).

2.5 MÉTODOS ÁGILES APLICADOS MÁS ALLÁ DEL “SOFTWARE”

Existen ejemplos limitados que demuestran el uso de enfoques ágiles en el diseño fuera del campo del “software”, aunque hay numerosas explicaciones de cómo un equipo puede intentar hacerlo en la forma de artículos de Internet y publicaciones en “blogs”.

2.5.1 Idoneidad del proyecto para metodologías ágiles

Alite (71) argumenta que las metodologías ágiles no son adecuadas para todos los proyectos. Una visión general de las características se describe en un proyecto que puede beneficiarse de un enfoque ágil, así como atributos o tipos de proyectos que se adaptan mejor a enfoques más tradicionales. Características del proyecto, el equipo, el cliente y la organización son importantes. Los proyectos a menudo tienen alto niveles de incertidumbre (especialmente en los requisitos), complejidad y riesgo de falla, y la base o plataforma debe ser flexible. El equipo debe ser pequeño con una gran cartera de habilidades, y comprometido con la calidad, las pruebas disciplinadas y la comunicación con el cliente.

El cliente debería participar activamente en todo el proyecto y sentirse cómodo con los conceptos ágiles. La organización debe apoyar el enfoque ágil, y el trabajo de desarrollo debe completarse dentro de la organización y equipo (lo ideal sería que el equipo esté en el mismo lugar). Alite (71) establece varias características importantes, incluida la alta tolerancia al riesgo del proyecto, un experimentado equipo de desarrollo, la capacidad del usuario final para adaptarse al cambio para seguir una evolución rápida, y buena flexibilidad dentro del presupuesto del proyecto.

Se presentan ejemplos de proyectos de desarrollo de "software", que el autor sugiere que no encajan bien con un enfoque ágil, que incluye bases de datos, "software" integrado, proyectos computacionales complejos, y equipos de respuesta a incidentes. Se puede argumentar que si bien el cuidado adicional o la modificación del proceso pueden ser necesario, estos proyectos se manejan fácilmente mediante un enfoque ágil. Es probable que Alite (71) reconozca que estos proyectos pueden necesitar un diseño inicial adicional y trabajo de arquitectura antes de la codificación. Esto se incluye fácilmente en un proceso ágil, manteniendo los valores y principios ágiles. En el caso de un equipo de respuesta a incidentes, un método "Scrum" puede no ser efectivo debido al cambio potencial de las prioridades. Un enfoque de tipo Kanban o Scrumban es probablemente mucho más apropiado. Alite (71) hace una observación interesante de que las metodologías ágiles son más complejas que las metodologías tradicionales. Se puede argumentar que si bien la descripción general del proceso a menudo es más detallada para un enfoque ágil más que para un proceso tipo cascada, la formalidad y los detalles dados a los entregables intermedios y producto de trabajo, es probablemente mucho mayor. Es obvio que cualquier método será más complejo que un enfoque ad-hoc. El autor resume el proceso

de seleccionar una metodología apropiada al sugerir preguntas que pueden usarse para ayudar en el proceso de toma de decisiones, como se cita a continuación:

- "¿Están claros los requisitos o pueden cambiar?"
- "¿El cliente entiende completamente y se siente cómodo con el método que el proyecto emplea?"
- "¿Esta el proyecto cómodo con la metodología y cuál es su actitud de riesgo?"
- "¿Cómo se llevará a cabo el proceso de desarrollo? ¿Se hace en la empresa, o se subcontrata parte del desarrollo?"
- "¿Cuáles son las características del equipo de desarrollo?"
- "¿Qué tan rápido necesita el producto ser operacional? ¿Se requiere funcionalidad completa en el primer lanzamiento?"
- "¿Cuáles son los requisitos de calidad, tiempo y costo?"

Estas preguntas ayudan a determinar las características del proyecto y evaluar la idoneidad para una metodología ágil. Alite (71) sugiere que seleccionar una metodología apropiada puede ser beneficiada de investigaciones adicionales sobre la influencia de las diferentes características en el éxito del proyecto, y una forma de cuantificar un enfoque ágil, frente a los enfoques tradicionales en términos de calidad, calendario y costo para una comparación directa. Finalmente, los desafíos asociados con el trabajo con clientes nuevos deben ser abordados.

2.5.2 Ingeniería de sistemas ágiles versus sistemas ágiles de Ingeniería

Haberfellner (72) trata de aclarar y definir la diferencia entre un proceso de diseño ágil (es decir, gestión ágil) y un diseño ágil (es decir, características de diseño), especialmente en el contexto de la ingeniería de sistemas. Se sugiere que un proceso de diseño ágil ayuda a abordar la incertidumbre durante el desarrollo (como los requisitos ambiguos del cliente), mientras que un diseño ágil considera la incertidumbre y los cambios potenciales a lo largo del ciclo de vida del producto o sistema. Un proceso ágil se esfuerza por ser adaptativo y receptivo a nueva información durante el desarrollo, en lugar de requisitos congelados y

soluciones tempranas como un enfoque tradicional puede alentar. En el contexto de la ingeniería de sistemas, Haberfellner (72) describe cómo determinar la necesidad de agilidad mediante la comprensión del dominio del problema, y luego describe tres niveles de agilidad dentro del proceso de diseño. Se necesita una mayor agilidad según, si los requisitos son menos estables: ¿Qué requisitos pueden cambiar?, ¿cuánto? y en qué dirección y el costo de lograr un amplio rango en los requisitos desde el comienzo (en lugar de ser capaz de cambiar el enfoque durante la ejecución).

Los tres grados de agilidad se describen a continuación;

- El primer grado es simplemente la introducción de alguna agilidad y mayor iteración al proceso de diseño de ingeniería de sistemas existente.
- El segundo grado sugiere el uso de la ingeniería por partes el uso de diseños modulares para fomentar la reutilización y reducir el acoplamiento.
- El tercer grado introduce el uso del diseño basado en conjuntos, mediante el cual el concepto de múltiple diseños, tal vez desarrollados por equipos de diseño independientes en competencia paralela, se presentan como cerca de la finalización como sea posible, en un esfuerzo por explorar la mayor cantidad de espacio de solución de diseño como sea posible, dentro de las limitaciones de tiempo, costo y requisitos.

Estas ideas son similares a las promovidas por una metodología ágil, pero ciertamente más limitadas dentro de los marcos existentes que de los sistemas grandes o los proyectos de ingeniería suelen habitar.

Dibujar la distinción entre agilidad en proceso y diseño, es una idea interesante: la primera es típicamente detallado en gran medida por la literatura que respalda las metodologías ágiles, mientras que el último se hace referencia como parte de los objetivos de excelencia técnica, calidad y la capacidad de adaptar rápidamente el diseño para cumplir con los requisitos cambiantes y los valores comerciales. El autor describe explícitamente algunas de las propiedades que debe tener un diseño ágil. Un diseño ágil debe ser flexible, reconfigurable, y / o extensible, que puede manifestarse como la capacidad de agregar o cambiar módulos para variar el rendimiento, o ser escalable para cumplir con las

demandas de capacidad. Tales características son importantes especialmente si el ciclo de vida del producto o sistema tiene incertidumbre. Las vidas cortas y los entornos operativos estables reducen la incertidumbre. Por otro lado, los sistemas diseñados para largos períodos de operación pueden encontrar un cambio imprevisto en los requisitos, debido a cambios en el mercado u otras partes del medio ambiente, especialmente si el costo de reemplazar o actualizar el sistema, es alto. Como se mencionó anteriormente, el costo de cumplir con una amplia gama de requisitos (es decir, diseñar anticipadamente el peor de los casos) puede ser muy costoso. Haberfellner (72) sugiere diseñar una flexibilidad para adaptarse a todo el rango del peor de los casos al mejor caso. Un diseño tan ágil debe tener elementos flexibles para soportar rápida y fácilmente los cambios, un método para medir cuándo se pueden necesitar los cambios (por ejemplo, sensores para monitorear el sistema), y un método para ayudar a tomar decisiones que equilibren el costo y el beneficio de realizar los cambios.

2.5.3 Una base empírica para la flexibilidad del producto

El concepto de flexibilidad del producto también es explorado por Rajan (73), quien define la flexibilidad como "la facilidad con el que se puede imponer un conjunto de cambios a una solución de diseño determinada." Un método empírico fue sugerido para evaluar la flexibilidad de los productos, conocidos como modos de cambio y análisis de sus efectos (CAEM), que sigue el mismo concepto que un AMEF. Esta técnica parece descomponerse un producto en módulos o partes y comprender cómo cada parte puede cambiar para reflejar las variaciones en requisitos o ciclo de vida, y cómo estos cambios afectan a otros componentes en el sistema. Esta se puede usar para mejorar la flexibilidad en el diseño. El autor sugiere que la técnica es útil en identificar las áreas clave para enfocarse durante el rediseño, así como durante los estudios de mercado y análisis de la competencia.

Se evaluaron varios productos utilizando la técnica de CAEM, identificando varios factores que mejoran la flexibilidad del producto:

- Modularidad y menos integración
- Módulos como parte del producto, que se conectan externamente
- Mayor uso de componentes e interfaces estándar

- Partición dirigida del diseño en un mayor número de elementos (más componentes y Funciones)
- La reducción del número de piezas por módulo no afecta la flexibilidad (pero probablemente mejore la capacidad de montaje)

El autor señala que el método CAEM puede consumir mucho tiempo debido al esfuerzo del cliente, la revisión y evaluación de la compañía, así como la falta de una métrica genérica para evaluar la flexibilidad no específica para cada caso

2.5.4 Aplicación de “Scrum” a un proyecto para prototipo de controles de vehículos

Bonderczuk (74) describe la adopción experimental de una versión ligera de “Scrum” en un equipo de tres integrantes de “software” para el desarrollo de un prototipo de un sistema de control de vehículos. El equipo trabajó en un entorno interdisciplinario, incluidos los ingenieros eléctricos y mecánicos que trabajan en otros subsistemas. La metodología de “Scrum” se modificó para adaptarse al equipo. Los cambios incluyeron la implementación la acumulación de productos como un documento de requisitos y especificaciones en vivo en lugar de historias de usuarios. Además, el equipo se reunió de forma intermitente ya que el proyecto era extracurricular y realizar reuniones diarias era imposible. En su lugar, las reuniones de revisión y planificación de “sprints” se combinaron en un solo evento en junto con otros líderes del equipo del subsistema, mientras que la comunicación se logró en los “Scrum”s diarios. En cambio, se logró a través de una lista de tareas de la pizarra colaborativa y mensajes de texto SMS entre los miembros del equipo. El autor concluyó que usar una metodología ágil de estilo “Scrum” en lugar del método ad hoc anterior mejoró la comunicación, la comprensión técnica, la colaboración, y claridad de los objetivos. También se observó que una larga cola de tarea de retraso acumulado, redujo la efectividad del equipo de desarrollo, mientras que la ventaja de colaboración clave fue en la comprensión del sistema actual en lugar de hacer referencia a la documentación obsoleta de los requisitos y las especificaciones.

2.5.5 “Scrum” en desarrollo de productos mecánicos

Reynisdóttir (75) exploró si un método ágil se puede aplicar al desarrollo de productos mecánicos. Esto se logró a través de la observación detallada de la introducción y la

aplicación de una metodología de "Scrum" para un equipo de ingeniería mecánica que trabaja en un solo proyecto, conjuntamente con un equipo de "software" incrustado de "Scrum". Este estudio cualitativo se llevó a cabo a través de observaciones, entrevistas y conversaciones informativas. El autor intenta comprender si la adaptación del método es necesario, ¿qué factores fueron críticos para el éxito?, los desafíos que se encontraron, y los efectos de dos equipos de "Scrum" que colaboran en el mismo proyecto. En general, Reynisdóttir (75) concluye que usar "Scrum" para administrar el equipo que lleva a cabo el producto mecánico el desarrollo, es factible, con algunas adaptaciones en el marco de "Scrum", especialmente para proyectos con alto riesgo, incertidumbre y cambios. No estaba claro si "Scrum" es mejor que otros métodos, pero fue una mejora con respecto a un enfoque ad hoc. Un desafío particular fue la entrega de un verdadero producto de trabajo en cada incremento, que es mucho más fácil de lograr en un desarrollo de "software". También se observó que "Scrum" no es aplicable a proyectos donde es posible una planificación escasa o nula, como en las solicitudes de mantenimiento. Una metodología Kanban puede ser una mejor opción en estos casos. El autor describe las adaptaciones a "Scrum" que realizó el equipo mecánico. Mientras "Scrum" alienta a un equipo completamente multifuncional, el equipo mecánico era más especializado y no tenía todas las habilidades requeridas para crear el producto final. Por lo tanto, el equipo mecánico trabajó con otros equipos asignados al proyecto. Las duraciones de "sprint" fueron variadas en contraste con el estricto período de tiempo de un marco "Scrum" normal para acomodar eventos tales como tiempos de entrega y prueba de campo. El backlog estándar del producto fue reemplazado por el plan de lanzamiento del equipo. No se mencionó si el conjunto de habilidades era suficiente para un sub proyecto claramente delimitado. El equipo debería ser responsable del 100 por ciento de la colaboración con el propietario del producto. El propietario del producto era responsable de la priorización y aprobación. Demostraciones y revisiones de productos al final de los "sprints" concluidos, son necesarias entre las partes interesadas fuera del equipo de desarrollo y propietario del producto para una mejor retroalimentación.

Una serie de factores se consideraron críticos para el éxito del proyecto. Tener una experiencia. El entrenador ágil y el maestro de "Scrum" ayudaron a la curva de aprendizaje y la eficacia del equipo. El apoyo y la comprensión de la alta gerencia también fueron importantes; idealmente, el impulso para adoptar los principios ágiles comienza en el nivel

superior. Por otro lado, el equipo también debe estar comprometido con el método, y debe contribuir a la adaptación y mejora del método, como lo sugiere la literatura ágil. El propietario del producto debe proporcionar una visión y dirección activamente para mantener al equipo de desarrollo enfocado. Finalmente, la colocación del equipo de desarrollo, así como la proximidad a otros equipos asignados al proyecto, mejoraron la productividad. Los desafíos específicos y deficiencias fueron delineados por el autor. El equipo de desarrollo encontró que el proceso de descomposición del trabajo es difícil, cuando se realizan tareas de diseño más grandes y se dividen en elementos más pequeños. Las revisiones necesitaban mejoras para ser más efectivas. Los miembros del equipo estaban acostumbrados a trabajar más de forma independiente y se requirió un ajuste a un enfoque más colaborativo. El dueño del producto no estaba tan disponible como el equipo de desarrollo lo necesitaba, y las dependencias externas no siempre estaban disponibles apropiadamente. Finalmente, el proyecto fue grande y complejo, lo que no necesariamente fue el mejor momento para experimentar con una nueva metodología. El equipo de desarrollo de productos mecánicos trabajó estrechamente con el equipo de desarrollo de "software" asignado al mismo proyecto general. El equipo de "software" ya estaba usando "Scrum" como parte de su estrategia de gestión. Una vez que el equipo mecánico adoptó el "Scrum" también, hubo una mejora en la coordinación, comunicación y cooperación entre los dos equipos. La sincronización de dependencias entre equipos como las pruebas se mejoró. Cada equipo también logró una mayor comprensión de su impacto en el otro, y de cómo lograr sus objetivos compartidos.

2.5.6 Recursos informales en línea

Usando "Scrum" en proyectos de "hardware" Maccherone (76) afirma que "Scrum" se puede aplicar efectivamente en el contexto del desarrollo de proyectos de "hardware" con algunos ajustes. En particular, las expectativas deben modificarse en el área de "característica comercializable mínima", así como en las historias de usuarios. Las metodologías ágiles suelen fomentar la entrega de características terminadas y utilizables en cada iteración que cubren una porción vertical completa del producto. El autor sugiere un punto de "equilibrio" entre el diseño y la implementación, debido al costo de fabricación de un producto físico.

En lugar de crear un producto terminado en cada iteración, los prototipos brindan un valor similar. Si esto es desafiante, incluso la producción de documentos, diseños o experimentos

puede ser suficiente, si se utilizan para provocar retroalimentación lo más cerca posible del cliente. Las historias de los usuarios ayudan a gestionar los requisitos en los enfoques ágiles. Si bien estos pueden ser un reto para adaptarse directamente a un proyecto de "hardware", el núcleo subyacente de ¿quién?, ¿qué?, ¿por qué en la historia y las conversaciones alrededor de crear la historia son beneficiosos?

Conversaciones entre el equipo de diseño y las partes interesadas ayudan a mantener la visión y a tener en cuenta consideraciones, como el costo en el diseño anticipadamente. El quién de la historia de un usuario alienta a pensar tanto en el usuario final, así como en la característica específica, y el valor entregado. La descripción de la característica debe permanecer independiente de la solución, mientras que el motivo o las razones subyacentes de la función para mejorar la flexibilidad y la creatividad de resolver el problema en lugar de cumplir con una especificación, deben permanecer dependientes.

Los proyectos de "hardware" deberían priorizar las características por valor comercial, en lugar de valor para el usuario final. El método de la historia del usuario para describir las características y especificaciones, también puede ser aumentado por otras técnicas de administración de requisitos. Por ejemplo, los casos de uso (CU) pueden ayudar a comprender las preocupaciones de seguridad y protección, los protocolos y las interfaces pueden requerir una mayor definición y descripción, así como las especificaciones del entorno operativo se puede expresar más claramente como pares métrica / valor (si esto se conoce). El autor alienta a tratar de alcanzar un objetivo de prototipos de iteración única con la ayuda de capacidades de creación de prototipos (interna o subcontratada), aunque el costo del proceso debe equilibrarse con el costo del tiempo. Si este objetivo no es razonable, desglosar el desarrollo del prototipo en varios, las etapas puede ser una alternativa. Esto ayuda a abordar los plazos de entrega y los recursos compartidos. Por ejemplo, el diseño del prototipo y la adquisición se pueden completar en una iteración, y el ensamblaje del prototipo y la prueba se completa después de omitir una iteración para permitir la fabricación y el envío.

La administración de las dependencias y la ruta crítica generalmente se maneja con planes y herramientas a más largo plazo, como Diagramas de Gantt, pero esto puede no ser realmente necesario. Estas actividades se manejan continuamente a través de planificación de las iteraciones. Es posible que los especialistas o recursos compartidos

asignados al proyecto no estén disponibles todo el tiempo. Maccherone (76) sugiere que si es posible, los especialistas sean parte del equipo a tiempo completo y entrenado para más tareas generalistas, si la carga de tareas del especialista es ligera. Esto refleja la preferencia de "Scrum", para construir un equipo de generalistas y consideraciones del costo del cambio de tarea. Accediendo a compartido los recursos pueden beneficiarse de un enfoque similar al utilizado para proveedores externos.

2.5.6.1 Desarrollo ágil para procesos mecatrónicos

Ackson (77) describe una serie de puntos clave a considerar cuando se aplican métodos ágiles a sistemas mecánicos, proyectos de diseño eléctrico y de sistemas. El "software" de trabajo es análogo al sistema funcional de prototipos. Esto debe ser respaldado por prácticas de prototipo rápido para fomentar iteraciones rápidas, la experimentación y el potencial de obtener retroalimentación rápida de fallas y éxitos.

Las metodologías ágiles a menudo fomentan la responsabilidad del equipo y el diseño colaborativo a través de la interacción, en lugar de documentación para lograr adaptabilidad y velocidad. Esto desafía la "responsabilidad tradicional", cultura de la ingeniería ", donde los ingenieros individuales asumen la responsabilidad mediante el estampado de documentos, lo que puede conducir a diseños más conservadores. Un desafío difícil entre proyectos de "software" y proyectos de "hardware ", es que el "software" puede mejorarse continuamente con actualizaciones y nuevas versiones, mientras que los productos de "hardware ", generalmente se bloquean en una sola configuración una vez enviados. Esta es una diferencia significativa en el paradigma, y la diferencia entre una versión de producción y una versión prototipo de un producto debe considerarse.

Thilmany (78) también sugiere que un enfoque ágil es beneficioso para el desarrollo de productos, especialmente si los requisitos no son estables. La adopción de ágil se ve desafiada por los plazos de entrega y las partes interesadas que son no directamente parte del proceso de diseño (por ejemplo, proveedores y fabricantes). El autor sugiere que el enfoque ágil solo se puede adaptar a las etapas iniciales de diseño donde el producto de trabajo se centra alrededor de un modelo CAD, con análisis, simulación y creación de prototipos virtuales. Las versiones entregan el modelo a un equipo de proyecto más grande para su revisión y comentarios, incluyendo fabricación, comercialización y calidad.

2.5.6.2 Ágil en un entorno de desarrollo integrado

Myllerup (79) afirma que una metodología ágil de estilo "Scrum" es fácilmente aplicable a los sistemas integrados de desarrollo basados en la experiencia y proporciona observaciones y sugerencias para garantizar el éxito. El desarrollo de sistemas integrados tradicionales es realizado por equipos funcionales (por ejemplo, equipo de "software", firmware), equipo de electrónica, equipo mecánico, etc. Una mejor alternativa es estructurar la organización en equipos de características completamente multifuncionales, con miembros de diferentes dominios. Los equipos se centran en el producto terminado, en lugar de hacerlo solamente en los objetivos relacionados con su profesión. Además, las habilidades y el conocimiento se mejorarán cuando se colabore en tareas fuera de la profesión. El autor proporciona dos formas de lidiar con los plazos de fabricación de circuitos impresos en tableros u otro "hardware ". Según lo sugerido por Maccherone (76), los prototipos del proyecto pueden dividirse en varios objetivos de "sprint", claros y medibles. Una segunda opción es trabajar para completar un subsistema (sub circuito) delineado por el diagrama de bloques a una etapa de prototipo comprobable encada iteración e integrado en el diseño final al mismo tiempo. Myllerup (79) también sugiere que el los criterios de aceptación (en relación con la calidad y el valor) sean consistentes en todas las disciplinas, incluso si la definición de hecho se adapta apropiadamente. "Scrum" normalmente solo incluye al propietario del producto en actividades de planificación dado un retraso, pero otras partes interesadas deberían participar para obtener el nivel suficiente de conocimiento y experiencia en todas las disciplinas. Finalmente, el autor sugiere algunos datos adicionales de planificación inicial a través de una serie de picos durante la iteración dirigido a la entrega de un conjunto de diagramas de bloques en bruto que delinea la arquitectura propuesta. Estos diagramas y la arquitectura serán refinarse en iteraciones posteriores, pero proporcionarán una base para el desarrollo de la acumulación inicial.

2.5.6.3 Zonas de aterrizaje

Wirfs-Brock (80) presenta las zonas de aterrizaje como un método alternativo de gestión de requisitos de encuadre. Este enfoque define un rango de atributos medibles para capturar un mínimo, objetivo y resultados sobresalientes. Cumplir con un conjunto mínimo proporcionará un producto viable mínimo que respalda la visión del producto. Los rangos estrechos pueden reflejar requisitos con baja incertidumbre o alta criticidad. El método ayuda a identificar las áreas de riesgo, mejorar las decisiones de compensación

multidimensionales y muestra los requisitos de tendencias hacia los objetivos a medida que avanza un proyecto. Dentro del contexto de las metodologías ágiles, los requisitos basados en la zona de aterrizaje pueden ayudar a evaluar la capacidad de liberación de un producto mediante el monitoreo de un subconjunto pequeño y significativo. Minimizando este grupo mejora la comprensión y la visibilidad incluso en proyectos grandes. Esto sigue el principio de visualización de Kanban para ayudar a entender el estado del proyecto. Si el grupo de atributos que representa la zona de aterrizaje es demasiado grande, puede ser beneficioso dividirlos en múltiples zonas de aterrizaje por categoría o área funcional.

Los objetivos iniciales deben reflejar los objetivos a corto plazo de un proyecto. En todo un proceso de desarrollo iterativo, los atributos, rangos y zonas de aterrizaje deberían revisarse con el resto del plan. Equilibrar la precisión y la flexibilidad de los atributos puede ser un desafío, y debe ser guiado por la visión del proyecto y el valor comercial. Aunque los requisitos sufrirán cambios, los objetivos concretos ayudan a orientar el desarrollo y proporcionan una base para la verificación cuantificable.

2.5.6.4 Visualización de diseños ágiles

Abrazar el cambio es un valor ágil clave. Habermellner (72) describe las características que hacen que un diseño sea ágil y fácil de cambiar Wirfs-Brock (80) resume el trabajo de Foote y Yoder (81), que se centraliza sobre el concepto de que algunas partes o capas de un sistema de "software" son mucho más fáciles de cambiar que otras, y también es más probable que experimenten cambios. Esta idea, a su vez, se tomó prestada de "Buildings Learn " (82), donde el concepto se aplicó a los edificios. Las "capas de corte" ilustran velocidad y facilidad de cambio de diferentes partes del sistema, como se muestra en la Figura 26. La adaptación al "software" basado en las propuestas de Foote y Yoder (81) se presenta en la Figura 27. Esta característica de los diseños es fácilmente extendido a muchos otros dominios también. Para mejorar la tolerancia de cambio de un sistema, debe diseñarse de manera que las capas que cambian rápidamente puedan modificarse sin ser obstruidas por las capas más lentas. Agrupar funciones y módulos que cambian a tasas similares, reducirá el esfuerzo requerido para el reproceso (aunque los cambios en las capas fundamentales inferiores requerirán un esfuerzo significativo). Esta idea respalda el objetivo de la modularidad y el acoplamiento reducido dentro de un diseño.

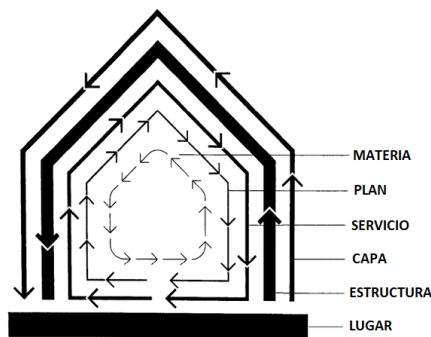


Figura 26. Capas de corte de Brand en edificios.
(Fuente: Donald Ryan (82))



Figura 27. Adaptación de capas de corte a sistemas de "software".
(Fuente: Modificación de Wirfs-Brock (80))

2.5.6.5 Aplicación de métodos "Scrum" al desarrollo de "hardware "

Backblaze Inc. (83) presenta algunas ideas sobre la aplicación de "Scrum" para el desarrollo de "hardware ". El ejemplo del proyecto "WikiSpeed" se describe como una aplicación exitosa de "Scrum" a un equipo multidisciplinario. El proyecto, en el que se desarrolló un prototipo de vehículo legal para el tráfico en aproximadamente tres meses con una cuidadosa adaptación de las prácticas de "software" ágil al diseño automotriz (84). Se describen los desafíos enfrentados al adoptar "Scrum" para el desarrollo de "hardware ", junto con sugerencias para adaptar el método "Scrum". El tiempo y el costo para crear un "hardware " en funcionamiento puede ser difícil de superar cuando se sigue un método ágil en el desarrollo de "hardware ", especialmente cuando el costo unitario es alto. Además, un producto físico a menudo es menos modular y más difícil de cambiar, lo que aumenta aún más el costo y el tiempo de futuras iteraciones y reelaboraciones. Esto resulta en una tendencia a recurrir a los métodos tradicionales en un esfuerzo por administrar el riesgo financiero. Los autores sugieren que esto se puede abordar adaptando la liberación frecuente de un producto funcional.

En lugar de entregar un prototipo funcional en cada iteración, una simulación virtual puede proporcionar un valor similar y oportunidad de retroalimentación mientras se reduce el tiempo y el costo requerido para la fabricación. Este producto también es fácil de modificar. Como alternativa, cuando la modularidad es suficiente, aislando subsistemas o componentes pueden ser entregados en cada iteración, con "sprints" adicionales utilizados para integración más cerca del final del proyecto. Si el proyecto o diseño tiene menos modularidad de lo deseado, se sugiere que el diseño se pueda cambiar para ajustarse mejor a un desarrollo con una metodología ágil.

2.5.6.6 Ayudando al "hardware " a ser ágil

Rothman (85) sugiere acercarse al desarrollo de "hardware" ágil usando un estilo de la metodología Kanban. Hay diferentes tipos de "hardware" presentes en los ciclos de desarrollo de productos típicos, incluyendo ingeniería mecánica, ingeniería eléctrica y electrónica, así como lógica programable (es decir, matrices de puertas programables en campo). Cada uno proporciona valor al proyecto en diferentes tiempos y tienen diferentes características de ciclo de desarrollo, fuertemente influenciadas por el costo de la encarnación física. El autor sugiere organizar el proyecto en equipos funcionales basados en el dominio de "hardware " o "software" apropiado, con cada equipo usando un tablero Kanban personalizado (que deben incluir interdependencias). El conjunto de tablas Kanban permite a cada equipo para entender tanto su propio progreso como el de los demás. La integración del equipo cruzado se planifica con la ayuda de una hoja de ruta general del proyecto que resume las prioridades de las funciones, las interdependencias y cada vez más demostraciones mensuales complejas, como entregables. Aumentando el número de puntos de integración y demostraciones, alienta la colaboración y la transferencia de conocimientos para hacer concesiones apropiadas y mejorar la comprensión del proyecto como un todo. La coordinación de entregas de demostración e integración apropiadas entre equipos puede ser un desafío, debido a la variación de tiempo y costo para crear prototipos. Los elementos de "software" son mucho más fáciles para iterar rápidamente en comparación con la electrónica y el "hardware" mecánico. Los tiempos de ciclo de iteración, deben estar sincronizado para garantizar que la retroalimentación se propague a través de todo el proyecto. Rothman (85) discute algunas maneras de reducir el tiempo y el costo de la creación de prototipos físicos, para que estos tiempos de ciclo puedan ser traídos en línea con los de los componentes del proyecto, más

suave. Diferentes tipos de modelado y simulación son sugeridos como métodos de bajo costo, mientras que se pueden diseñar prototipos, modelos o maquetas rentables para validar un subconjunto de parámetros de diseño (por ejemplo, tamaño, forma, operación del subsistema). Al administrar el riesgo de construir un prototipo, varias consideraciones pueden guiar el proceso de toma de decisiones. La forma física provisional debe proporcionar valor, especialmente cuando se considera el costo de esta forma y el intervalo de tiempo para el cual el prototipo es válido o útil. Si un prototipo físico que refleja el estado del diseño no proporciona valor. Un modelo o demostración virtual alternativa debe ser considerado para obtener la retroalimentación y la reducción de riesgos.

CAPÍTULO 3. PROPUESTA METODOLÓGICA

Propuestas sobre cómo abordar el proceso de diseño y cómo los métodos ágiles pueden ser aplicados más allá del "software", han sido expuestas en la sección anterior. Donde es posible identificar que algunos métodos del diseño presentan mayor tendencia en el campo del diseño mecánico. Por lo anterior, para el planteamiento de este nuevo método se tomarán como base los modelos prescriptivos que presenta Pahl & Beitz (23), y Ullman (34). Resaltando las modificaciones que se tienen que realizar. Debido a que dichos métodos seleccionados prescriben un patrón de actividades de diseño en secuencia de cascada, donde los pasos del proceso se llevan a cabo uno por uno de forma secuencial a lo largo del tiempo. Entonces, como métodos secuenciales, no es posible realizar un nuevo paso sin finalizar el anterior. Mostrando que el aspecto más problemático de estos métodos tradicionales es la obligación de definir todos los requisitos al comienzo del proyecto y la restricción rígida de no poder regresar en un paso previo en caso de falla. Resaltando que este tipo de procesos secuenciales no interactúan bien con los entornos que cambian rápidamente.

El método ágil seleccionado para esta nueva propuesta debe estar enfocado a los PDT, teniendo como característica principal el enfoque a los proyectos complejos. Donde el proyecto complejo se puede definir con base en la clasificación que presenta el marco de Cynefin (86). Tipos de dominios: simple, complicado, caótico, complejo y el quinto dominio, desorden, que se produce cuando no se sabe en qué otro dominio se encuentra (ver la Figura 28). Por lo anterior, esta propuesta se basará en el método de "Scrum" (55), debido a que es particularmente adecuado para operar en un dominio complejo. En donde la capacidad de sondear (explorar), detectar (inspeccionar) y responder (adaptarse) es crítica. Así como ofrecer funciones de trabajo integradas, probadas y valiosas para el desarrollo de proyectos complejos, donde en cada iteración conduce a resultados rápidos. Apoyando a las organizaciones a tener éxito en un mundo complejo, donde es necesaria una rápida adaptación basada en las acciones interconectadas de los clientes, usuarios, desarrolladores, patrocinadores y otras partes interesadas, como lo explica Rubin (55).

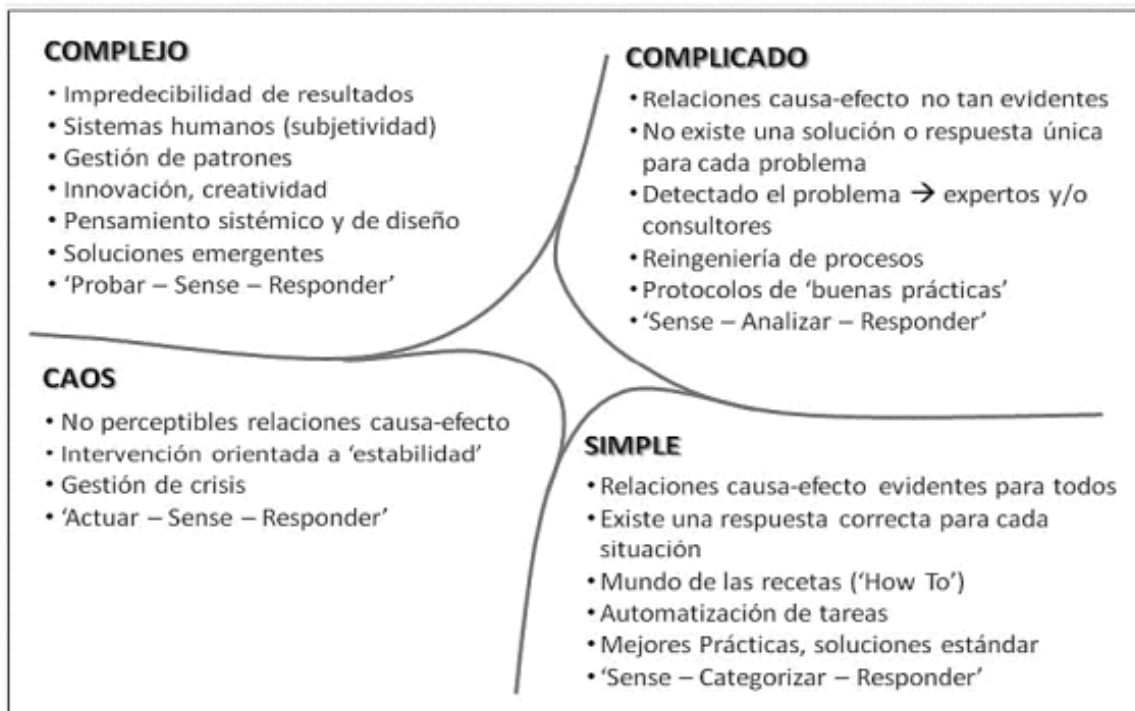


Figura 28. Marco de Cynefin.

(Fuente: Cynefin (86))

La metodología "Scrum" (55) debe ser usada en un proceso iterativo como se muestra en la Figura 29. Donde podemos resaltar que el propietario del producto tiene una visión de lo que quiere crear (el gran cubo). En caso de tener un cubo muy grande, se divide en un conjunto de características que se recopilan en una lista de prioridades denominada acumulación de productos. Una iteración comienza con su planificación, abarcando desde el trabajo a desarrollar durante la iteración hasta su revisión y retrospectiva.

Entonces, el nuevo método de diseño mecánico que se presenta en esta tesis es basado en los métodos de diseño que presentan Pahl & Beitz (23), Ullman (34) y "Scrum" (55) como metodología ágil. Buscando obtener productos ágiles y técnicamente fundamentados en los PDT.



Figura 29. Proceso de "Scrum".
 (Fuente: ESSENTIAL "Scrum" (55))

3.1 NUEVO MÉTODO PARA DISEÑO MECÁNICO

En este nuevo método, los fundamentos técnicos del diseño mecánico se acoplan con los principios ágiles, donde las funciones del producto a desarrollar serán el vínculo entre los principios de la metodología "Scrum" [40] y los métodos, herramientas y técnicas especializadas del diseño mecánico [27]. En la figura 30, se muestran los elementos principales de esta propuesta metodológica para el diseño mecánico ágil.

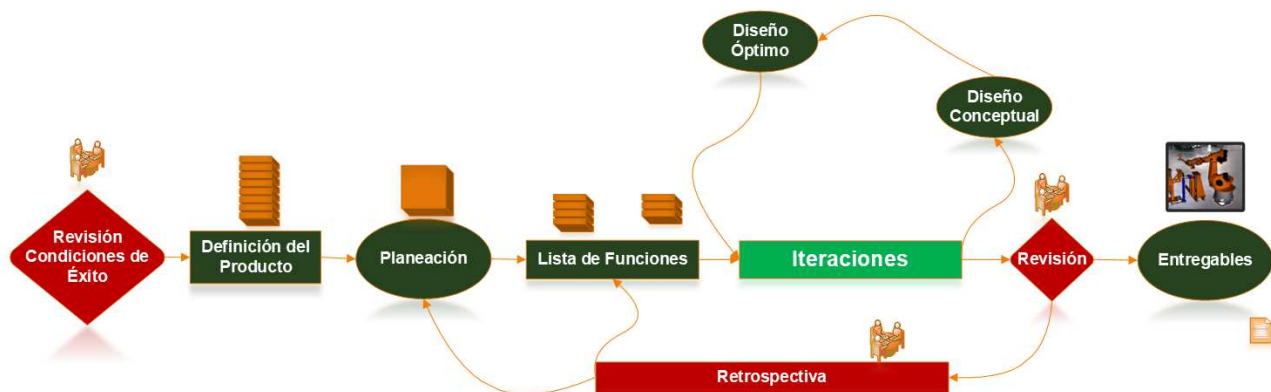


Figura 30. Nuevo método de diseño mecánico ágil.
 (Fuente: Elaboración propia)

La metodología "Scrum" inicia con la definición del producto, donde el dueño del producto es responsable de esta definición (55). Sin embargo, la incertidumbre al inicio del proyecto es una característica de los PDT. Por lo tanto, en este método, para obtener la definición del producto es necesario conocer las condiciones en que recibimos el proyecto. Entonces, en la primera etapa de este método se revisan las condiciones de éxito del proyecto, para proceder al análisis funcional que definirá al producto.

3.2 REVISIÓN DE CONDICIONES DE ÉXITO

Revisar las condiciones en las que se recibe el proyecto, será la base para establecer una estrategia adecuada en el desarrollo del diseño mecánico, así como lograr una planeación acertada, basada a las características específicas de cada PDT. Esta revisión utilizará toda la información generada por la organización, antes del inicio del diseño mecánico, frecuentemente, las fuentes de información puede ser: la cotización, evaluación de viabilidad, solicitud de cotización, lecciones aprendidas, entrevistas, entre otras. Para agilizar esta revisión, se propone utilizar una lista de verificación con cuestionamientos relevantes prestablecidos sobre el diseño mecánico. El contenido de preguntas en la lista de verificación para cada categoría serán definidas por la organización. Ejemplos de preguntas típicas para cada categoría se muestran en la tabla 4.

Tabla 4. Lista de verificación de las condiciones de éxito.
(Fuente: Elaboración propia)

ANÁLISIS DE CONDICIONES DE ÉXITO			
INDICADOR GENERAL			
RESPONSABLE			PROYECTO
Categoría	Descripción	Semáforo	Comentarios
Alcance	¿Se conocen los entregables del proyecto?, ¿Se conocen las restricciones del proyecto?, ¿Se conocen los criterios de aceptación del proyecto?		
Tiempo	¿El proyecto está estructurado bajo un enfoque ágil?, ¿Se conoce el tiempo asignado para el proyecto general?, ¿Se conoce el tiempo asignado para el diseño mecánico?		
Presupuesto	¿Se conocen las horas asignadas para el diseño mecánico?, ¿Se conoce el presupuesto de materiales asignados para el desarrollo mecánico?, ¿Se conoce el presupuesto para servicios asignados para el desarrollo mecánico?		
Producto	¿Se conoce la madurez del producto que interactúa con este desarrollo?, ¿Se tienen los modelos en ambientes virtuales, tales como CAD 3D?, ¿Se tienen los planos de especificaciones del producto, tales como tolerancias?		

El contenido de preguntas para cada categoría no es limitativo. La organización puede incluir o modificar las preguntas, así como las métricas de los indicadores correspondientes.

3.2 DEFINICIÓN DEL PRODUCTO

“Scrum” inicia con la creación de las características que definirán al producto, donde el dueño del producto debe tener una clara visión de lo que se debe obtener al final del proyecto. En este método se sugiere obtener las características del producto utilizando un nuevo proceso de análisis funcional. Donde el principio radica en obtener las expectativas del cliente al inicio del proyecto, en términos de funciones y los requisitos del producto.

En métodos de diseño tradicionales, el análisis funcional del diseño es dependiente de la previa identificación de necesidades a un nivel general, a través de comprender el mercado que impulsa la necesidad, para convertir posteriormente las necesidades a requisitos del cliente. Sin embargo, el objetivo del análisis funcional de esta propuesta es obtener la definición del producto a través de conocer o confirmar las expectativas del cliente en términos de funciones y requisitos del producto. Por lo tanto, los métodos tradicionales necesitan ser modificados para obtener dicha definición del producto, principalmente porque el proceso se invertirá, ya que los principales requisitos del producto se obtendrán a partir de las funciones del producto identificadas en términos de entradas y salidas de cada función, generando un balance y una fuerte relación entre las funciones y requisitos del producto (Figura 31). A continuación se expone este nuevo procedimiento para el análisis funcional que consta de seis pasos, fundamentados en los principios que expone Pahl & Beitz (23) y Ulwick (87).



Figura 31. Balance entre funciones y requerimientos.
(Fuente: Elaboración Propia)

Definición del producto. Análisis funcional, en seis pasos.

Paso 1. Declarar la secuencia de operación del producto esperada por el cliente.

La obtención de la secuencia de operación debe estar enfocada a las expectativas del cliente. Donde se busca identificar el resultado que espera el cliente y no su necesidad, como lo recomienda Ulwick (87) . Se recomienda obtener funciones a un nivel alto.

Para facilitar la obtención de esta secuencia de operación se recomienda estructurar un formato que simplifique y agilice la captura de todos los pasos de la secuencia de operación. Debido a que este análisis se realizará a un alto nivel, se recomienda usar un diagrama de bloques, ya que el objetivo principal será obtener todas relevantes las características que definirán al producto. En la tabla 5, se muestra un documento de referencia con una estructura predefinida.

Paso 2. Identificar y declarar todas las operaciones necesarias para lograr la funcionalidad del producto y las cuales no hayan sido expuestas por el cliente.

Obtener una secuencia de operación lógica y completa. En nivel de conocimiento del cliente será proporcionar al nivel de tareas que declara en la secuencia de operación. Entonces, el equipo de diseño debe completar dicha secuencia en caso de ser necesario y por supuesto al final de la obtención validarla con el cliente, buscando definir las características principales que espera el cliente como resultado del proyecto.

Paso 3. Convertir las operaciones identificadas en funciones.

Las tareas que se declaran en la secuencia de operación pueden estar en términos diferentes a la estructura que debe tener una función, en ese caso, se debe realizar una conversión para redactar claramente cada función que realizará el producto. Las funciones deben ser declaradas como verbos acompañados de un sujeto y un objeto.

Sujeto --- Verbo --- Objeto

Operador --- Carga --- Modulo

Cepillo dental --- Remueve --- Placa

Considerar que se pueden encontrar funciones activas y pasivas. Las funciones activas son fáciles de identificar, pues son aquellas acciones que lleva a cabo el sujeto. Ejemplo: La pluma marca el papel.

Las funciones pasivas son aquellas que permiten "hacer cosas". Para identificarlas es necesario observar las diferentes partes y características de cada elemento y preguntarse el por qué están ahí, la respuesta por lo general, es para llevar a cabo una función. Ejemplo: El tapón de la pluma restringe el paso del aire hacia la punta. ¿Por qué? El paso del aire seca la tinta. El tapón conserva la tinta es estado líquido.

Paso 4. Clasificar las funciones dentro de los siguientes rubros: básicas, mandatorios y/o innovación.

Métodos tradicionales sugieren identificar la función principal para todas las funciones del producto. Sin embargo, este proceso típicamente demanda mucho tiempo y esfuerzo para el equipo de diseño y en muchos proyectos se encuentran funciones ya desarrolladas o el caso contrario, se puede encontrar funciones que no han sido exploradas. Entonces, se propone una alternativa para identificar y clasificar las funciones más importantes del proyecto.

Se busca etiquetar las funciones que definen al producto en términos de los siguientes criterios:

Funciones Básicas: son aquellas funciones que se dan por hecho y frecuentemente no se declaran en las tareas de la secuencia de operación.

Funciones Mandatorios: son aquellas funciones donde el cliente y/o el equipo de trabajo ya no quieren indagar en otra solución.

Funciones de Innovación. No se tiene ningún dato en ese momento sobre una posible solución, para estas funciones.

Entonces, a cada función se le asignará una categoría basada en los criterios mencionados. Se propone utilizar una estructura estándar para etiquetar estas funciones como se muestra en la matriz de funciones de la tabla 5.

Paso 5. Las funciones clasificadas en “innovación” se deben convertir en funciones primarias.

Para la conversión, se aplicará en varias ocasiones una pregunta a cada función de innovación. ¿Para qué?, y nuevamente ¿Para qué?..., hasta llegar a respuestas redundantes de acuerdo al contexto del proyecto.

El nivel de la función quedará en definida por el contexto del proyecto. Y será identificada cuando se presenten respuestas redundantes. Tener precaución de no salir del contexto del proyecto ya que es posible perderse en la búsqueda de la función principal fundamental.

Paso 6. Declarar las entradas y salidas de cada función, en términos de Materia, energía o información.

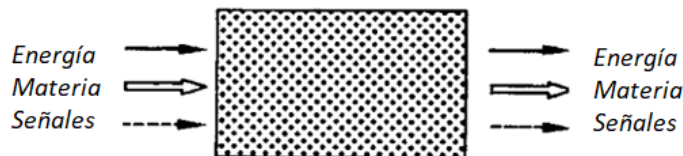


Figura 32. Diagrama básico de funciones.

(Fuente: Modificación Ullman (34))

El objetivo principal de las funciones es la conversión de la materia, energía o información principalmente, como se muestra en la figura 32 y se ejemplifica a continuación.

La energía puede ser convertida en una variedad de maneras, ejemplos: Mecánica, térmica, eléctrica, química, óptica, nuclear..., también fuerza, corriente, calor, etc.

Los materiales pueden ser convertidos en una variedad de maneras, ejemplos: Gas, líquido, sólido, polvo..., también materia prima, muestras para prueba, pieza de trabajo... producto final, componente, etc.

La información, cada planta debe procesar información en la forma de las señales. Ejemplos: Magnitud, visualización, impulso de control, datos, información, etc.

Las entradas y salidas de las funciones serán el vínculo para declarar las fronteras de cada función, en términos de requisitos, especificaciones y su método de validación. Para la

obtención de los requisitos el método está basado en los principios de QFD (88) y el resultado se debe declarar en una matriz de requisitos mostrada en la tabla 6. Posteriormente, se clasificarán en requisitos básicos, desempeño o deleite, como lo recomienda el modelo de KANO (89).

Tabla 5. Matriz de funciones.
(Fuente: Elaboración propia)

MATRIZ DE FUNCIONES			
ENTRADAS	FUNCIONES	SALIDAS	TIPO
Materia: Energía: Información:		Materia: Energía: Información:	<p>Básicas: son aquellas funciones que se dan por hecho y no se declaran en las tareas de la secuencia de operación principal.</p> <p>Mandatorios: son aquellas donde el cliente y/o el equipo de trabajo ya no quieren indagar en otra solución.</p> <p>Innovación. No se tiene información de posible solución.</p>
Materia: Energía: Información:		Materia: Energía: Información:	<p>Básica.</p> <p>Mandatorio.</p> <p>Innovación.</p>

Para la obtención de los requisitos del producto, los autores explican que la falla del proyecto es probable si el mercado no está bien entendido de una manera objetiva. Se presentan métodos para llevar a cabo estudios de mercado. Una herramienta adicional recomendada es el Modelo KANO (figura 33), para la evaluación de las necesidades del cliente, que busca clasifique los requisitos en cinco tipos: imprescindible, unidimensional, atractivo, indiferente o inverso. Esto se basa en la idea básica de que algunas necesidades son críticas, mientras que otras son simplemente deseables. Continuando con el tema de la comprensión del problema de una manera neutral de solución, los requisitos se usan para identificar funciones y crear una estructura funcional a través de la descomposición.

Tabla 6. Matriz de requisitos.
(Fuente: Elaboración propia)

MATRIZ DE REQUISITOS			
	REQUERIMIENTOS	ESPECIFICACIÓN	MÉTODO DE VALIDACIÓN
<p>BÁSICOS Son aquellos cuya ausencia provoca insatisfacción del cliente, aunque su presencia se dé por hecha y no se valore especialmente.</p>			
<p>DESEMPEÑO Características o atributos que aumentan la satisfacción del cliente de modo aproximadamente lineal con el aumento de su funcionalidad.</p>			
<p>DELEITE Hay atributos que la gente valora si están presentes, pero que no perciben si están ausentes, cuando esto ocurre estamos en presencia del deleite. Son características no esperadas que generan una gran satisfacción.</p>			

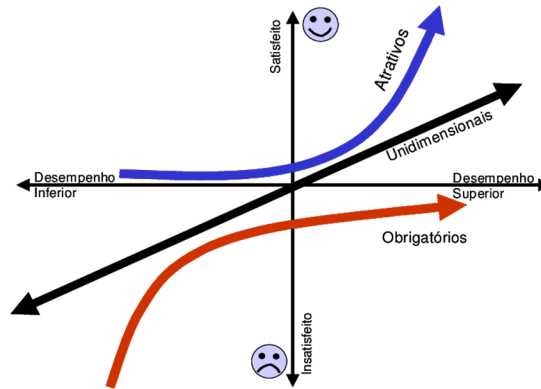


Figura 33. Esquema del modelo de KANO.
(Fuente: Verduyn (89))

Requisitos adicionales podrán ser expuestos y atendidos durante las iteraciones durante el desarrollo del proyecto, dichos requisitos podrían ser: requisitos legales, ergonómicos, entre otros. Durante el proceso de las iteraciones, se realizarán los análisis específicos para el diseño orientados a un campo temático específico y serán atendidos utilizando el método de diseño para la excelencia, donde se deberá aplicar el diseño para la seguridad, ergonomía, sustentabilidad, eco diseño, según la X considerada que agrega valor al diseño.

El entregable de esta etapa será la definición del producto, bajo una matriz de funciones y una lista de requisitos. Nuevas funciones y nuevos requisitos, serán documentados en relación a los incrementos necesarios para completar el diseño mecánico.

3.3 PLANEACIÓN DEL PRODUCTO

La planeación del producto usará todas las funciones declaradas en la etapa anterior como entrada y se deberán estructurar como se muestra en la tabla 7. Dichas funciones representan las características del producto que declaran los principios de "Scrum". El equipo de trabajo debe considerar al menos los tres roles, el dueño del producto, el coordinador del diseño mecánico y al equipo de trabajo.

En este método, el coordinador del diseño mecánico atenderá los dos primeros roles como se muestra en la figura 34. Debido a que las características del proyecto fueron determinadas en la etapa anterior. Por lo tanto, el entendimiento de los principios de "Scrum", no serán necesarios, ya que con este método el equipo de trabajo puede

entender perfectamente las actividades a realizar y cumplir con los principios y valores de "Scrum" de manera sistemática.



Figura 34. Equipo de diseño.
(Fuente: Modificación Kanneth (55))

El coordinador del diseño y el equipo de diseño mecánico, deben declarar el plan de trabajo, en términos de tiempo, prioridad y el diseñador que desarrollará cada función del producto. Al final de esta etapa obtendremos la duración total que demandará el diseño mecánico así como la estrategia de prioridades para cada función.

Para establecer dicha planeación se recomienda el uso del "póker" de Planificación que recomienda "Scrum". Donde, la clave está en reconocer que si los diseñadores comienzan a votar uno por uno, los primeros votos se influirán en el resto, y reducirá la calidad de las estimaciones. Para evitar este sesgo cognitivo se recomienda utilizar el póker de planificación. Cada miembro del equipo tiene una serie de cartas con unos valores, y cada uno debe elegir una carta que representa su estimación personal, basada en su opinión, y que mantiene boca abajo. Cuando todos han elegido, se muestran todas las cartas a la vez y entonces se comprueban los valores. Si, por ejemplo, alguien cree que el valor de la función es de 2 puntos, y otra cree que es de 20 puntos. Entonces, se puede estar seguros de que al menos uno de ellos no ha entendido bien la función. Así pues, deberá discutirse de nuevo y volver a votar.

Cuando todos los votos están en la misma línea, se calcula un promedio y eso sería el valor estimado para el desarrollo de cada función del Producto.

Tabla 7. Elementos básicos de una planeación.
(Fuente: Elaboración propia)

PLAN GENERAL			
FUNCIONES	TIEMPO	DISEÑADOR	PRIORIDAD

3.4 PLANEACIÓN DE LAS ITERACIONES

La magnitud de los proyectos, permitirá definir el nivel y número de iteraciones que se deben programar hasta completar el diseño mecánico del producto. El plan de iteraciones debe estructurarse como se muestra en la tabla 8.

Durante la planificación de cada iteración, el coordinador del diseño mecánico y el equipo de trabajo deben acordar un objetivo que define lo que se supone que logrará cada iteración. Con este objetivo, el equipo de diseño debe revisar las funciones de productos y determina las funciones de alta prioridad que el equipo puede realmente lograr en la iteración, mientras trabaja a un ritmo sostenible.

Para adquirir confianza en lo que se puede hacer, muchos equipos de trabajo desglosan cada función específica en un conjunto de tareas. La recopilación de estas tareas, junto con sus elementos asociados de la cartera de productos, forma una segunda cartera llamada la cartera de las iteraciones (ver la Figura 35 y 36).

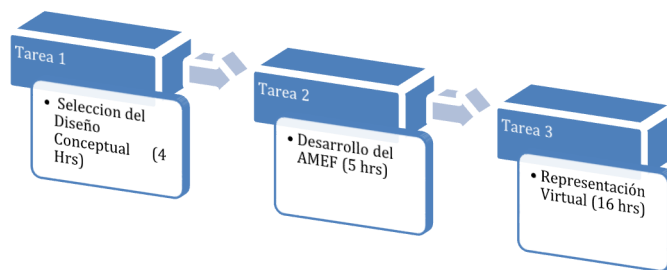


Figura 35. Desglose de tareas en cada función.
(Fuente: Elaboración propia)

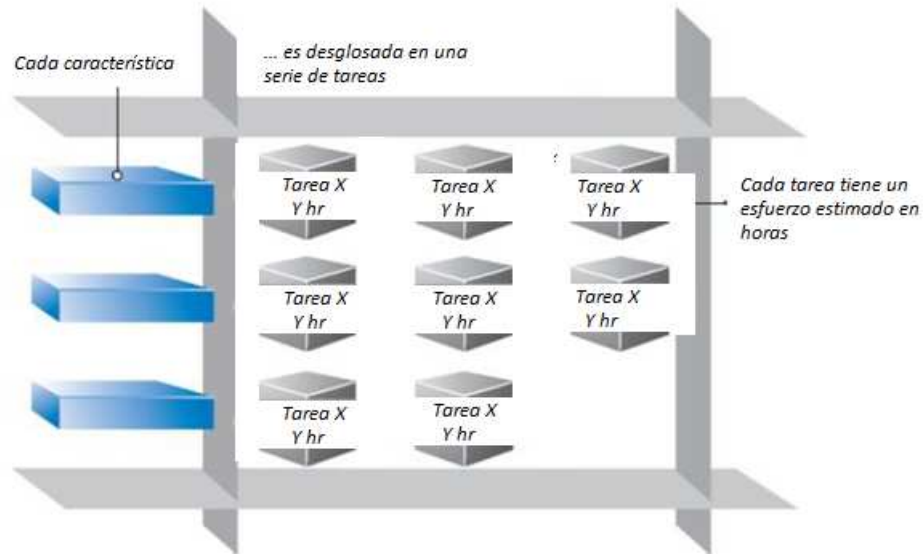


Figura 36. Cartera de Iteraciones.
(Fuente: Modificación Kanneth (55))

El equipo de diseño proporcionará una estimación del esfuerzo requerido para completar cada tarea. La división de los elementos de las funciones de los productos en tareas es una forma de diseño y planificación justo a tiempo sobre cómo realizar las funciones.

Para realizar iteración de dos semanas a un mes de duración, se deben completar la planificación de la iteración en aproximadamente cuatro a ocho horas. Una iteración de una semana no debería tomar más de un par de horas para planificar. Luego, el enfoque es seleccionar un elemento de la función de los productos, dividir el elemento en tareas y determinar si el elemento seleccionado se ajusta a la iteración. Sí se ajusta a la iteración y hay más capacidad para completar el trabajo, repita el ciclo hasta que el equipo no esté en capacidad para hacer más trabajo.

Tabla 8. Plan de iteraciones.
(Fuente: Elaboración propia)

PLAN DE ITERACIONES					
No	FUNCION	TAREA	DISEÑADOR	DURACIÓN (HRS)	FECHA DE FIN

3.5 ITERACIONES

Una vez que finaliza la planificación de las iteraciones y el equipo está de acuerdo con el contenido de la próxima iteración, el equipo de diseño mecánico, guiado por el coordinador del diseño, realiza todo el trabajo al nivel de tareas necesarias, hasta completar las funciones (ver Figura 37), donde la calidad del producto, dependerá de un proceso sistemático descrito más adelante.

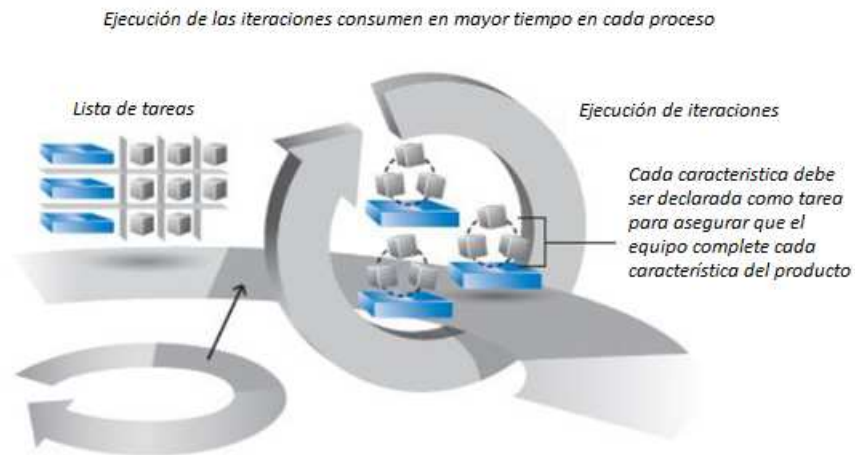


Figura 37. Iteraciones.

(Fuente: Modificación Kanneth (55))

Exactamente qué tareas debe realizar el equipo de diseño, depende por supuesto, de la naturaleza del trabajo. Sin embargo, la filosofía del método es tener un producto potencial para construirse en cada iteración. Por lo tanto, en cada iteración el equipo de diseño deberá realizar un diseño conceptual y un diseño óptimo, con el objetivo de potencializar la solución y pueda ser un entregable. Dichos pasos se definen con los siguientes procesos.

Se recomienda tener reuniones diarias para verificar los avances y estado de desarrollo de cada iteración. Realizar estas reuniones en un máximo de 15 min al día. Considerando al menos obtener respuestas a las siguientes preguntas.

- ¿Qué se logró desde la última reunión diaria?
- ¿En qué se planea trabajar para hoy?
- ¿Cuáles son los impedimentos u obstáculos que impiden avanzar?

El diseño por definición es un proceso complejo, donde muchos principios técnicos se desarrollan a partir de los requerimientos y las especificaciones a un alto nivel de certidumbre. En el diseño mecánico, la complejidad técnica es proporcional a la complejidad de la solución. Entonces, en este método, la solución en las primeras iteraciones debe ser básica, pero funcional. En la figura 38 se muestra el desarrollo de una solución bajo un concepto ágil, el cual será usado para desarrollar las dos etapas: diseño conceptual y diseño ágil para una solución básica-funcional.

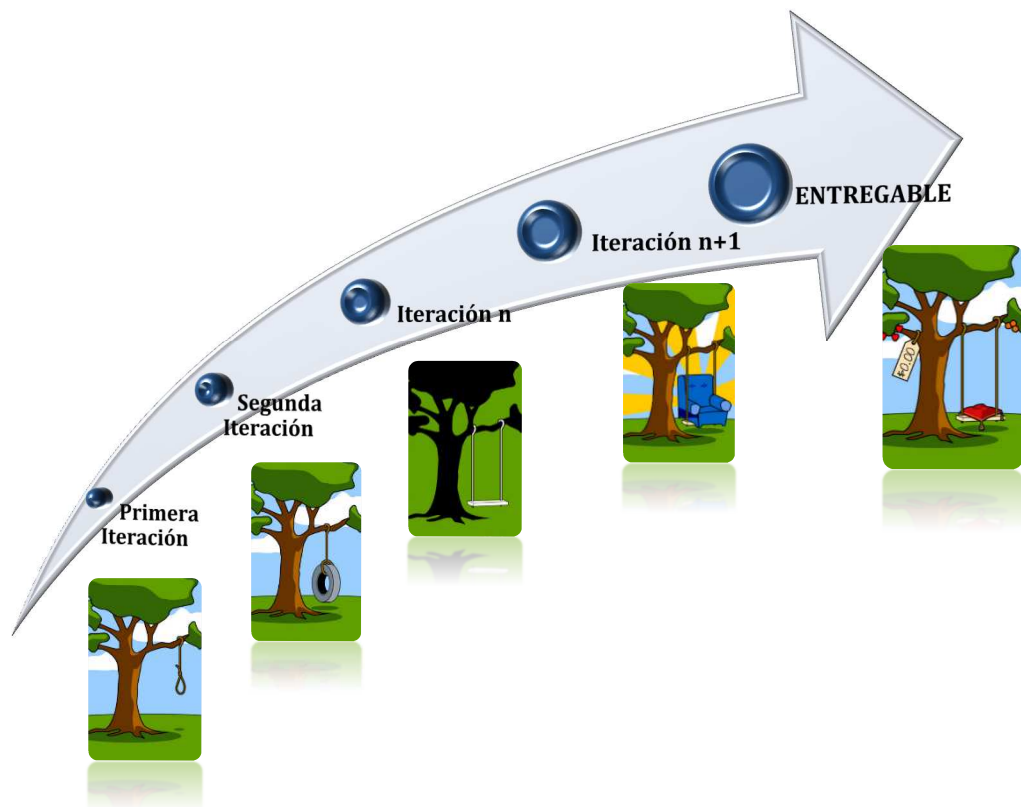


Figura 38. Incrementos en cada iteración. (Fuente: Elaboración propia)

3.5.1 Diseño conceptual

El diseño conceptual, es la fase donde la ruta de una solución básica se establece a través de la elaboración de un principio de solución. El diseño conceptual especifica la solución básica principal. En la figura 39 se muestran las etapas que se deben realizar para obtener el diseño conceptual del producto mecánico.

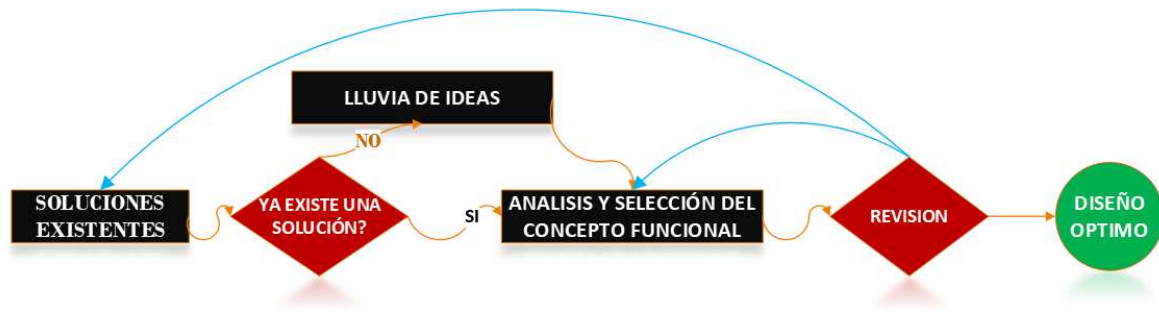


Figura 39. Proceso del diseño mecánico conceptual.
(Fuente: Elaboración propia)

3.5.1.1 Soluciones existentes

Como lo observan Haik y Shahin (51), el proceso de diseño debe guiar a un diseñador para lograr los objetivos, pero no obstaculizar la creatividad. Entonces, en esta sección la búsqueda de soluciones se realizará en varias etapas, con el objetivo de que la creatividad del diseñador sea aplicada específicamente en las funciones que realmente agregan valor. La filosofía es obtener diseño ágiles sin omitir la confiabilidad técnica del producto. En el caso contrario, se pueden encontrar soluciones ya desarrolladas que podrían únicamente ser integradas en el diseño y reducir riesgos técnicos, así como reducir el tiempo de ejecución de la iteración. Antes de iniciar el desarrollo de cualquier posible solución, El diseñador debe realizar una investigación en los niveles que se muestran en la figura 40.

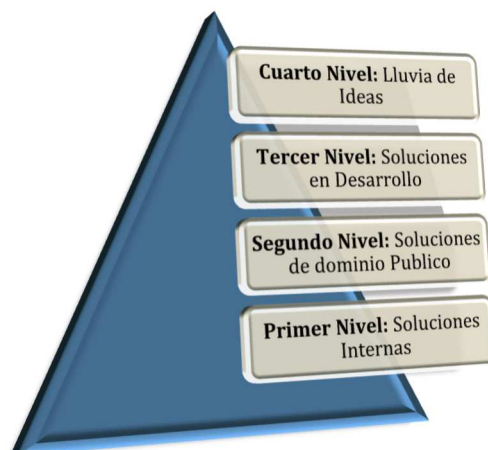


Figura 40. Niveles de búsqueda de soluciones.
(Fuente: Elaboración propia)

Primer nivel: Búsqueda en equipos desarrollados anteriormente en la empresa. Se refiere a realizar una investigación en proyectos históricos, buscando soluciones similares.

Segundo nivel: Soluciones de dominio público, son una fuente de información muy extensa e importante. Frecuentemente se encuentran soluciones en videos, hojas de datos, páginas de competidores, entre otros. En caso de seleccionar alguna solución en este nivel, es fundamental confirmar que sean soluciones de dominio público, es decir que no estén patentada o bajo algún dominio jurídico. Instituciones como el IMPI u OMPI pueden apoyarnos en esta investigación.

Tercer nivel: Búsqueda en desarrollos científicos, tales como patentes, artículos.

Búsqueda en desarrollos científicos, tales como patentes, artículos, etc. Para este nivel de búsqueda se deben utilizar herramientas de vigilancia tecnológica tales como bases de datos, motores de búsqueda, entre otros. En caso de que no se identifique una solución, el diseñador debe plantear una lluvia de ideas como cuarto nivel de búsqueda usando la propuesta de Cross (10 págs. 48-56). Llegar al tercer nivel de búsqueda, vislumbra una alta probabilidad de innovar. Para atender estas funciones de innovación se propone utilizar los principios de la metodología TRIZ (90), buscando dar una solución a un problema de inventiva.

Lluvia de ideas

Los pasos principales de la lluvia de ideas se muestran a continuación.

1.- Definir el objetivo que queremos conseguir.

Por ejemplo: ¿De qué formas se puede realizar la función?

2.- Definir los participantes, la duración de la reunión y recopilar el material.

Escribir las ideas y pegarlas en un panel

3.- Introducción al objetivo.

El dinamizador de la reunión informará sobre el objetivo a conseguir, poniendo sobre la mesa antecedentes y otros conocimientos para entender de dónde partimos y qué es lo que queremos conseguir.

4.- Ciclo de generación de ideas:

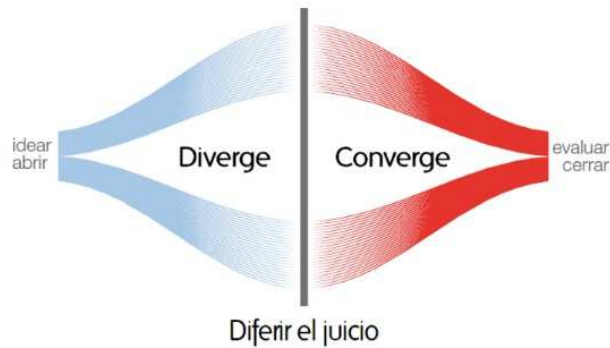


Figura 41. Divergencia y convergencia de ideas.

(Fuente: Modificación, NORMAN (13))

El objetivo durante el proceso de divergencia es generar todo tipo de ideas que vayan pasando por la cabeza hasta que se tenga la sensación de que se han agotado todas las vías posibles, como se muestra en la figura 41.

Durante la sesión, el responsable de la reunión o dinamizador irá anotando en la pizarra las ideas generadas por los participantes. El objetivo es investigar todos los puntos de vista posibles en diversos ángulos, así como dejar volar la imaginación y no poner límites o restricciones de ningún tipo. Se ve que cuando una persona propone una idea, da pie al que el resto de participantes construyan sobre esa idea, es decir, a partir de la idea de alguna persona se puede ocurrir otra idea al resto de los miembros del equipo, relacionada o no, con lo que él está comentando.

Cuando parezca que ya se han terminado las ideas, aún se dará un último empujón con el objetivo de conseguir más opciones o ideas cuando parece que ya no queda ninguna.

5.- Selección de ideas:

El hecho de haber estado "aplazando el juicio" permite que en el proceso de divergencia se haya sido mucho más creativos que de costumbre. Se deben seleccionar, razonablemente, las ideas que cumplan con el objetivo.

3.5.1.2 Análisis y selección del concepto

Este análisis y selección del concepto, se basa en el esquema de clasificación al que Zwicky (91) se refiere como la "matriz morfológica" (ver tabla 9). Aquí, las funciones y las soluciones apropiadas (principios de solución) se ingresan en las filas de la tabla 9.

Al menos se deben tener dos soluciones para cada función. Para proporcionar la solución global, las soluciones deben ser combinadas sistemáticamente en una solución global. Si hay principios de la solución 1 para la función F1, m2 para la función F2, etc., luego de completar las combinaciones, se debe obtener $N = m_1 \cdot m_2 \cdot m_3 \cdot \dots \cdot m_n$, como una solución global teóricamente posible de las variantes.

El riesgo en diseño radica cuando se cree que se tiene la mejor opción, sin antes someterla a una evaluación. Se debe concentrar en combinaciones prometedoras y establecer por qué deberían preferirse por encima del resto. Los criterios para evaluar cada solución se muestran en la tabla 9 y se describen a continuación.

Tabla 9. Matriz de análisis y selección del diseño conceptual.
(Fuente: Elaboración propia)

MATRIZ DE ANÁLISIS Y SELECCIÓN DEL DISEÑO CONCEPTUAL						
				CONCEPTO 1	CONCEPTO 2	CONCEPTO 3
FUNCIÓN	CRITERIO 1	PESO 1	REF			
	CRITERIO 2	PESO 2	REF			
	CRITERIO N	PESO N	REF			
SUMA PONDERADA						

El criterio se refiere a la descripción de la frontera de la función, puede ser; tiempo, costo, calidad, entre otros que considere la empresa.

El peso se refiere a la importancias que tiene el criterio, se lo asigna el cliente y/o equipo de trabajo del proyecto.

La referencia se refiere al valor objetivo del criterio, por ejemplo; sí el criterio es costo, la referencia debe ser un monto cotizado de +/- \$\$\$\$. Para lograr una comparación ponderada.

La escala de ponderación se refiere a la evaluación que realicemos para cada concepto. Dónde; +1 (Mejor), -1 (Peor) y 0 (Igual).

La suma ponderada será la suma de la ponderación multiplicada por el peso de cada criterio y definirá el concepto ganador.

3.5.2 Revisión del concepto

Antes de proceder al desarrollo del diseño óptimo, es recomendable realizar una revisión informal con todos los interesados del proyecto. No se requiere de una validación oficial del concepto, debido a que el diseño puede modificarse durante el proceso de diseño óptimo y generar expectativas incorrectas. El objetivo de esta revisión es sólo identificar las áreas de oportunidad del concepto seleccionado.

3.5.3 Diseño óptimo

Cada solución que resulte de las iteraciones debe ser potencialmente apta para ser construida. Entonces, el diseño óptimo se refiere a la evaluación y optimización de nuestro diseño, bajo criterios técnicos, usando principios de ingeniería y metodologías específicas para el diseño mecánico, como se muestra en la figura 42. Esta etapa será el principal diferenciador entre otras propuestas metodológicas de aplicación de "Scrum", ya que el principio es lograr un diseño ágil y fundamentado técnicamente.

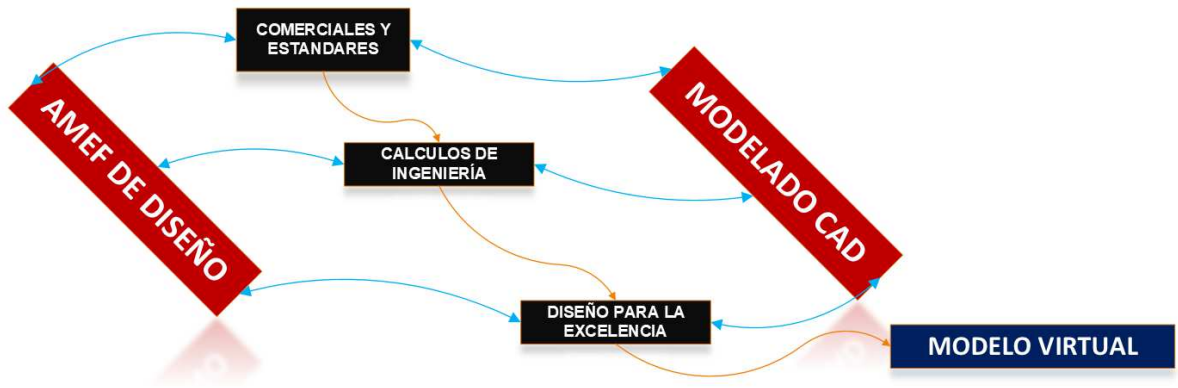


Figura 42. Estructura del diseño mecánico ágil.
(Fuente: Elaboración propia)

3.5.3.1 Sistemas comerciales y estándares

Dentro de las soluciones seleccionadas en el diseño conceptual, existirán sistemas que se pueden obtener comercialmente o se adapten a un sistema estándar, buscando no invertir tiempo y minimizar el riesgo técnico en el diseño, utilizando equipos y/o elementos que ya fueron probados y es posible únicamente integrar a nuestro diseño, tales pueden ser: engranajes, actuadores, válvulas, entre otros.

El diseñador puede explorar toda la tecnología disponible o la combinación de estas tecnologías comerciales o estandarizadas. Tales como: sistemas de conversión de energía, transmisión de energía, almacenamiento de energía hasta sistemas de localización, entre otros. Como lo recomienda el método de base tecnológica expuesto por Tooley(2009, pág. 470).

3.5.3.2 Cálculos de ingeniería

En esta etapa deben realizarse todos los cálculos y simulaciones necesarias que demande la temática del proyecto. Se utilizarán todas las herramientas ingenieriles disponibles. El AMEF de diseño proporcionará los sistemas que requieran mayor atención. Para el desarrollo del AMEF del diseño se puede basar en la propuesta de Anleitner (92). El análisis de modos de falla debe desarrollarse simultáneamente durante todas las etapas del diseño óptimo, como se muestra en la figura 46. Análisis, cálculos o simulaciones adicionales a los que detone el AMEF es recomendable realizarlos en esta etapa del proceso, ya que la configuración del producto deberá estar casi concluida al término de esta etapa.

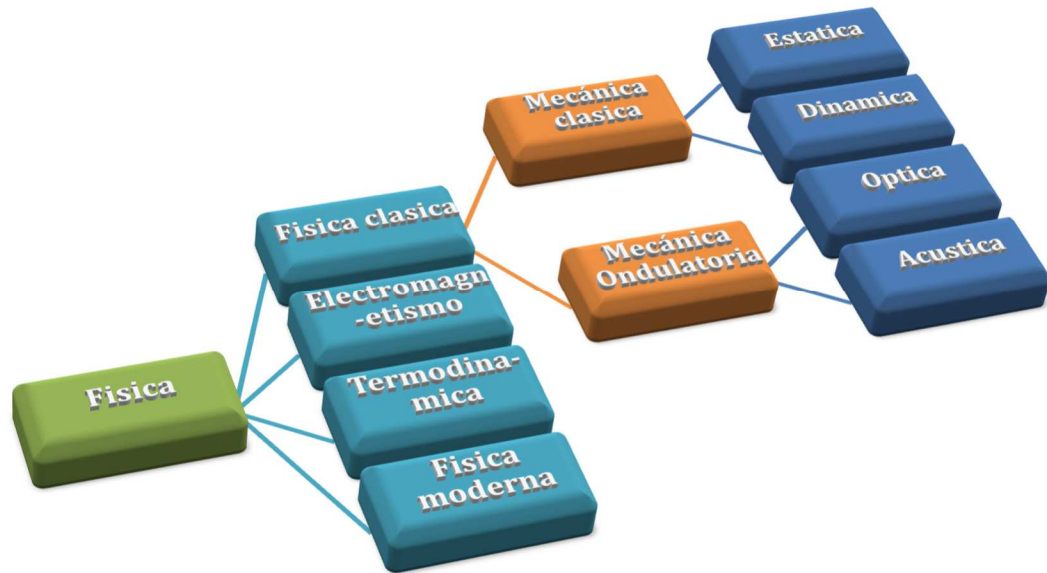


Figura 43. Estructura de la mecánica clásica y ondulatoria.
(Fuente: Elaboración propia)

Los temas del diseño mecánico son variados, por lo que esta propuesta no tiene como objetivo declarar herramientas de cálculos específicos, sino apoyarse en herramientas y métodos técnicos ya desarrollados, para utilizarlos en esta etapa y lograr un diseño técnicamente fundamentado. En la figura 43 se ilustran las temáticas principales relacionadas con la mecánica.

3.5.3.3 Diseño para la excelencia

Hasta esta etapa del proceso, la funcionalidad ha representado el 100% de las características desarrolladas del producto y todos los esfuerzos se han centrado en dicha funcionalidad. Sin embargo, en la actualidad se observa que los productos van mas allá de la funcionalidad incluyendo aspectos como; el costo, servicio, medio ambiente, manufactura, ensamble, entre otros.

El diseño para X se refiere al uso de una metodología formal para optimizar un aspecto específico de un diseño. La variable X representa las áreas de enfoque y de acuerdo a la aportación de Tichem, Tichem y Tormenta, Huang y Mak (93). se pueden dividir en dos partes:

1. Fases del ciclo de vida (por ejemplo, fabricación, entorno, pruebas, entre otros).
2. Características que se deben optimizar (por ejemplo, costo, calidad, velocidad).

Las propias directrices de diseño (ver figura 44) suelen proponer un método que pueda ayudar a generar y aplicar el conocimiento técnico para controlar, mejorar, o incluso inventar características particulares de un producto. Entre las metodologías más usadas se encuentran: el diseño para la manufactura y ensamble. Nuevos requerimientos serán expuestos a través de asignar la variable X, y estos requisitos deben ser documentados en la matriz de requisitos mostrada en la tabla 6.



Figura 44. Diseño para la excelencia.
(Fuente: Elaboración propia)

3.5.3.4 AMEF de diseño

En el proceso de diseño, toman relevancia los tres tipos de AMEF; de sistema, diseño y proceso, teniendo relación durante todo el desarrollo del producto, ilustrada en la figura 45.

AMEF DE SISTEMA (SAMEF) – Asegura la compatibilidad de los componentes del sistema. Se recomienda su uso posterior a la definición de las funciones del producto y antes de seleccionar el “hardware” específico.

AMEF DE DISEÑO (DAMEF) – Reduce los riesgos por errores en el diseño. Se utiliza para analizar componentes de diseños. Se enfoca hacia los modelos de falla asociados con la

funcionalidad de un componente u operación, causados por el diseño. Se recomienda su uso posterior a seleccionar el “hardware” del producto basado en sus funciones y antes de que el diseño sea aprobado.

AMEF DE PROCESO (PAMEF) – Revisa los procesos para encontrar posibles fuentes de error. Se utiliza para analizar los procesos de instalación, fabricación y ensamble. Se enfoca a la incapacidad para producir el requerimiento que se pretende, o un defecto. Los modos de falla se pueden derivar de causas identificadas en el AMEF de Diseño. Se recomienda su uso cuando los dibujos preliminares de producto y sus especificaciones están disponibles.

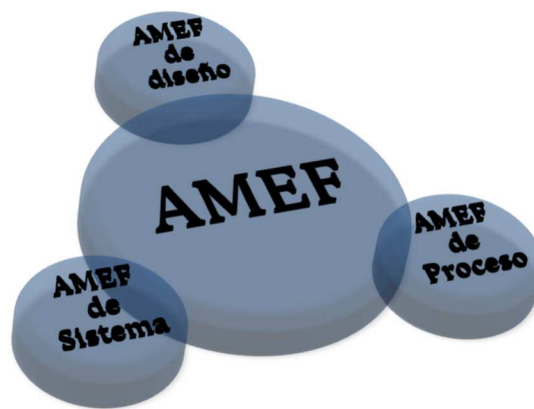


Figura 45. Tipos de AMEF.
(Fuente: Elaboración propia)

Durante todo el proceso de las iteraciones es factible el uso del AMEF. Incluso, para la selección de los conceptos se puede utilizar el AMEF del sistema. Sin embargo, este método sugiere el uso del AMEF a partir del diseño óptimo. Donde se evalúa la funcionalidad de todos los elementos del producto ya definidos, como lo expone Anleitner (92), buscando identificar los riesgos potenciales del diseño mecánico. El AMEF del diseño expondrá las funciones críticas que requieran análisis más detallados o simulaciones más complejas. Al mismo tiempo, permitirá reducir el tiempo de análisis para funciones que no representan un riesgo potencial. Para la ejecución ágil de este proceso, el método recomienda usar las bases de datos de lecciones aprendidas, declaradas en el proceso de retrospectiva de proyectos anteriores, con el objetivo de aprovechar información de acciones ya

comprobadas de otras funciones similares. Los elementos principales del proceso de desarrollo del AMEF se muestran en la figura 46.

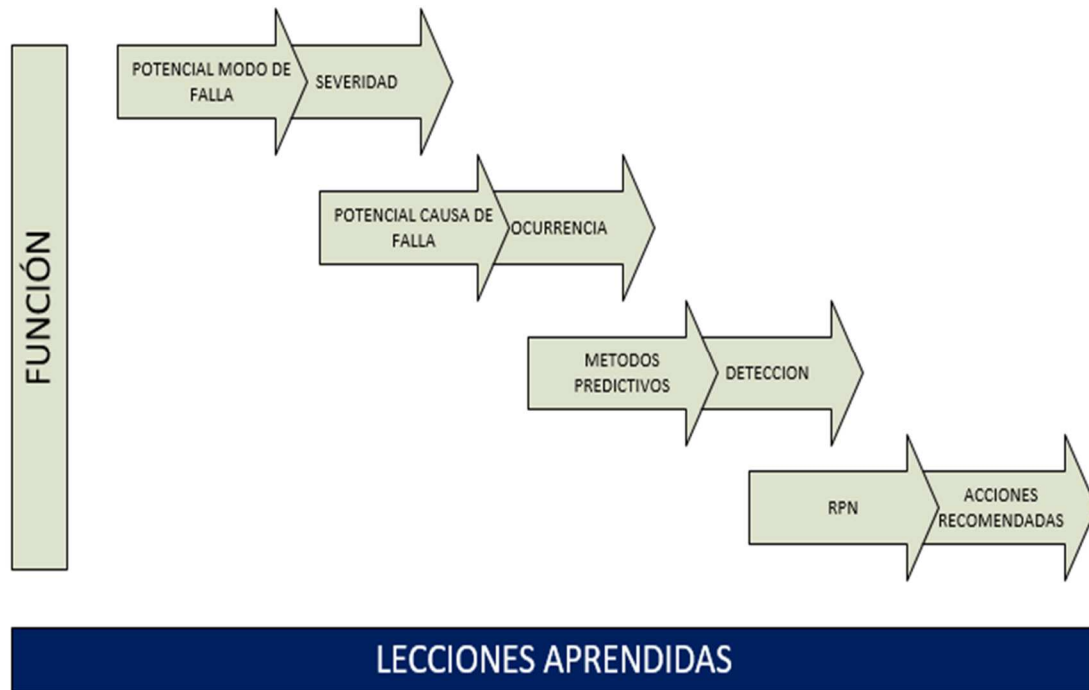


Figura 46. Proceso estándar para desarrollo del AMEF de diseño.
(Fuente: Elaboración propia)

3.5.3.5 Representación virtual

La representación virtual se divide en dos etapas: en *la primera etapa* se generarán los modelos 3D, simulaciones y otras representaciones del diseño. Mostrando en un ordenador la evolución del diseño, desde el concepto hasta el diseño óptimo. Como referencia se hace mención algunos de los “softwares” de modelado 3D más utilizados en la industria: SolidWorks, Catia, inventor, entre otros. Algunos de los “softwares” para el diseño se muestran en la figura 47.

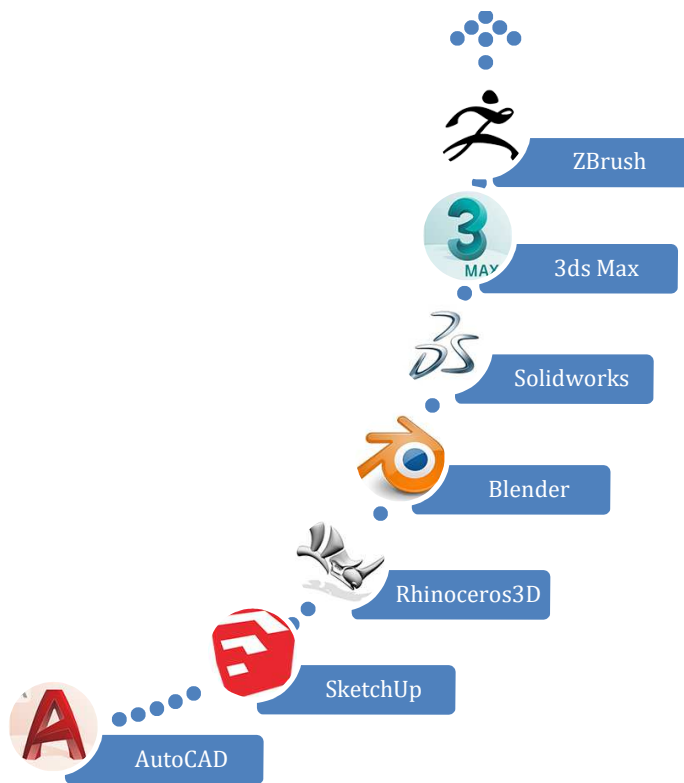


Figura 47. "software"s más importantes para modelado 3D.
(Fuente: Elaboración propia)

La segunda etapa es la representación en realidad aumentada, utilizando los modelos 3D y/o simulaciones, como objetos de entrada a los dispositivos de realidad aumentada, como lo expone en su propuesta Park (94) y Ong & Huang (95). Los dispositivos de realidad virtual con mayor uso en el diseño mecánico son: HoloLens (Microsoft), Atheer One, CastAR, Sony Smart EyeGlass, Vuzix M300 Smart Glasses, Recon Jet, ORA-1 Smart Glasses (optinvent). Algunas representaciones de los modelos en realidad aumentada se muestran en la figuras 48 y 49.



Figura 48. Demostración de una aplicación de los Hololens de Microsoft.
(Fuente: Hololens Microsoft (96))



Figura 49. Realidad aumentada en simulación de fluidos.
(Fuente: Hololens Microsoft (96))

3.6 VALIDACIÓN DEL CLIENTE

Esta validación se realizará usando el diseño mostrado en realidad aumentada, utilizado como fundamento técnico, al menos la matriz de funciones, la matriz de requisitos y el AMEF de diseño del producto. Lo anterior es debido a que las funciones fueron el vínculo de estudio durante el proceso de diseño mecánico y los requerimientos colocaron las fronteras de cada función para que finalmente el AMEF del diseño presente todas las protecciones que atiendan los modos de falla potenciales identificados en el diseño.

Como resultado de esta revisión, se pueden obtener algunos de los siguientes escenarios:

- Cambios parciales de la función. Lo cual, genera una nueva iteración y se debe planear la tarea en la siguiente lista de iteraciones.
- Cambios radicales en las funciones. Genera un nuevo análisis de expectativas, por lo que se requiere regresar hasta la etapa de planeación.
- Aceptación del diseño mecánico. El cliente firma de conformidad, y se procede a generar los entregables del diseño mecánico

3.7 RETROSPECTIVA

Como lo establece la metodología "Scrum" (55), se realizará una etapa de retrospectiva con el equipo del diseño mecánico, para generar y analizar las lecciones aprendidas. Con el objetivo de usar esta información en las siguientes iteraciones y proyectos.

3.8 ENTREGABLES

Todos los entregables de este proceso deberán ser utilizados en las siguientes etapas de ciclo de vida del producto, como lo propone Ullman (34 págs. 366-370). Los principales documentos que destacan como entregables son: Documentación para el aseguramiento y control de calidad de producto, Instrucciones de manufactura e instrucciones de ensamble, instalación, operación y mantenimiento. Para esta propuesta metodológica, se proponen los siguientes entregables: Planos de partes y ensambles, listas de materiales Comerciales y manufacturados, Manuales de mantenimiento y el AMEF de diseño. La organización definirá los entregables necesarios en función al ciclo de vida específico del producto en cada sector empresarial.

CAPITULO 4. RESULTADOS

4.1 CASO DE ESTUDIO

Este método se aplicó al proyecto denominado: Banco de pruebas dimensionales y de conductividad eléctrica para controladores de motores. Como referencia, en la figura 30 se muestra el producto del cliente.



Figura 50. Representación de un controlador de motores.
(Fuente: Elaboración propia)

4.1.1 Revisión de condiciones de éxito

El resultado de la evaluación de las condiciones de éxito del proyecto fue de un 66.65% de un máximo de 100%, el cual se obtendría en el caso de responder a todas las preguntas mostradas en la tabla 3. Cada categoría contiene tres preguntas que representan un valor máximo de 25% por cada categoría. En este caso de estudio se observa un valor de 8.33% para la categoría del alcance, correspondiente al 33.33% de un máximo de 25%, debido a que no se conocían las restricciones, ni el criterio de aceptación del proyecto durante esta evaluación. Se cumplió el objetivo de esta etapa del método, identificando y declarando las condiciones iniciales del proyecto para generar un plan de acción que atienda cada criterio faltante. Al darse a conocer el resultado de esta evaluación a los interesados en el proyecto, además de generar el plan de acción, se generó una lección aprendida para el departamento de estimaciones. Resaltando los criterios que no fueron atendidos antes del inicio del proyecto.

Tabla 10. Análisis de condiciones de éxito, caso de estudio.
(Fuente: Elaboración propia)

ANÁLISIS DE CONDICIONES DE ÉXITO			
INDICADOR GENERAL			66.6 %
Categoría	Descripción	Semáforo	Comentarios
Alcance	<p>¿Se conocen los entregables del proyecto?</p> <p>¿Se conocen las restricciones del proyecto?</p> <p>¿Se conocen los criterios de aceptación del proyecto?</p>	8.333 %	<p>Si, En la cotización se declara el entregable principal.</p> <p>No se tiene un documento enviado por el cliente donde se establezcan las restricciones del proyecto.</p> <p>No se conocen los criterios de aceptación del proyecto</p>
Tiempo	<p>¿El proyecto está estructurado bajo un enfoque ágil?, ¿Se conoce el tiempo asignado para el proyecto general?, ¿Se conoce el tiempo asignado para el diseño mecánico?</p>	16.667 %	<p>No, Se tiene una estructura tradicional</p> <p>Si, 8 semanas para todo el proyecto</p> <p>Si. 3 semanas para el diseño mecánico.</p>
Presupuesto	<p>¿Se conocen las horas asignadas para el diseño mecánico?, ¿Se conoce el presupuesto de materiales asignados para el desarrollo mecánico?, ¿Se conoce el presupuesto para servicios asignados para el desarrollo mecánico?</p>	25%	<p>Si, se tienen 90 horas</p> <p>Si, se tienen 10,000 USD</p> <p>Si, Se tienen 1,000 USD para la certificación dimensional.</p>
Producto	<p>¿Se conoce la madurez del producto que interactúa con este desarrollo?, ¿Se tienen los modelos en ambientes virtuales, tales como CAD 3D?,</p> <p>¿Se tienen los planos de especificaciones del producto, tales como tolerancias?</p>	16.667 %	<p>Si, el controlador ya está en producción.</p> <p>Si, Se entregaron todas las versiones de los controladores en CAD 3D.</p> <p>No, Los diseño se realizan en otro país y la empresa no puede obtenerlos.</p>

4.1.2 Expectativas del cliente

Siguiendo los pasos del método obtuvimos el siguiente resultado:

Paso 1; Secuencia de Operación.

1. Operador coloca el controlador en el banco de pruebas.
2. Operador inspecciona que las holguras estén dentro de las tolerancias del diseño del controlador.
3. Operador activa la cerradura.
4. Operador manualmente conecta las terminales para medir continuidad eléctrica.
5. Operador verifica que un led se encienda confirmando que existe continuidad.
6. Operador manualmente desconecta las terminales eléctricas.
7. Operador retira el controlador del banco de pruebas.

Paso 2; Identificar todas las operaciones necesarias y no expuestas por el cliente para lograr una secuencia de operación lógica. En este paso no se identificó ninguna operación adicional.

Paso 3-4; Declarar las operaciones como funciones, mostradas en la tabla 11.

Tabla 11. Matriz de funciones, caso de estudio.
(Fuente: Elaboración propia)

FUNCIONES			
ENTRADAS	FUNCIONES	SALIDAS	TIPO
	Cargar Controlador		Básica
	Inspeccionar Controlador		Innovación
	Activar Cerradura		Mandatorio
	Conectar Controlador		Innovación
	Inspeccionar Continuidad		Mandatorio
	Desconectar Terminales		Básica
	Descargar Controlador		Básica

Paso 5; Convertir las funciones de innovación en funciones primarias (Inspeccionar y Conectar)

- *Inspeccionar Controlador.*

¿Para Qué? Para confirmar que las dimensiones cumplen con la especificación de los controladores.

- *Conectar las terminales.*

¿Para Qué? Para generar continuidad.

¿Para Qué? Para confirmar que se realizaron las conexiones internas del controlador.

Posterior a realizar los cuestionamientos sugeridos por el método, se observa que la respuesta se está saliendo de las fronteras del proyecto, ya que se requiere tener solo un control interno de controlador y las características ya definidas del controlador. Por lo tanto, se considera inspeccionar y conectar como funciones primaria.

Paso 6; Declarar las entradas, salidas de las funciones y los requisitos, para lograr obtener las fronteras de las funciones como requisitos, especificaciones y su método de validación. Para este caso de estudio se muestra el resultado de las funciones en la tabla 12 y requisitos en la tabla 13.

Tabla 12. Matriz de funciones completa, caso de estudio.
(Fuente: Elaboración propia)

FUNCIONES			
ENTRADAS	FUNCIONES	SALIDAS	TIPO
E. Carga Manual M. Controlador (varios modelos)	Cargar Controlador		Básica
M. Instrumentos de medición.	Inspeccionar Controlador	I. Pasa No Pasa	Innovación.
E. Activación Manual	Activar Cerradura		Mandatorio
E. Activación Manual	Conectar controlador		Innovación
	Inspeccionar Continuidad	I. Activar Led	Mandatorio
E. Activación Manual	Desconectar Terminales		Básica
E. Activación Manual	Descargar Controlador		Básica

Tabla 13. Matriz de requerimientos, caso de estudio.
(Fuente: Elaboración propia)

MATRIZ DE REQUERIMIENTOS			
	REQUERIMIENTO	ESPECIFICACIÓN	MÉTODO DE VALIDACIÓN
BÁSICO	La carga y descarga del controlador debe ser ergonómico.	Altura de trabajo : 0.95m	Flexómetro Calibrado.
DESEMPEÑO	Probar todos los modelo de controladores	12, 18,28,24,30 36 42 48 54 60	Planos de especificaciones de los controladores.
	Diseñar el banco para rechazar controladores fuera del máximo de material.	Tolerancias del controlador de: +/- 1.27 mm	Certificado de Calibración
	La carga de los controladores al banco será manual	El cliente proveerá el dispositivo para manejo de controladores fuera del banco. Por lo que se excluye de este proyecto.	Minuta de acuerdo con el cliente.
	Tiempo de Ciclo de operación	2 min. (Desde la carga hasta la descarga)	Cronometro
	Realizar prueba de continuidad	Suministro de 5 a 10 V, entre terminales y desconectador	Multímetro
	Durabilidad del banco	10 años	Análisis de confiabilidad teórico
	Garantía	1 año	Minuta
DELEITE	Facilitar la carga y descarga del controlador	Plataforma móvil con un desplazamiento de al menos 200 mm	Flexómetro

Se observa un requerimiento de deleite, ya que el cliente no está incluyendo en el proyecto un sistema para carga y descarga. Entonces, se incluirá un requisito para diseñar un sistema que facilite la carga y descarga de los controladores del banco de pruebas, tratando con una simple corredera, aumentar la satisfacción del cliente. Los otros requisitos serán mandatorios y validados en la revisión del diseño virtual.

4.1.3 Planeación general

El resultado de la planeación se muestra en la tabla 14.

Tabla 14. Plan general, caso de estudio.
(Fuente: Elaboración propia)

PLAN GENERAL		
FUNCIONES	TIEMPO (Hrs.)	DISEÑADOR
Carga del controlador	18	Clarisa Parreño
Inspeccionar dimensiones del controlador	27	David Flores
Activar cerradura	27	David Flores
Conectar controlador	27	David Flores
Inspeccionar continuidad	18	David Flores
Desconectar Controlador	18	David Flores
Descargar controlador	18	Clarisa Parreño

Para determinar las horas para cada función realizamos la reunión de inicio del proyecto y con base a la opinión de los expertos acordamos las horas objetivo mostradas en la tabla 14, posteriormente el coordinador del diseño mecánico determinó la duración del proyecto que se muestra en la figura 51. Para este caso usamos "Project" para mostrar la duración y dependencias de las tareas. Concluyendo que el diseño mecánico terminará en 3 semanas (14 días).

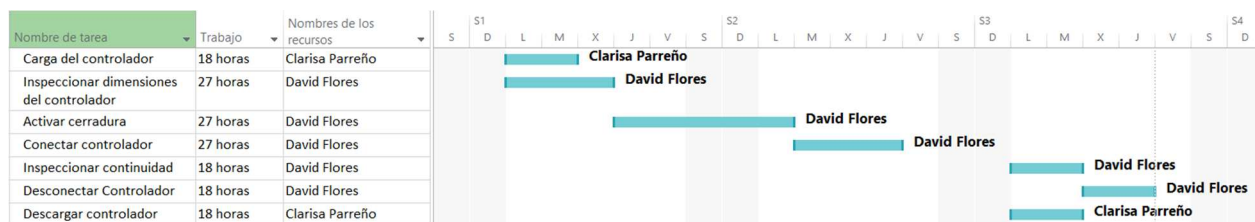


Figura 51. Plan general, caso de estudio.
(Fuente: Elaboración propia)

4.1.3 Planeación de las iteraciones

Se generó una lista de funciones que se atenderán semanalmente, en la primera semana se asignaron tres funciones como se muestran en la tabla 15. De forma similar, se generó cada semana la lista de funciones hasta completar todo el diseño mecánico del banco de pruebas. Donde los hallazgos principales fueron en la estimación de horas ya que en algunos casos excedía y en otros no eran suficientes horas para realizar las funciones.

Tabla 15. Lista de funciones, caso de estudio.
(Fuente: Elaboración propia)

LISTA DE FUNCIONES				
FUNCION	TAREA	DISEÑADOR	DURACIÓN	FECHA DE FIN
Carga del controlador	Completar el diseño mecánico del sistema para carga del controlador	Clarisa Parreño	18 horas	17-07-2018
Inspeccionar Dimensiones del controlador	Completar el diseño mecánico del sistema para inspección del controlador	David Flores	27 horas	18-07-2018
Activar Cerradura	Completar el diseño conceptual de la primera iteración	David Flores	18 horas	20-07-2018

4.1.4 Iteraciones

Se desarrollaron iteraciones de cada función hasta completar el diseño, con fines de síntesis, solo se expone la iteración para la función "Inspección dimensional". Debido a que esta función representa uno de los principales objetivos del producto.

4.1.4.1 Diseño conceptual

A continuación, se muestran los resultados en cada paso del diseño conceptual para el desarrollo de esta función.

Paso 1: Soluciones existentes.

Se procedió con la búsqueda de soluciones para "Inspeccionar dimensiones", considerando los requerimientos relacionados a la función de inspección expuestos en la tabla 16.

Tabla 16. Requerimientos de la función de inspeccionar dimensiones.
(Fuente: Elaboración propia)

REQUERIMIENTO	ESPECIFICACIÓN	MÉTODO DE VALIDACIÓN
Probar todos los modelos de controladores	12, 18,28,24,30 36 42 48 54 60	Planos de especificaciones de los controladores.
Diseñar el banco para rechazar los controladores que esté fuera del máximo de material.	Tolerancias del controlador de: +/- 1.27 mm	Certificado de Calibración
La carga de los controladores al banco será manual	El cliente proveerá el dispositivo para manejo de controladores fuera del banco. Por lo que se excluye de este proyecto.	Minuta de acuerdo con el cliente.

Primer nivel: No se encontraron soluciones similares en proyectos anteriores para esta función.

Segundo nivel: Se encontraron muchos sistemas que realizan la función de inspeccionar dimensionalmente, entre los que destaca: Fixtures, Banco de dimensiones, robot de inspección dimensional y calibradores, pasa no pasa.

Se tienen muchas alternativas para resolver la función, por lo tanto, para este proyecto no consideramos necesario revisar el tercer nivel de búsqueda de soluciones. Ni tampoco convocar a una reunión de lluvia de ideas como lo establece el método.

Paso 2: análisis y selección del concepto.

Usando la matriz de análisis y selección del concepto, se obtuvo que el mejor concepto es el sistema pasa no pasa. Como se muestra en la tabla 17.

Tabla 17. Análisis y selección del concepto.
(Fuente: Elaboración propia)

MATRIZ DE ANÁLISIS Y SELECCIÓN DEL DISEÑO CONCEPTUAL						
				PASO NO PASA	FIXTURE	ROBOT
INSPECCION DIMENSIONAL	COSTO	40 %	3000 USD	+1	0	-1
	TIEMPO CICLO	30 %	5 Sec	-1	0	+1
	ESPACIO DISPONIBLE	30 %	3 * 3 * 2 m	+1	1	0
SUMA				+1	+1	0
SUMA PONDERADA (función del peso del criterio)				0.4	0.3	0.1

4.1.4.2 Revisión

Se realizó una revisión interna y con el cliente, donde se confirmó que la solución conceptual está orientada a las expectativas del cliente. No se generó documento alguno de aprobación ya que el método sugiere solo tener una retroalimentación general del concepto, sin limitar ninguna modificación en la morfología que resultara del diseño final.

4.1.4.3 Diseño óptimo

Basados en el desarrollo preliminar del AMEF, la variable principal para la función de inspección dimensional, es la repetibilidad del dispositivo, entonces los siguientes pasos se enfocan principalmente al análisis de esta variable.

Paso 1; Cálculos de ingeniería.

Los cálculos principales para esta función fueron el análisis de tolerancias y la selección de materiales. Con el objetivo de definir las holguras requeridas y la estabilidad de los materiales durante un proceso de producción continua. Se obtuvo una holgura máxima de 0.5 mm en cada eje y el material seleccionado fue UHMW-PE por su resistencia al desgaste como principal variable de estudio.

Paso 2; Componentes comerciales y estándares.

Para este sistema, solo se seleccionaron algunos accesorios comerciales, tales como: arandelas de ajuste, indexadores de posición y tornillería principalmente.

Paso 3, Diseño para la excelencia.

La característica crítica de esta función son las dimensiones que se inspeccionarán, por lo tanto, la selección de la X que agrega valor es la manufactura y el ensamble. Se utilizó la teoría de DFMA expuesta por Boothroyd, Dewhurst, & Knight (97). El resultado de este análisis fue la selección del proceso de manufactura CNC para la realización de los componentes y una eficiencia del ensamble de 85 %. Se aceptó ese nivel de eficiencia ya que no será un producto que se manufactura en serie y cumple con el presupuesto asignado.

El modo de falla principal detectado fue: Banco acepte controladores fuera de especificaciones. Para lo cual se declararon acciones de certificar el Banco ante un organismo metrológico. A si como seleccionar materiales adecuados para mantener las dimensiones durante un periodo mínimo de uso de un año en condiciones normales de operación.

Tabla 18. AMEF de diseño.
(Fuente: Elaboración propia)

DESIGN FAILURE MODE AND EFFECTS ANALYSIS (DFMEA)																
Developed by: David Flores																
Project Number: Testing Fixture																
Order #	Manufacturing Step and Function	Potential Failure Mode (Failure Issue)	Potential Effect(s) of Failure (Future effect, safety & interactions)	Severity	Potential Cause(s) of failure (Failure Cause)	Prevention Controls	Occurrence	Controls Detection	Detection	RPN	Recommended Action	Responsibility & Target Completion Date	Severity	Occurrence	Detection	RPN
1		Atoramiento del controlador con el banco de pruebas	Daños al Bucket	5	Incorrecta colocación del bucket en el banco de pruebas por operadores no entrenados	Declarar en el manual de operación el correcto uso del banco	3	No hay	1	15						0
2			Daño al banco de pruebas	5	Controlador fuera de especificaciones dimensionales o desgaste de los elementos del banco de pruebas.	Realizar un análisis dimensional para el aseguramiento de las condiciones óptimas de los elementos de desgaste del banco	3	No hay	1	15						0
3			Daño al operador	10	Falla de manipulador que carga el controlador	Redactar un documento que separe los riesgos por manipulación del controlador, ya que no se incluye en el alcance de este proyecto.	2	No hay	3	60						
4	Inspeccionar dimensionalmente los controladores	No ajusta el controlador en el banco	Aceptar controladores fuera de especificaciones dimensionales	5	Descalibración del banco de pruebas	No hay	9	No hay	6	270	Declaración en el manual de operación y mantenimiento, el peso máximo que puede cargar el operador.	Diseño Mecánico	6	5	3	90
5			Deja pasar piezas malas o rechaza piezas buenas	5	No se ajusto el banco para absorber la variación permitida de los controladores	No existe	9	No hay	6	270	Mesa de apoyo con guías para la inserción del bucket, que ayude a deslizarlo hasta su posición final.	Diseño Mecánico	4	5	3	60
6			5	Desgaste de los materiales que hacen contacto entre el banco y el controlador	No existe	9	No hay	6	270	Realizar un análisis de tolerancias y selección adecuado de material. Para determinar la vida útil de los elementos de desgaste	Diseño Mecánico	2	5	2	20	

Para la representación virtual se modelo el diseño usando “SolidWorks”, figura 52. Y para la representación ven realidad aumentada utilizamos las herramientas que ofrece el dispositivo de HoloLens de Microsoft (96).



Figura 52. Diseño final de la primera iteración en un ambiente virtual.
(Fuente: Elaboración propia)

4.1.5 Revisión con el cliente

Durante la revisión con el cliente, fue grata la agilidad de la revisión. El modelo virtual con todo el respaldo técnico, basado en las funciones y sus fronteras como requisitos, prácticamente fue un protocolo la revisión. Durante la interacción virtual el cliente detectó un área de oportunidad relacionada con la forma de conexión del controlador, ya que el operador tendría que desplazarse significativamente, lo que impactaría en el tiempo ciclo definido. Para lo cual solicito un cambio para esta función. Se realizó una segunda iteración, donde el resultado fue diseñar un mecanismo que permitiera conectar el controlador al momento de colocarlo en el banco de pruebas, y con este cambio se aprobó el diseño del banco de pruebas para controladores.

4.1.6 Retrospectiva

Las lecciones aprendidas principales fueron; la comunicación con el cliente y la estimación de horas asignadas para las iteraciones.

4.1.7 Entregables

Con la ayuda de macros en la plataforma de "Solidworks", se realizaron los planos de fabricación y ensamble, así como la lista de partes maquinadas y comerciales de manera automática. También, se realizaron los manuales operación y mantenimiento, como lo establecen los entregables mandatorios para la empresa Industrial Automation México.

CONCLUSIONES

Se cumplió el objetivo principal de este trabajo de tesis, desarrollando un nuevo método para el diseño mecánico, enfocado a PDT. Donde se concluye que es factible acoplar los principios ágiles con los métodos y herramientas del diseño mecánico tradicional, obteniendo productos ágiles y técnicamente fundamentados, como se declaró en la hipótesis de esta tesis. Algunos principios ágiles y técnicas del diseño se modificaron para darle sentido a este método y obtener productos que se adapten a los cambios de los clientes, sin descuidar la calidad técnica del producto. La primera adecuación fue en el proceso de definición del producto, donde la participación del cliente con el equipo de diseño fue crucial para confirmar las expectativas del producto en términos de funciones y sus requisitos, en lugar de obtener las características del producto por una sola persona, como lo establecen los principios ágiles. También, durante el proceso de iteraciones, se contrastan los principios de "Scrum", donde se establece que la calidad del producto dependerá de la experiencia y confianza hacia el equipo de diseño, lo cual tuvo que ser modificado en esta propuesta metodológica, buscando que la calidad dependa de un proceso sistemático, que permita a diseñadores de poca experiencia lograr diseños de calidad similar o mejor al diseño de un equipo experimentado. En el proyecto piloto se mostró este beneficio durante el desarrollo de la función de inspección dimensional. Donde gracias al proceso del diseño conceptual, se identificó una solución de dominio público, usando el conocimiento de sistemas ya probados, para únicamente someterlos al proceso de optimización de dicha propuesta. Logrando una reducción del 30% sobre el tiempo estimado para el desarrollo del proyecto. Obteniendo un diseño técnicamente fundamentado en todas sus características tanto físicas como funcionales. Finalmente, durante la evaluación del diseño, la representación del producto en realidad aumentada fue clave para realizar un incremento en una de las funciones del producto. Gracias a la interacción del cliente con el holograma del producto le permitió identificar un parámetro que difícilmente se hubiera identificado antes de liberar el diseño. Demostrando con este ejemplo, la factibilidad de lograr incrementos en el producto desde la etapa del diseño, gracias a los avances actuales de la tecnología en el campo de la realidad aumentada.

APORTACION DE LA TESIS

Un nuevo método para el diseño mecánico, es la propuesta de aporte y originalidad de este trabajo de tesis. Este nuevo método busca ser una opción para responder a las necesidades actuales de las empresas, instituciones y centros de investigación que desarrollan PDT en el campo del diseño mecánico. Logrando que a través de una nueva herramienta metodológica, les permita obtener productos exitosos para incrementar su sustentabilidad. En relación a la aportación sobre la base del conocimiento científico, esta tesis presenta un nuevo método que podrá ser aplicado en la ciencia de la mecánica. Debido a que a través del acoplamiento de los modelos prescriptivos que presenta Pahl & Beitz (23), Ullman (34) y "Scrum" (55), éste método representa una evolución de los métodos tradicionales en el campo del diseño mecánico, a partir de romper la estructura del proceso tradicional en cascada, permitiendo lograr incrementos en el diseño mecánico bajo un enfoque de métodos ágiles.

RECOMENDACIONES

La metodología ha sido implementada en el departamento de diseño mecánico de la empresa Industrial Automation México, e iniciará un proceso de maduración. Sin embargo, se recomienda expandir esta filosofía ágil más allá del departamento del diseño mecánico, con el objetivo de unificar los criterios y la cultura ágil de todos los departamentos de la organización involucrados en el desarrollo de proyectos. Debido a que la mayoría de PDT que desarrolla la empresa Industrial Automation, requieren de la integración de varias disciplinas para obtener un nuevo producto.

A las empresas e instituciones en general que desarrollan PDT, se recomienda capacitar a los equipos de diseño mecánico, sobre la ideología, la teoría y la filosofía detrás de las prácticas y métodos ágiles, previo a la implementación de este nuevo método. Ya que el coordinador del diseño mecánico no solo debe conocer el método sino los principios y valores detrás de éste, buscando generar confianza en los nuevos equipos, debido a que la experiencia los alienta a ver que el esfuerzo de cambio puede funcionar para ellos. Es importante no dejarse llevar por un método a ciegas, ser crítico y ver el valor de cada paso del método, ya a que a menudo se necesitan adaptaciones para mejorar el trabajo de los equipos y aumentar la aceptación del nuevo método por el equipo de diseño. Sin embargo, no corte partes del método a menos que esté bien razonado. Probablemente habrá cierta renuencia al cambio, esté preparado para esto, sea paciente y trabaje para ayudar al equipo a ver los beneficios y cómo el método puede ayudarlos. Las revisiones y retrospectivas de las iteraciones, pueden ser un desafío para los equipos, sin embargo pueden proporcionar un gran valor en la retroalimentación para la mejora continua de la organización.

BIBLIOGRAFÍA

1. REYNISDOTTIR, Þórdís. Scrum in mechanical product development, case study of a mechanical product development team using scrum. Gothenburg, Sweden : Chalmers university of technology, 2013.
2. ALSO, Joan Josep Mauri. Integration of the scrum methodology in mechatronic product development. España : Universitat de Vic Escola Politècnica Superior, 2015.
3. DWYER, Stephen. Agile design project methodology for small teams developing mechatronic systems. Canada : University of Alberta, 2017.
4. JENNIFER. The standish group. CHAOS. [En línea] Jennifer. [Citado el: 20 de 12 de 2018.] <https://www.standishgroup.com/>.
5. CONSEJO NACIONAL DE CIENCIA Y TECNOLOGIA. Cartas de cierre para proyectos de la convocatoria 2016. Ciudad De México : CONACYT, 2017.
6. CONSEJO NACIONAL DE CIENCIA Y TECNOLOGIA. Cartas de cierre para proyectos de la convocatoria 2017. Ciudad De Mexico : CONACYT, 2018.
7. A survey of design philosophies, models, methods and systems. EVBUOMWAN, SIVALOGANATHAN y JEBB. 4, London : Springer, 1996, Journal of Engineering Manufacture, Vol. 210, págs. 301-317.
8. Generations in design methodology. BROADBENT, Jonh. 1, New York : The Design Journal, 2015, Vol. 6.
9. ARCHER, L. B. Development in design methodology. London, pp. 57-82 : John Wiley & Sons Ltd, 1984. págs. 57-82.
10. CROSS, N. Engineering design methods strategies for product design. Chichester; New York : Wiley, 2000.
11. LAWSON, B. R. Cognitive strategies in architectural design. London : John Wiley, 1984. págs. 209-220.

12. Design for quality and design methodology. HUBKA, V. 1, New York : Springer, 1992, Journal of Engineering Design, Vol. 3, págs. 3(1), 5-15.
13. NORMAN, Don. The design of everyday things (Revised and expanded ed.). New York : Basic Books, 2013.
14. ADAMS, KEVIN Macg. Design methodologies. [aut. libro] KEVIN MACG ADAMS. Non-Functional Requirements in Systems Analysis and Design. Switzerland : Springer International Publishing, 2015, págs. 15-42.
15. Design methodologies: industrial and educational applications. TOMIYAMA, T., y otros. 2, New York : Elsevier, 2009, Manufacturing Technology, Vol. 58, págs. 543–565.
16. Modeling design processes. TAKEDA, Hideaki. 4, Tokyo : AI MAGAZINE, 1990, AI Magazine Vol. 11, No. 4, págs. 37-48.
17. A review of research in mechanical engineering design. part i: descriptive, prescriptive, and computer-based models of design processes. FINGER, S. y DIXON, J. R. Pennsylvania : Springer, 1989, Research in Engineering Design Vol. 1, No. 1, págs. 51-67.
18. MARCH, L. In developments in design methodology. [aut. libro] N. Cross. The Logic of Design. London : John Wiley, 1984, págs. 265-276 .
19. MATTCHET, E. y BRIGGS, A. H. Practical design based on method (fundamental design method). [aut. libro] S.A. Gregory. The Design Method. London : S. A. Gregory, 1966, págs. 183-199.
20. An evolutionary process model of design. GERO, J. y HYBS, I. 3, Sydney, Australia : Elsevier, 1992, Design Studies, Vol. 13, págs. 273-290.
21. Conference on design methods. JONES, J. C. y THORNLEY, D. G. London : Pergamon, Press, September 1962. A Method of Systematic Design.
22. ASIMOW, M. Introduction to design. Engle-wood Cliffs New Jersey : Prentice-Hall, 1962.
23. PAHL, G y BEITZ, W. Engineering design, third edition. London : Springer, 2007.

24. THE ASSOCIATION OF GERMAN ENGINEERS. VDI Guideline. Berlin : The Association of German Engineers (VDI), 1987.
25. WATTS, R. D. The Design Method. [aut. libro] Ronald D. Watts. The elements of design. Butterworth, London : Springer, 1966, págs. 85-95.
26. MARPLES, D. L. The decisions of engineering design. London : The Institution of Engineering Designers, 1960.
27. KRICK, E. V. Engineering and engineering design, 2nd edition. New York : John Wiley, 1969.
28. FRENCH, M. J. Conceptual design, 3rd edition. London : Springer, 1999.
29. can design be taught? HARRIS, Sir A. 2, New York : ICE Publishing, August de August, 1980, Proc. Instn Civ. Engrs, Vol. 70, págs. Part 1, No. 68, 409-416.
30. PUGH, S. Total design-integrated methods for successful product engineering, 1st edition. England : Addison, Wesley, 1990.
31. STANDARDS, BRITISH. BS-7000, management of design. London : British Standard Institution, 2015.
32. SUH, P.Nam. The principles of design, ed 1. New York, USA : Oxford University Press, 1990.
33. TAGUCHI, Genichi. Introduction to quality engineering : designing quality into products and processes. Tokyo : Asian productivity organization, 1990.
34. ULLMAN, David G. The mechanical design process. New York, US : McGraw-Hill, 2010.
35. A model of cognitive activities in design. JOHNNY Huysentruyt, VÉRONIQUE Lespinet-Najib, DAVID Chen. Bordeaux - France : HAL, 2012. Conférence Internationale de Modélisation, Optimisation et SIMulation - MOSIM'12.
36. Design philosophy: the state-ofthe-art. YOSHIKAWA, H. 2, Tokyo : Elsevier, 1989, Keynote paper, Annals of the CIRP, Vol. 38, pág. 2.

37. IA and new computational models of design. GERO, J. S. Beijing, China : Colloquium on Knowledge-based systems in civil engineering, 1993. págs. 25-32.
38. A Design Process Model That Unifies General Design Theory And Empirical Findings. TOMIYAMA, Tetsuo. 5, Tokio : ASME, 1995, Design Engineering Technical Conferences, Vol. 2.
39. Elargissement de principes de la cybernétique vers l'ingénierie et la production. Lhote, F., Chazelet, Ph., and M. Dulmet. s.l. : Nancy, June 1998. INCOM'98. págs. 23-25.
40. VISSER, W. The cognitive artefacts of designing. New Jersey : Lawrence Erlbaum Associates, Inc, 2006.
41. IA/OR computational model for integrating qualitative and quantitative design methods. AGOGINO. Amherst Massachusetts : University of Massachusetts, 1989. Proceedings of NSF Engineering in Design Conference. págs. 97-112.
42. Computational models of design processes. NEVILLE, G. E. 1, Dordrecht : Springer, 1988, Design Theory '88, Vol. 17, págs. 82-116.
43. MOSTOW, J. A domain-independent model of knowledge-based design. [aut. libro] Sandra L. NewsomeW. R. SpillersSusan Finger. Design Theory. New York : Springer Verlag, 1988, págs. 97-99.
44. DIXON, J. R. Iterative redesign and respecification: research on computational models of design processes. [aut. libro] Sandra L. NewsomeW. R. SpillersSusan Finger. Proceedings of the 1988 nsf grantee workshop on design theory and methodology. design theory 88. New York : Srpinger Verlang, 1988, págs. 104-107.
45. Engineering design is a computable function. FITZHORN, P. A. 1, London : Cambridge University Press, 1994, Artif, Intell, for Enging Des., Analysis and Mfg, Vol. 8, págs. 8, 35-44.
46. Dimensional variable expansion - A formal approach to innovate design. CAGAN, J. y AGOGINO, A. M. 2, Pennsylvania USA : Springer, 1991, Res. in Enging Des., Vol. 4, págs. 3, 75-85.
47. DYM, Clive L. y and LITTLE, Patrick. Engineering design: a project-based introduction. New York : John Wiley, 2004.

48. DIETER, E. George y SCHMIDT, Linda C. Engineering design. New York : McGraw-Hill Education, 2012.
49. ERTAS, Atila y JONES, Jesse C. Engineering design process 2ed. New York : John Wiley & Sons, 1996.
50. HILL, Percy H. The science of engineering design. Wisconsin : Ilustrada, 1970.
51. HAIK, Yousef y SHAHIN, Tamer M. Engineering design process. New York : Cengage Learning, 2010.
52. Scaling agile methods to regulated environments: an industry case study. B. FITZGERALD, K. J. STOL, R. O'SULLIVAN, and D. O'BRIEN. San Francisco, CA, USA : IEEE, 2013. 35th International Conference on Software Engineering (ICSE).
53. CUNNINGHAM, Ward. Manifiesto for agile software development. [En línea] Ward Cunningham, 2016. [Citado el: 14 de 09 de 2018.] [http: // agilemanifesto.org/](http://agilemanifesto.org/).
54. HIGHSMITH, James A. Agile software development ecosystems. Boston : Addison-Wesley, 2002.
55. KENNETH, S. Rubin. Essential scrum: a practical guide to the most popular agile process. Arbor, Michigan USA : Addison-Wesley, 2013.
56. CONSORTIUM, gile Business. Agile business consortium home page. [En línea] Agile Business Consortium. [Citado el: 18 de September de 2018.] [https: // www.agilebusiness.org/](https://www.agilebusiness.org/).
57. COCKBURN, Alistair. Crystal clear: a human-powered methodology for small teams. Boston : Addison-Wesley, 2005.
58. MOORE, Geoffrey. Inside the tornado: marketing strategies from silicon valley's cutting edge. New York : HarperBusiness, 1995.
59. CARMICHAEL, David J. Anderson and ANDY. Essential kanban condensed. Seattle : Blue Hole Press, 2016.

60. ANDERSON, David J. Kanban: successful evolutionary change for your technology business. USA : Blue Hole Press, 2010.
61. LADAS, Corey. Scrum-Ban. [En línea] Corey Ladas. [Citado el: 14 de October de 2018.] <http://leansoftwareengineering.com/ksse/scrum-ban/>.
62. REDDY, Ajay. The scrumban [r]evolution: getting the most out of agile, scrum, and lean kanban. New York : Addison-Wesley, 2016.
63. WEINBERG, Gerald M. Quality software management): systems thinking. Dorset House : Publishing Co, Inc., 1992.
64. SAFOUTIN, Michael John. A methodology for empirical measurement of iteration in engineering. Washington US : University of Washington, 2003.
65. WYNN, David Charles. Model-based approaches to support process improvement in complex product development. Cambridge : University of Cambridge, 2007.
66. EDC, Cambridge. Cambridge advanced modeller. [En línea] [Citado el: 18 de 09 de 2018.] <https://www-edc.eng.cam.ac.uk/cam/>.
67. SUSS, Samuel. Coordination in complex product development. McGill University : PhD thesis, 2011.
68. A framework for comparing design. EISENBART Boris, GERICKE Kilian, BLESSING Luciënne. Denmark : Proceedings of the 18th International Conference on Engineering Design (ICED 11), Impacting Society through Engineering Design Vol. 2: Design Theory and Research Methodology, Lyngby/Copenhagen, 2011.
69. Design models in mechatronic product development. ANDREASEN, Jacob Buur and MOGENS Myrup. 3, DK-2800 Lyngby, Denmark : Elsevier, 1989, Vol. 10.
70. Stages in product lifecycle:trans-disciplinary design context. QURESHI AHMED Jawad, GERICKE Kilian , and BLESSING Lucienne. 20, Luxembourg : Elsevier, 2014, Vol. 21.
71. ALITE, Besiana y NIKOLAY, Spasibenko. Project suitability for agile methodologies. Umeå, Zweden : Umeå School of Business., 2008.

72. Agile systems engineering versus agile systems engineering. HABERFELLNER, Reinhard y WECK, Olivier de. New York : INCOSE International Symposium, 15(1):1449–1465, 2005.
73. P.K. Palani Rajan, MICHAEL VAN WIE, Matthew I. CAMPBELL, Kristin L. Wood, and KEVIN N. Otto. An empirical foundation for product flexibility. Cambridge, USA : Design Studies, 26(4):405 – 438, 2005.
74. DEREK Bonderczuk, PATRICK Currier, and MATTHEW Nelson. Application of principles from the scrum agile method to a prototype vehicle control development cycle. Montreal, Canada : Proceedings of the ASME 2014 international mechanical engineering congress and exposition, November 2014.
75. REYNISDÓTTIR, Þórdís. Scrum In Mechanical Product Development. Chalmers University of Technology : Master's thesis, 2013.
76. MACCHERONE, Larry. Top 10 questions when using agile on hard ware projects. [En línea] [Citado el: 15 de February de 2018.] <http://maccherone.com/larry/2010/02/23/top-10-questions-when-using-agile-on-hardware-projects/>.
77. JACKSON, Chad. Agile development for mechatronics products? [En línea] [Citado el: 13 de December de 2018.] <http://www.lifecycleinsights.com/engineering-management/agile-analogues/>.
78. THILMANY, Jean. Can mechanical engineers adopt agile product development? [En línea] [Citado el: 8 de December de 2018.] <http://www.lifecycleinsights.com/engineering-profession/an-mechanical-engineers-adopt-agile-product-development/>.
79. MYLLERUP, Bent. Why agile does matter in an embedded development environment? [En línea] [Citado el: 11 de December de 2018.] <https://www.scrumalliance.org/community/articles/2011/march/why-agile-does-matter-in-an-embedded-development-e>.
80. WIRFS-BROCK, Rebecca. Agile landing zones. [En línea] [Citado el: 16 de December de 2018.] <http://wirfs-brock.com/blog/2011/07/28/agile-landing-zones/>.
81. FOOTE, Brian y YODER, Joseph. Big ball of mud. [En línea] [Citado el: 10 de June de 2018.] <http://www.laputan.org/mud/>.

82. BRAND, Stewart. How buildings learn?: what happens after they're built? New York : Penguin Books, 1995.
83. BACKBLAZE, INC. Application of scrum methods to hardware development. s.l. : White Paper, Online, July 2015.
84. DENNING, Steve. Wikispeed: how a 100 mpg car was developed in 3months. [En línea] [Citado el: 30 de December de 2018.] <http://www.forbes.com/sites/stevedenning/2012/05/10/wikispeed-how-a-100-mpg-car-was-developed-in-3-months/#3499b4a03f3e>.
85. ROTHMAN, Johanna. Helping hardware be agile. [En línea] [Citado el: 10 de December de 2018.] <http://www.jrothman.com/mpd/agile/2015/12/helping-hardware-be-agile-part-1/>.
86. CYNEFIN. Cynefin: la complejidad que nos rodea. [En línea] Jun de 2009. [Citado el: 20 de December de 2018.] <https://martinalaimo.com/es/blog/cynefin>.
87. ULWICK, Antony W. What customers want. USA : McGraw-Hill, 2005.
88. ZULTNER, Richard, TERNINKO, John y MAZUR, Glenn H. House of quality (qfd) tutorial. [En línea] [Citado el: 15 de 06 de 2018.] <http://www.qfdonline.com/qfd-tutorials/house-of-quality-tutorial/>.
89. VERDUYN, Dave. kanomodel.com. About the kano model. [En línea] [Citado el: 17 de March de 2018.] <https://www.kanomodel.com/articles/>.
90. SAVRANSKY, Semyon D. Engineering of creativity: introduction to triz methodology of inventive problem. Florida USA : CRC Press, 2000.
91. ZWICKY, Fritz. Morphologischen weltbild. [aut. libro] Fritz Zwicky. Entdecken, erfinden, forschen im morphologischen weltbild. Munchen : Baeschlin, 1989.
92. ANLEITNER, Michael A. The power of deduction: failure modes and effects analysis for design. Wisconsin, USA : ASQ Quality Press, 2010.
93. Designer support for product structuring - development of a dfx tool within the design coordination framework. TICHEM, Marcel y STORM, Ton. 2-3, Delft, The Netherlands : Elsevier, 1997, Computers in Industry 33, Vol. 33, págs. 155-163.

94. Ar/Rp-based tangible interactions for collaborative design. PARK, Jae Yeol Lee & GUE Won Rhee & HYUNGJUN. 7–8, Gwangju, South Korea : Springer, 2009, Int J Adv Manuf Technol 45, Vol. 45, págs. 649–665.
95. Structure design and analysis with integrated ar-fea. ONG, S.K. y HUANG, J.M. 1, Singapore : Elsevier, 2017, CIRP Annals - Manufacturing Technology 66, Vol. 66, págs. 149–152.
96. HOLOLENS, Microsoft. Microsoft hololens user guide. [En línea] [Citado el: 28 de July de 2017.] <https://archive.org/details/microsoft-hololens/page/n1>.
97. BOOTHROYD, Geoffrey, DEWHURST, Peter y KNIGHT, Winston A. Product design for manufacturing and assembly 3rd edition. Massachusetts, USA : CRC Press, 2010.
98. QUALITY-ONE, International. Quality-one international. [En línea] [Citado el: 15 de December de 2018.] <https://quality-one.com/fmea/>.
99. FOWLER, Martin y BECK, Kent. Planning extreme programming. USA : Addison-Wesley, 2000.
100. A framework for research in design computing. GERO, J. S. and ML., Maher. Vienna, Austria : Challenges of the Future [15th eCAADe Conference Proceedings, 1997. ECAADE'97.
101. MCKENNA, Dave. The art of scrum. Pennsylvania, US : CA Press, 2016.
102. NORMATIVIDAD MEXICANA. Gestion de la tecnología - proyectos tecnologicos - requisitos. Norma Mexicana IMNC. 2008. MNX-GT-002-IMNC-2008.
103. Reddy, Ajay. The scrumban [r]evolution: getting the most out of agile, scrum, and lean kanban (agile software development). USA : Addison-Wesley, 2015.
104. SUTHERLAND, JJ. Scrum.Org. The home of scrum. [En línea] [Citado el: 01 de 07 de 2018.] <https://www.scrum.org/resources/what-is-scrum>.
105. Structuring design knowledge for better design synthesis. TOMIYAMA, T. Stockholm : Design Society, 2003. In CIRP International Design Seminar. págs. 24-26.

106. TOOLEY, Mike. Design engineering manual. Surrey, England : Butterworth-Heinemann, 2009.

107. Does agile work? — A quantitative analysis of agile. SERRADOR, Pedro y PINTO, Jeffrey K. 15, Toronto, Canada : Elsevier, 2015, International Journal of Project Management, Vol. 33, págs. 1040-1051.

ANEXO

Tabla 19. Severidad AMEF de diseño.
(Fuente: Modificación Quality One (98))

Effect	Criteria: Severity of Effect on Product (Customer Effect)	Rank	Effect	Criteria: Severity of Effect on Product (Manufacturing/Assembly Effect)
Failure to Meet Safety and/or Regulatory requirements	Potential failure mode affects safe vehicle operation and/or involves noncompliance with government regulation without warning	10	Failure to Meet Safety and/or Regulatory requirements	May endanger operator (machine or assembly) without warning.
	Potential failure mode affects safe vehicle operation and/or involves noncompliance with government regulation with warning	9		May endanger operator (machine or assembly) with warning.
Loss or Degradation of Primary Function	Loss of primary function (vehicle inoperable, does not affect safe vehicle operation)	8	Major Disruption	100% of product may have to be scrapped. Line shutdown or stop ship
	Degradation of primary function (vehicle operable, but at reduced level of performance)	7	Significant Disruption	A portion of the production run may have to be scrapped. Deviation from primary process including decreased line speed or added manpower
Loss or Degradation of Secondary Function	Loss of secondary function (vehicle operable, but comfort/convenience functions inoperable)	6	Moderate Disruption	100% of production run may have to be reworked off line and accepted
	Degradation of secondary function (vehicle operable, but comfort / convenience functions at reduced level of performance)	5		A portion of the production run may have to be reworked off line and accepted
Annoyance	Appearance or Audible Noise, vehicle operable, item does not conform and noticed by most customers (> 75%)	4	Moderate Disruption	100% of production run may have to be reworked in station before it is processed
	Appearance or Audible Noise, vehicle operable, item does not conform and noticed by many customers (50%)	3		A portion of the production run may have to be reworked in-station before it is processed.
	Appearance or Audible Noise, vehicle operable, item does not conform and noticed by discriminating customers (<25%)	2	Minor Disruption	Slight inconvenience to process, operation or operator
No Effect	No discernible effect	1	No effect	No discernible effect

Tabla 20. Ocurrencia AMEF de diseño.
(Fuente: Modificación Quality One (98))

Likelihood of Failure	Criteria: Occurrence of Cause - DFMEA (Design life/reliability of item/vehicle)	Criteria: Occurrence of Cause - PFMEA (Incidents per items/vehicles)	Rank
Very High	New technology/new design with no history.	≥ 100 per thousand ≥ 1 in 10	10
High	Failure is inevitable with new design, new application, or change in duty cycle/operating conditions.	50 per thousand 1 in 20	9
	Failure is likely with new design, new application, or change in duty cycle/operating conditions.	20 per thousand 1 in 50	8
	Failure is uncertain with new design, new application, or change in duty/operating conditions.	10 per thousand 1 in 100	7
Moderate	Frequent failures associated with similar designs or in design simulation and testing.	2 per thousand 1 in 500	6
	Occasional failures associated with similar designs or in design simulation and testing.	.5 per thousand 1 in 2,000	5
	Isolated failures associated with similar design or in design simulation and testing.	.1 per thousand 1 in 10,000	4
Low	Only isolated failures associated with almost identical design or in design simulation and testing.	.01 per thousand 1 in 100,000	3
	No observed failures associated with almost identical design or in design simulation and testing.	≤ .001 per thousand 1 in 1,000,000	2
Very Low	Failure is eliminated through preventative control.	Failure is eliminated through preventive control.	1

Tabla 21. Detección AMEF de diseño.
(Fuente: Detección Quality One (98))

Opportunity for Detection	Criteria: Likelihood of Detection by Design Control	Rank	Likelihood of Detection	Opportunity for Detection	Criteria: Likelihood of Detection by Process Control
No detection opportunity	No current design control; Cannot detect or is not analyzed.	10	Almost Impossible	No detection opportunity	No current process control; Cannot detect or is not analyzed
Not likely to detect at any stage	Design analysis/detection controls have a weak detection capability; Virtual Analysis (e.g., CAE, FEA, etc.) is not correlated to expected actual operating conditions.	9	Very Remote	Not likely to detect at any stage	Failure Mode and/or Error (Cause) is not easily detected (e.g. random audits)
Post Design Freeze and prior to launch	Product verification/validation after design freeze and prior to launch with pass/fail testing (Subsystem or system testing with acceptance criteria such as ride and handling, shipping evaluation, ect.).	8	Remote	Problem Detection Post Processing	Failure Mode detection post-processing by operator through visual/tactile/audible means
	Product verification/validation after design freeze and prior to launch with test to failure testing (Subsystem or system testing until failure occurs, testing of system interactions, etc.).	7	Very Low	Problem Detection at Source	Failure Mode detection in-station by operator through visual/tactile/audible means or post-processing through use of attribute gauging (go / no-go, manual torque check, clicker wrench, etc.)
	Product verification/validation after design freeze and prior to launch with degradation testing (Subsystem or system testing after durability test, e.g., function check).	6	Low	Problem Detection Post Processing	Failure Mode detection post-processing by operator through use of variable gauging or in-station by operator through use of attribute gauging (go/no-go, manual torque check/clicker wrench, etc.)
Prior to Design Freeze	Product validation (reliability testing, development or validation tests) prior to design freeze using pass/fail testing (e.g., acceptance criteria for performance, function checks, etc.).	5	Moderate	Problem Detection at Source	Failure Mode or Error (Cause) detection in-station by operator through use of variable gauging or by automated controls in-station that will detect discrepant part and notify operator (light, buzzer, etc.) Gauging performed on setup and first-piece check (for set-up causes only)
	Product validation (reliability testing, development or validation tests) prior to design freeze using test to failure (e.g., until leaks, yields, cracks, etc.)	4	Moderately High	Problem Detection Post Processing	Failure Mode detection post-processing by automated controls that will detect discrepant part and lock part to prevent further processing
	Product validation (reliability testing, development or validation tests) prior to design freeze using degradation (e.g., data trends, before/after values, etc.).	3	High	Problem Detection at Source	Failure Mode detection in-station by automated controls that will detect discrepant part and automatically lock part in station to prevent further processing.
Virtual Analysis Correlated	Design analysis/detection controls have a strong detection capability. Virtual analysis (e.g., CAE, FEA, etc.) is highly correlated with actual or expected operating conditions prior to design freeze	2	Very High	Error Detection and/or Problem Prevention	Error (Cause) detection in-station by automated controls that will detect error and prevent discrepant part from being made.
Detection not applicable; Failure Prevention	Failure cause or failure mode can not occur because it is fully prevented through design solutions (e.g., proven design standard, best practice or common material, etc.)	1	Almost Certain	Detection not applicable; Error Prevention	Error (Cause) prevention as a result of fixture design, machine design or part design. Discrepant parts cannot be made because item has been error proofed by process/product design

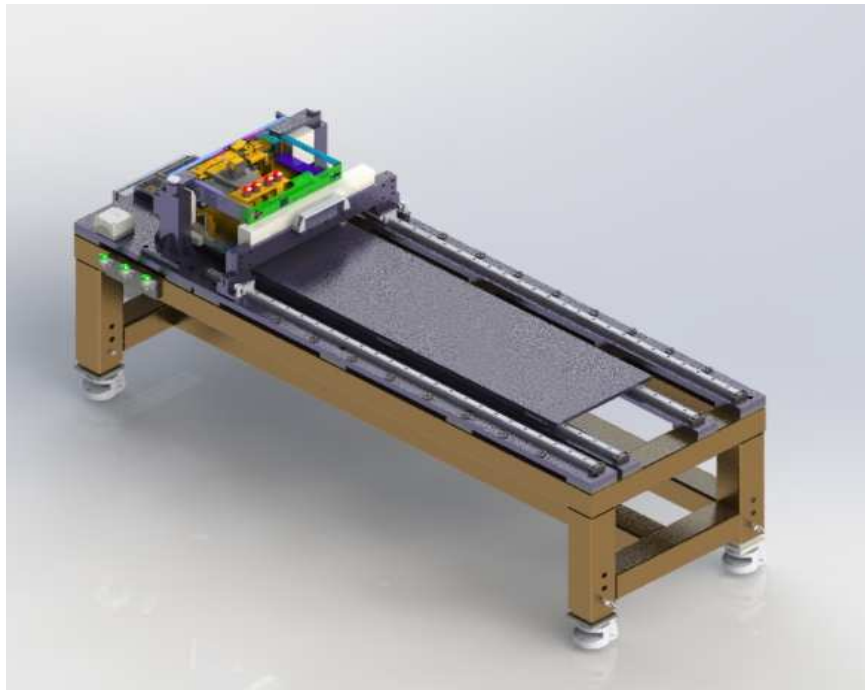


Figura 53. Diseño final de la primera iteración en un ambiente virtual.
(Fuente: Elaboración propia)

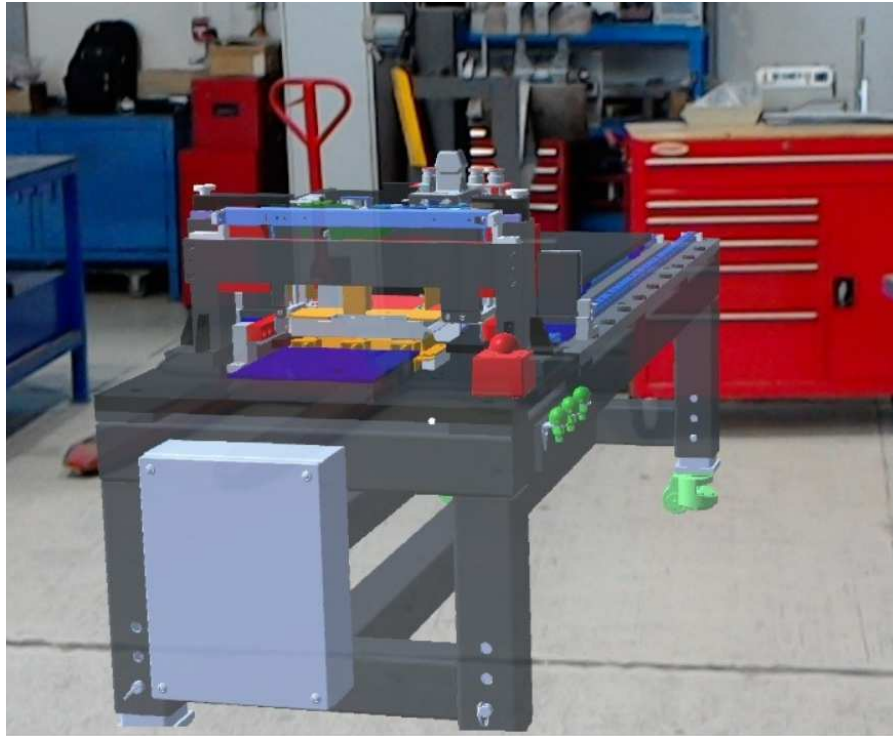


Figura 54. Diseño final de la primera iteración usando Realidad aumentada.
(Fuente: Elaboración propia)

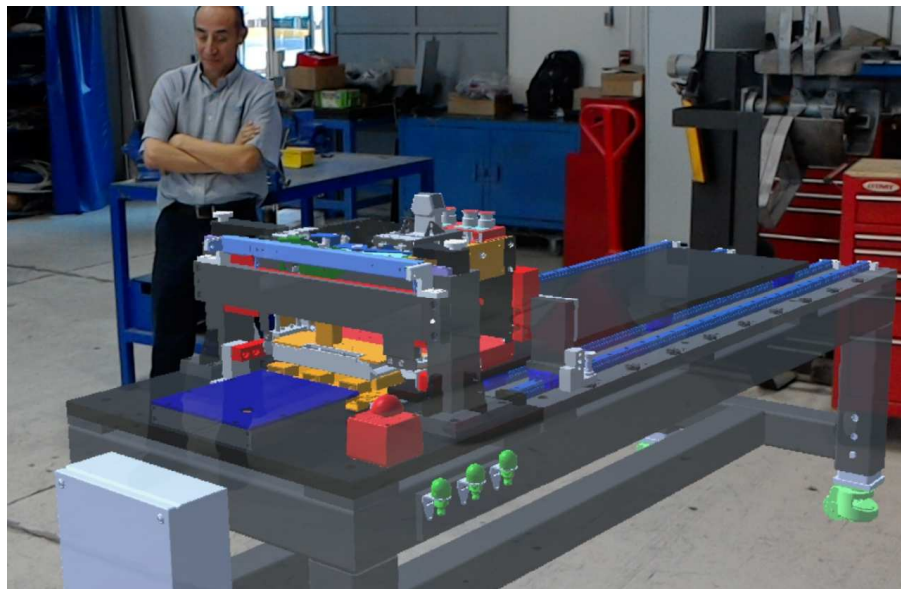


Figura 55. Diseño final de la primera iteración usando Realidad aumentada.
(Fuente: Elaboración propia)

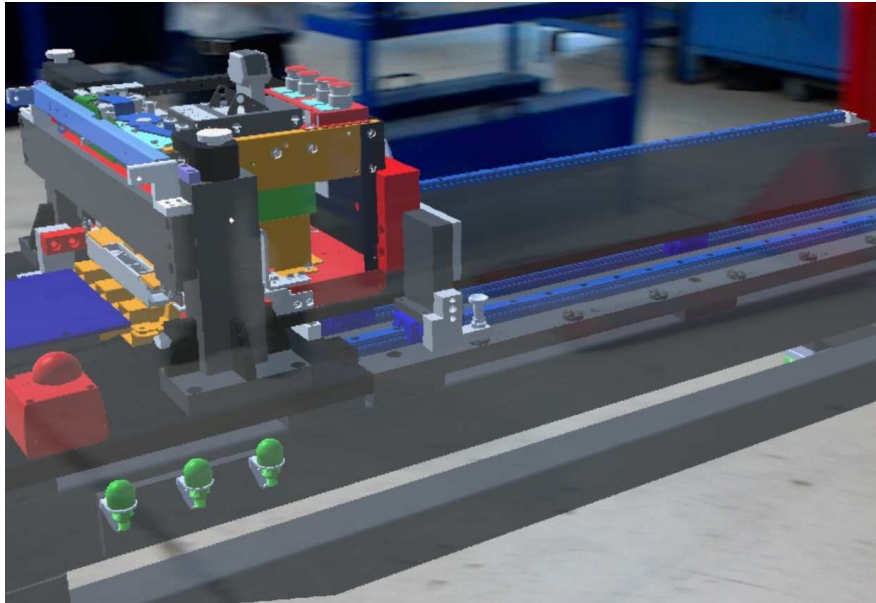


Figura 56. Diseño final de la primera iteración usando Realidad aumentada.
(Fuente: Elaboración propia)

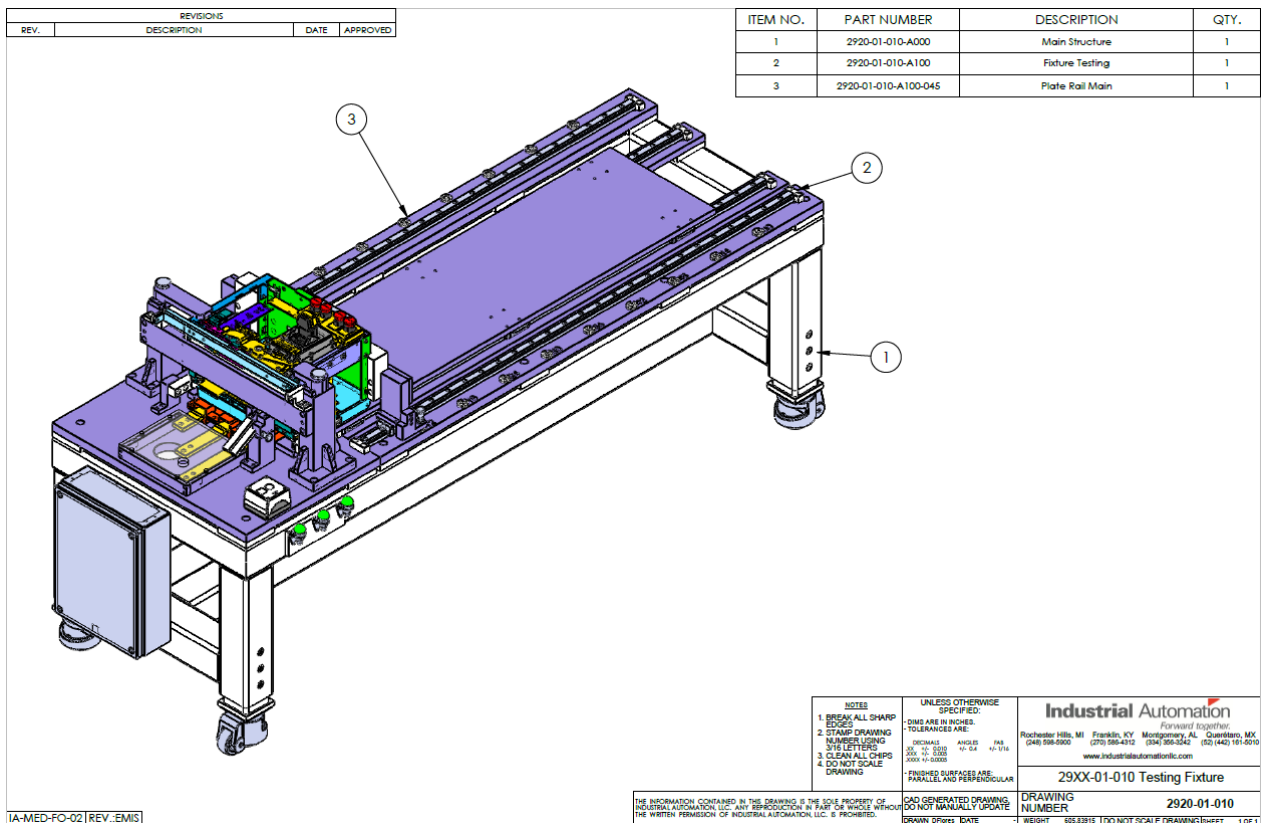


Figura 57. Diseño final, plano de ensamble.
(Fuente: Elaboración propia)