

3.2 CICLO DE VIDA Y FASES DE LA METODOLOGÍA

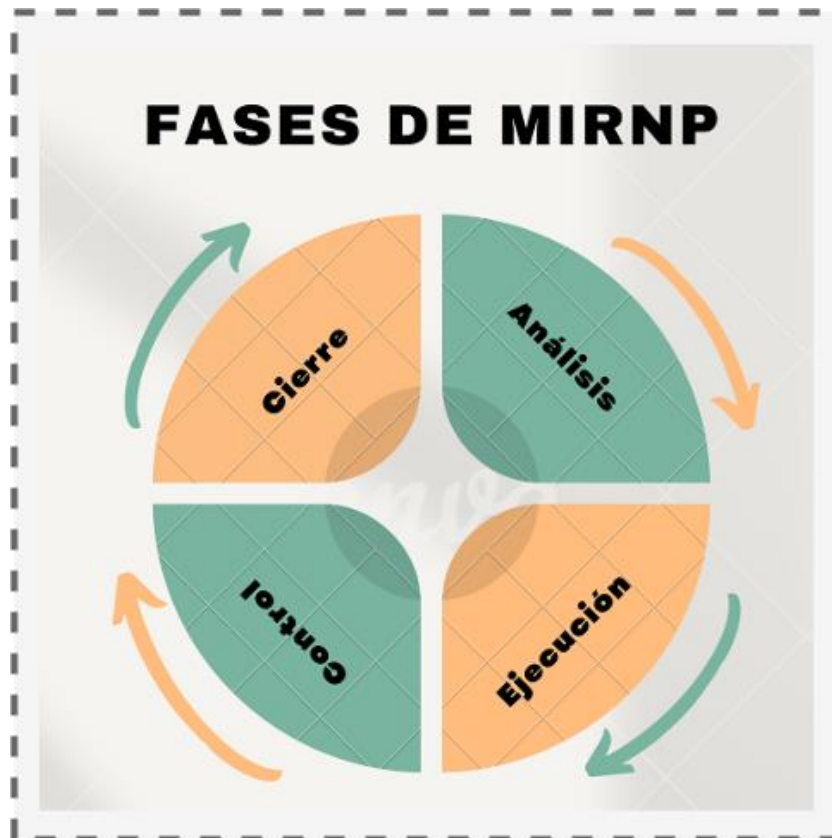


Figura 1: Fases de la Metodología.

Fuente: Elaborado por el Autor

3.2.1 CICLO DE VIDA

En la encuesta realizada al personal de la empresa MONKYAPP departamento TIC's se constata que utilizan la metodología SCRUM y DevOps para el desarrollo de software para no ocasionar conflictos se define el ciclo de vida Iterativo – Incremental para la metodología MIRNP.

Este ciclo de vida software permite que se repitan actividades del proyecto en cada iteración o fase en consecuencia se logra aumentar el conocimiento y entendimiento del producto solicitado. Cada ciclo desarrollado por el equipo de software anida y añade funcionalidad al producto.

3.2.2 DIAGRAMA ITERACIÓN Y RECURSIVIDAD EN LA APLICACIÓN DE PROCESOS.

En la Figura 11 se detalla la configuración para la ejecución de las fases que componen la MIRNP. Finalizada cada iteración se consigue un entregable que sirve como

recursividad para establecer y lograr alcanzar el producto final sea este un sistema software o un artefacto de software.

La ventaja de este ciclo de iteración en los procesos ayuda al equipo software incorporar cambios o retroalimentación en consecuencia ir disminuyendo la incertidumbre del producto solicitado.

Durante toda la ejecución de las fases de MIRNP se debe de mantener y tener en cuenta los documentos Análisis del Negocio, Mapa de StakeHolder, especificar Requisitos. Esto proporciona claridad en el contexto de los niveles de abstracción inferiores del sistema o software, también sirven para especificar las restricciones de diseños, rendimiento o elementos del sistema como las métricas y Kpi's.

La ejecución recursiva e iterativa de requisitos, su arquitectura y procesos de diseño se utilizan para capturar estos requisitos.

Teniendo en cuenta la estructura algunos requisitos no deben derivarse hasta que evoluciones de la arquitectura o el diseño, por que interactúan con elementos del sistema. Los requisitos de alto riesgo siempre deberán analizarse más a fondo.

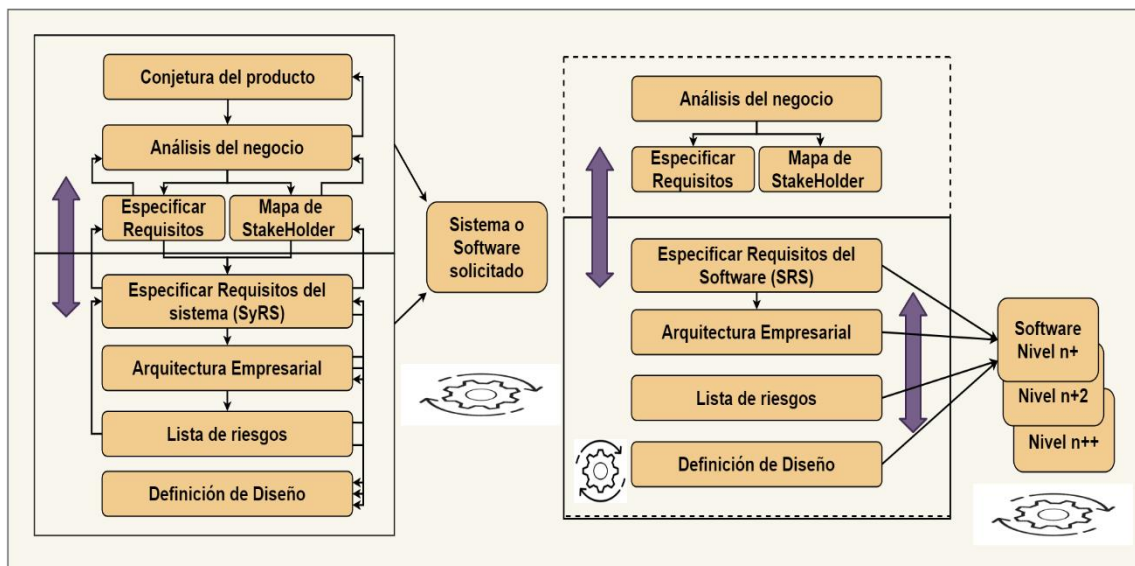


Figura 2: Iteración y recursividad en la aplicación de procesos, MIRNP.
Fuente: Adaptado por el Autor de Norma ISO/IEC/IEEE 29148:2018. [53]

3.2.3 ROLES GENÉRICOS DE MIRNP

<i>Roles Genéricos</i>	Definición
<i>Analista de Negocio</i>	Se encarga de realizar el análisis, interpretación de las necesidades empresariales para luego graficarlas y participar en el desarrollo funcional de la solución, puede ocupar otro rol dentro de MIRNP.
<i>Director de Desarrollo</i>	Líder del equipo del desarrollo. Se focaliza en clasificar y resumir las necesidades de la empresa o cliente.
<i>Equipo de desarrollo</i>	Grupo de personas profesionales en diferentes áreas de conocimiento, encargados en desarrollar el sistema o artefacto software.
<i>Cliente o Regulador</i>	Encargado de optimizar, maximizar y validar la calidad del producto software. Decide sobre características de diseño e Interfaz.

Figura 3: Roles Genéricos para MIRNP

Fuente: *Elaborado por el Autor*

3.2.4 ESTRUCTURA Y TRAZABILIDAD DE LA METODOLOGÍA PROPUESTA.

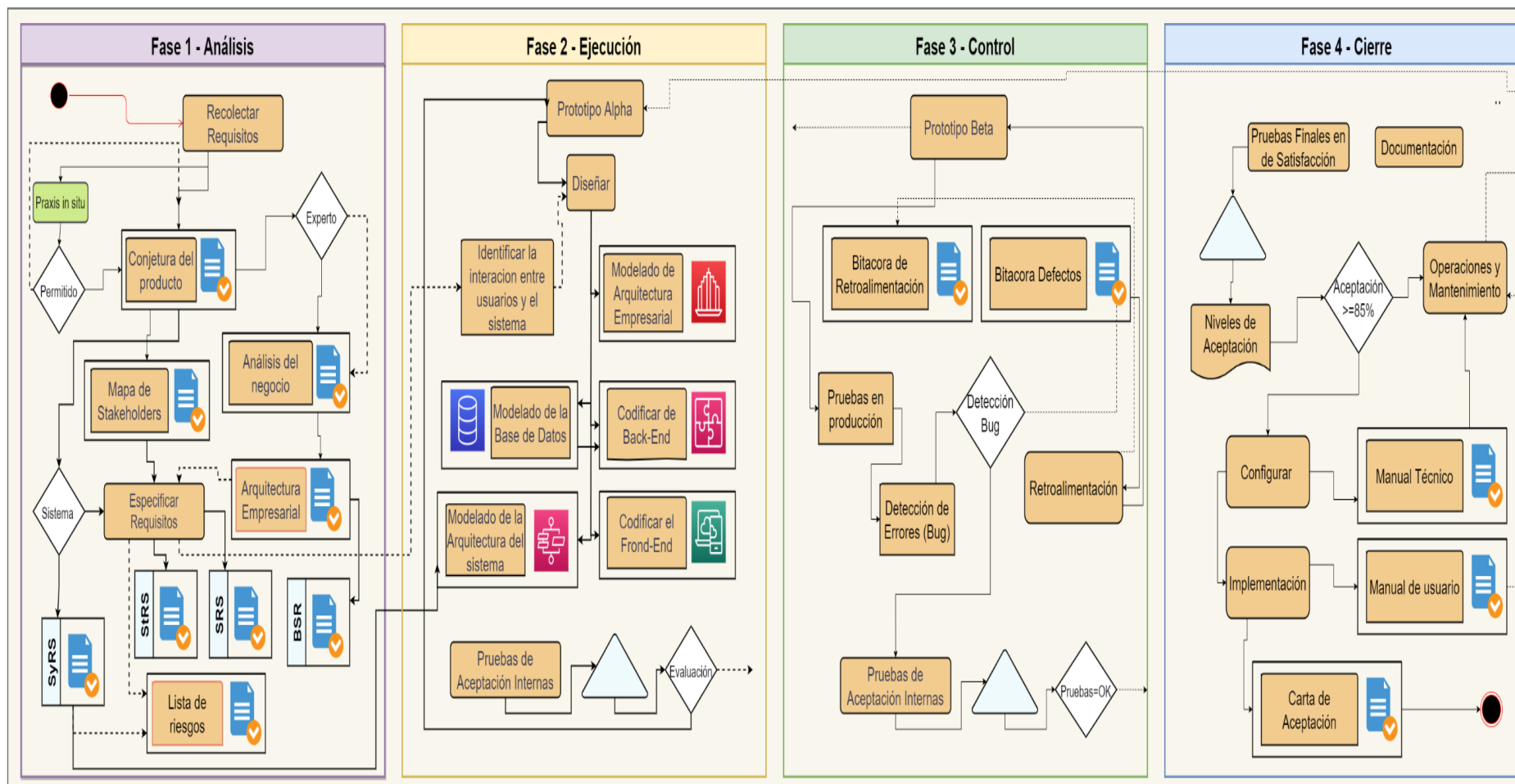


Figura 4: Estructura propuesta para MIRNP.
Fuente: Elaborado por el Autor

3.2.5 EXPLICACIÓN DEL DIAGRAMA ESTRUCTURA DE LA METODOLOGÍA MIRNP.

El resultado del primer objetivo específico se presenta que la empresa utiliza la Metodología SCRUM y DevOps para el desarrollo de productos software, esto se puede constatar en el Anexo 1. En consecuencia, la presente metodología se adapta a las antes mencionadas para establecer los tiempos y cronogramas de entrega de los módulos, artefactos software.

La estructura de la metodología alias MIRNP está compuesta por características y funcionalidades que se basaron en la norma ISO/IEC/IEEE 29148, Guia Babok 2.0 y el Framework Togaf.

Los documentos entregables desarrollados para MIRNP están subidos a la internet en el sitio <https://github.com> de acceso libre para que cualquier investigador interesado después pueda contribuir, refutar, utilizar la metodología.

La MIRNP tiene una trazabilidad vertical y horizontal secuencial permite al Equipo de Desarrollo y director de desarrollo retornar a una Fase u entregable para realizar las actualizaciones necesarias para cumplir satisfactoriamente el producto software.

3.2.6 FASE DE ANÁLISIS

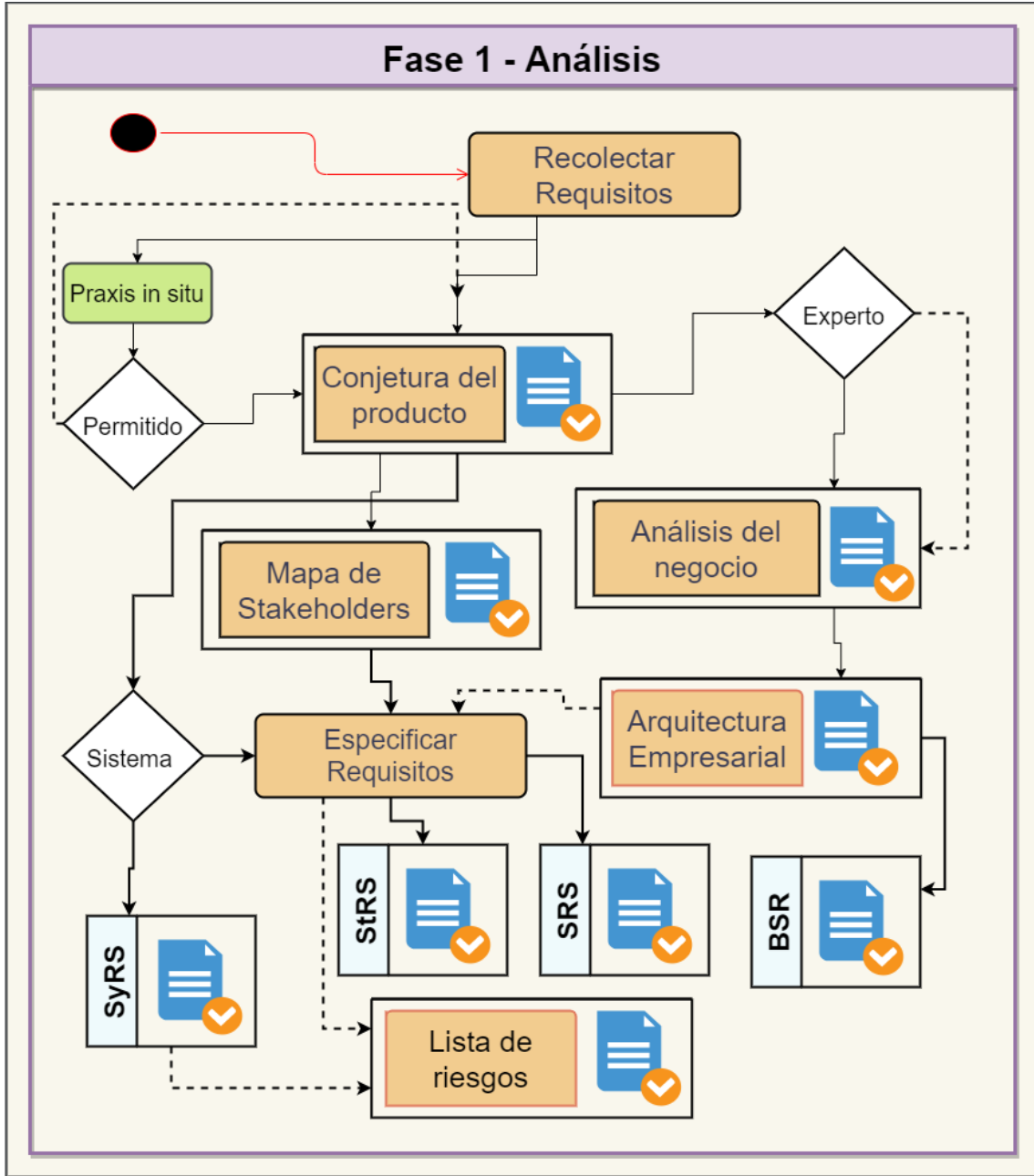


Figura 5: Fase de Análisis, MIRNP
Fuente: Elaborado por el Autor

3.2.6.1 CONJETURA DEL PRODUCTO.

Propósito

Establecer una hipótesis del posible producto solicitado por el cliente a la micro empresa de desarrollo software.

Descripción.

Interpretar las necesidades del negocio desde datos incompletos o una preposición que se supone cierta, La conjetura supone teóricamente que producto final es la solución a la necesidad de la empresa, cuando se demuestra la veracidad de la conjetura se establece la coherencia y su calidad.

Consideraciones de uso

Responsable: Director de Desarrollo, Analista del negocio.

Técnicas recomendadas a Utilizar: Observación, Entrevista, Praxis in situ (Opcional, Condicional).

Descarga: para acceder al documento y realizar esta tarea acceder al repositorio: [https://github.com/cDsants/MIRN/blob/entregables MIRN P/ConjeturaProducto.docx](https://github.com/cDsants/MIRN/blob/entregables%20MIRN/ConjeturaProducto.docx).

Diagrama de Tarea:

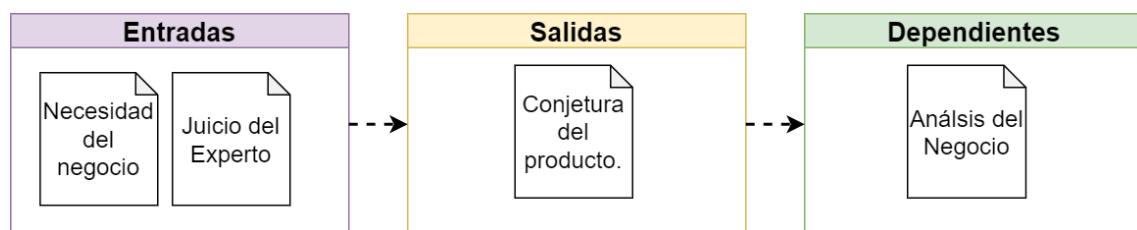


Figura 6: Fase de Análisis, Conjetura del Producto

Fuente: Elaborado por el Autor

3.2.6.2 ANÁLISIS DEL NEGOCIO

Propósito

Explicar y comprender cómo una empresa, institución u organización está funcionando para alcanzar sus objetivos y metas.

Descripción

Sintetizar toda la información proporcionada por los interesados que laboran en el negocio, personal del área administrativa, operativa y ejecutiva. Enfocados en los requisitos de las necesidades del negocio, estrechamente conectados a sus metas que conlleven a la caracterización del producto o solución [46] .

En la tabla 5 se muestra la estructura para la creación y adaptación de este entregable se tuvo en cuenta la estructura del framework Togaf, la Guía Babok 2.0 e ISO/IEC/IEEE 29148:2018:

	Framework Togaf	Guía Babok 2.0	ISO 29148:2018
Necesidad u Objeto Social	Fase preliminar	5.1	9.3.2
Ámbito del Negocio	Fase preliminar		9.3.3
Misión, Visión Empresarial	Fase B	5.1.4	9.3.7
Métricas e indicadores claves de desempeño		5.1.4	
Estructura Empresarial			9.3.15

Tabla 1: Estructura del documento entregable Análisis del Negocio.

Fuente: Elaborado por el Autor

Consideraciones de uso.

Responsable: Director de Desarrollo, Analista del negocio.

Técnicas recomendadas a Utilizar: Guía Babok 2.0 , Togaf, Smart.

Descarga: para acceder al documento para realizar esta tarea acceder al repositorio:

https://github.com/cDsants/M_I_R_N_P/blob/entregables_M_I_R_N_P/AnalisisNegocio.docx

Diagrama de Tarea:

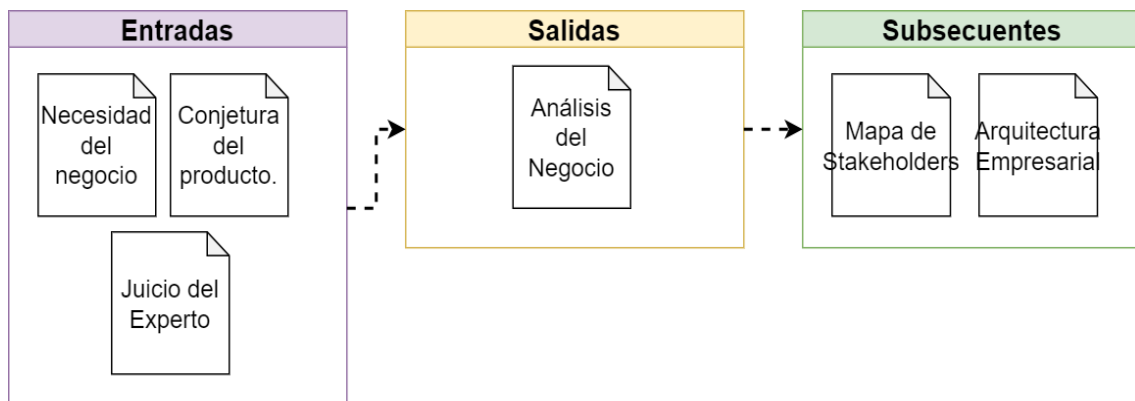


Figura 7: Fase de Análisis, Análisis del Negocio.

Fuente: Elaborado por el Autor

3.2.6.3 ARQUITECTURA EMPRESARIAL

Propósito

Identificar los principales componentes de la organización (empresa, negocio u institución), establecer la relación coexistente entre los objetivos del negocio y el producto software solicitado.

Descripción

Arquitectura Empresarial o (EA) es el modelo o diseño conceptual para el análisis, planificación, diseño e implementación empresarial a través de técnicas y metodologías

que logran satisfacer las necesidades del negocio con crecimiento rentable y estratégica [44] [16].

En la tabla 6 se muestra la estructura para la creación y adaptación de este entregable se tuvo en cuenta la estructura del framework Togaf: Arquitectura de Negocio (Fase B), Arquitectura de Sistemas de Información (Fase C), Arquitectura de Tecnología (Fase D). También se introdujo un apartado para Definir el caso del Negocio extraído y adaptado de la guía babok 2.0.

	Framework Togaf	Guía Babok 2.0
Arquitectura de Negocio	Fase B	
Arquitectura de Sistemas de Información	Fase C	
Arquitectura de Tecnología	Fase D	
Definir el caso del Negocio		5.5

Tabla 2: Estructura del documento entregable Arquitectura Empresarial.

Fuente: Elaborado por el Autor

Consideraciones de uso

Responsable: Director de Desarrollo, Analista del negocio.

Técnicas recomendadas a Utilizar: Método de desarrollo de arquitectura, Guía Babok 2.0 , Togaf.

Descarga: para acceder al documento y realizar esta tarea acceder al repositorio:

<https://github.com/cDsants/MIRN/blob/entregablesMIRN/ArquitecturaEmpresarial.docx>

Diagrama de Tarea:

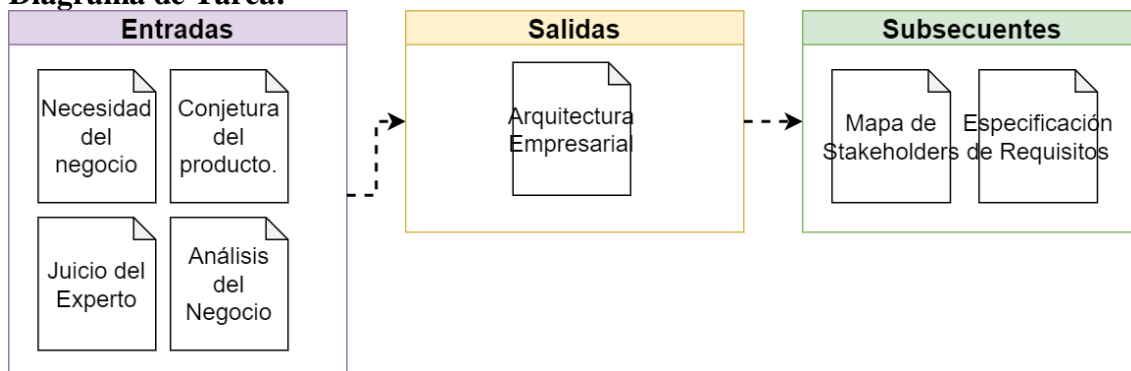


Figura 8: Fase de Análisis, Arquitectura Empresarial.

Fuente: Elaborado por el Autor

3.2.6.4 MAPA DE STAKEHOLDERS

Propósito

Identificar a las personas que intervienen con la solución o producto software.

Descripción

Es considerado como el primer paso para la gestión de personas o grupos interesados en lograr una meta u objetivo, este permite definir la relación, poder e influencia de cada uno de los involucrados.

Crear este entregable permite mejorar la comunicación y mantener satisfecho de forma eficaz. Los pasos a desarrollar son:

- Identificar a los interesados de la solución.
- Clasificar a los interesados.
- Agregar a los interesados o partes con alta influencia.
- Integrar a las personas que debo administrar de cerca.
- Incluir a los interesados de baja influencia con interés.
- Añadir a los interesados de baja influencia y poco interés.

Nota: También se consideró de la ISO/IEC/IEEE 29148:2018 el apartado 9.3.5.

Consideraciones de uso

Responsable: Director de Desarrollo, Equipo de desarrollo.

Técnicas recomendadas a Utilizar: stakeholder mapping.

Descarga: para acceder al documento para realizar esta tarea acceder al repositorio:

https://github.com/cDsants/M_I_R_N_P/blob/entregables_M_I_R_N_P/MapaStakeholder.docx

Diagrama de Tarea:

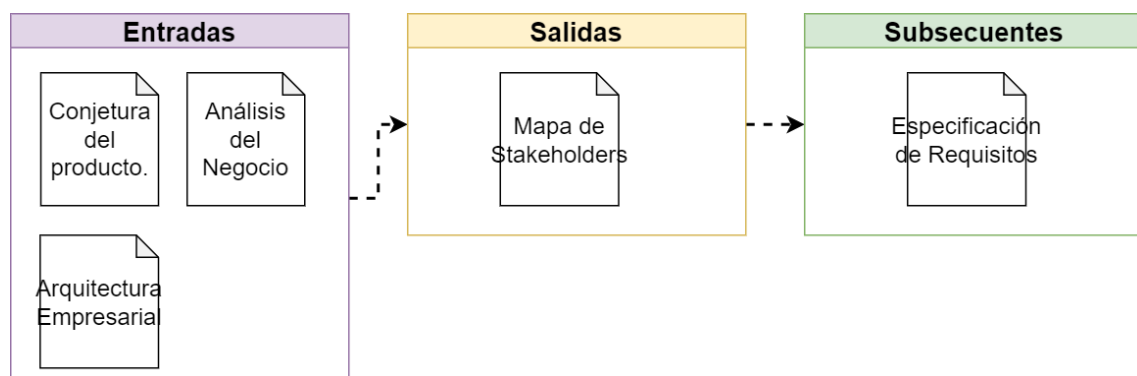


Figura 9: Fase de Análisis, Mapa de Stakeholders.

Fuente: Elaborado por el Autor

3.2.6.5 ESPECIFICACIÓN DE REQUISITOS EMPRESARIALES (BRS)

Propósito

Describir cual es la meta de la organización, por qué se va a adaptar a un nuevo entorno, cuáles son las necesidades de sus usuarios, personal operativo o administrativo.

Descripción

La especificación de Requisitos empresariales (BRS) es información proporcionada y especificada por la empresa. El área administrativa es la responsable de ayudar y aportar en la elaboración de este entregable. Además, el BRS sirve como parte primordial para la participación activa de los interesados de la solución [53].

En la tabla 7 se especifica la estructura para la creación de este entregable también se adapta el esquema ejemplo que proporciona la norma ISO/IEC/IEEE 29148:2018 en su literal 8.2.2.

	Framework Togaf	ISO/IEC/IEEE 29148:2018
Establecer procesos del negocio	Fase B	9.3.10
Políticas, reglas y limitaciones de los procesos del negocio		9.3.11
Modelo de Negocio	Fase B	9.3.8
Establecer métricas operativas de calidad del negocio		9.3.14

Tabla 3: Estructura del documento entregable Especificación de Requisitos Empresariales (BRS).

Fuente: Elaborado por el Autor

Consideraciones de uso

Responsable: Consultor, Analista de negocio, Director de Desarrollo.

Técnicas recomendadas a Utilizar: ISO/IEC/IEEE 29148:2018, UML, C4.

Descarga: para acceder al documento y realizar esta tarea acceder al repositorio:

[https://github.com/cDsants/MIRNP/blob/entregables/MIRNP/EspecificaciónRequisitosEmpresariales%20\(BRS\).docx](https://github.com/cDsants/MIRNP/blob/entregables/MIRNP/EspecificaciónRequisitosEmpresariales%20(BRS).docx)

Diagrama de Tarea:

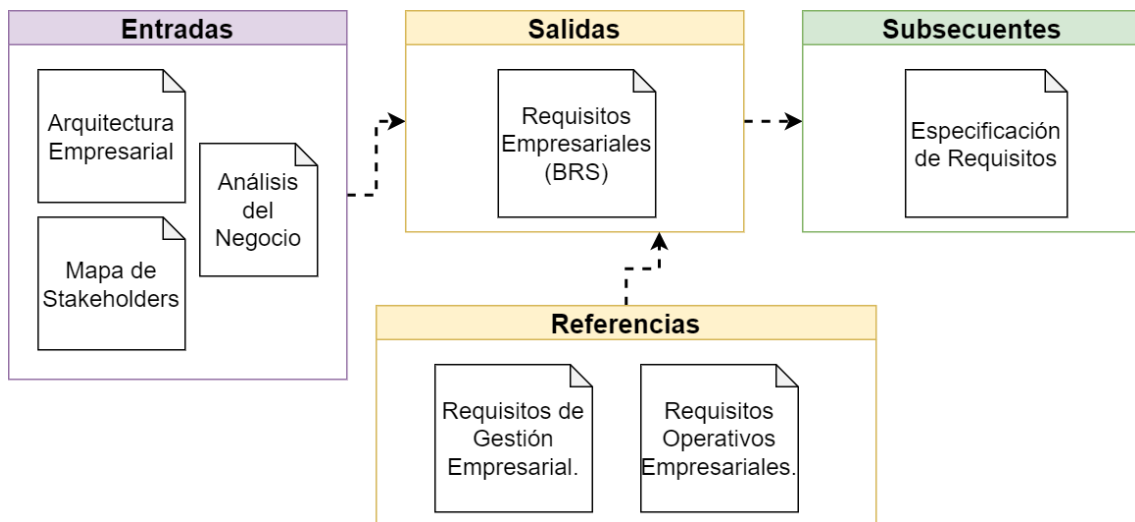


Figura 10: Fase de Análisis, Especificación de Requerimientos Empresariales (BRS)

Fuente: Elaborado por el Autor

3.2.6.6 ESPECIFICACIÓN DE REQUISITOS DEL SISTEMA (SYRS)

Propósito

Proporcionar la descripción de que funcionalidad va a tener el sistema, refiriéndose a interacciones o interfaces del sistema con el entorno externo.

Descripción

Tradicionalmente el SyRS comunica en un documento los requisitos del cliente a los desarrolladores del sistema.

El SyRS puede tomar la forma de un documento físico o digital, modelos, prototipos o la combinación de todas las anteriores. El SyRS presenta los resultados la definición de la necesidad, el concepto operativo del sistema y la arquitectura.

Se considera un problema de creación de sistemas a la comunicación entre los subgrupos dentro de los grupos de desarrollo.

Para la creación de este entregable se adapta al esquema ejemplo que proporciona la norma ISO/IEC/IEEE 29148:2018 en su literal 9.5.

	Guía Babok 2.0	ISO/IEC/IEEE 29148:2018
Propósito del sistema	5.1-5.4	9.5.3
Modos y estados del sistema		9.5.10
Requisitos de políticas y reglamentos		9.5.15
Requisitos de embalaje, manipulación, envío y transporte		9.5.17
Acrónimos y Abreviaturas		5.2

Tabla 4: Estructura del documento entregable Especificación de Requisitos Sistema (SyRS).

Fuente: Elaborado por el Autor

Responsable: Director de Desarrollo, Equipo de desarrollo

Técnicas recomendadas a Utilizar: ISO/IEC/IEEE 29148:2018, UML, C4.

Descarga: para acceder al documento para realizar esta tarea acceder al repositorio:

[https://github.com/cDsants/M_I_R_N_P/blob/entregables_M_I_R_N_P/Especificacion%20de%20Requisitos%20del%20Sistema%20\(SyRS\).xlsx](https://github.com/cDsants/M_I_R_N_P/blob/entregables_M_I_R_N_P/Especificacion%20de%20Requisitos%20del%20Sistema%20(SyRS).xlsx)

Diagrama de Tarea:

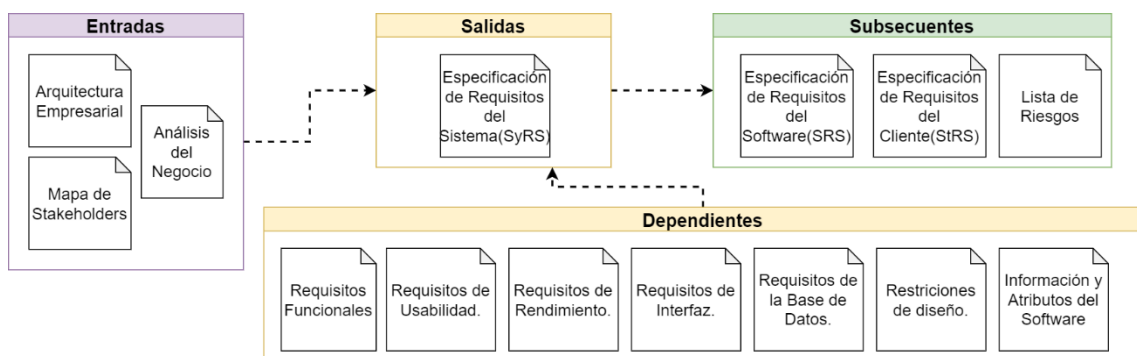


Figura 11: Fase de Análisis, Especificación de Requerimientos del Sistema.

Fuente: Elaborado por el Autor

3.2.6.7 ESPECIFICACIÓN DE REQUISITOS DEL CLIENTES (STRS)

Propósito

Describe el cómo la organización utilizara el sistema a fin de contribuir a la empresa y en contexto a lo descrito en la especificación de requisitos empresariales (BSR).

Descripción

La Especificación de Requisitos de los interesados (StRS) describe el por qué la empresa está desarrollando, integrando o actualizando un sistema, en este entregable se definen los procesos y reglas(restricciones) que dependen al sistema.

También se documenta los requisitos del nivel jerárquico superior desde el paradigma de todas las partes interesadas, esto incluye las necesidades de los usuarios, área operativa, área administrativa.

En la tabla 9 se determinan los literales tomados para la creación de este entregable se adapta al esquema ejemplo que proporciona la norma ISO/IEC/IEEE 29148:2018 en su literal 8.3.2.

	Guía Babok 2.0	ISO/IEC/IEEE 29148:2018
Especificación de requisitos de Interesados (StRS)	3.1	9.4.15
Transformar necesidades en requisitos	3.2	5.2.3
Construcción de Requisitos		5.2.4
Requisitos criterios de Lenguaje		5.2.7
Características individuales de los requisitos.		5.2.5
Atributos de los requisitos		5.2.8.2

Tabla 5: Estructura del documento entregable Especificación de Requisitos del cliente (StakeHolder) (StRS).

Fuente: Elaborado por el Autor

Consideraciones de uso

Responsable: Director de Desarrollo, Equipo de Desarrollo.

Técnicas recomendadas a Utilizar: ISO/IEC/IEEE 29148:2018, UML, encuestas, praxis in situ (opcional - condicional), observación, entrevistas.

Descarga: para acceder al documento y realizar esta tarea acceder al repositorio:
[https://github.com/cDsants/MIRN/blob/entregables_MIRN_P/Especificacion%20de%20Requisitos%20de%20los%20Interesados\(StakeHolder\)%20\(StRS\).xlsx](https://github.com/cDsants/MIRN/blob/entregables_MIRN_P/Especificacion%20de%20Requisitos%20de%20los%20Interesados(StakeHolder)%20(StRS).xlsx)

Diagrama de Tarea:



Figura 12: Fase de Análisis, Especificación de Requisitos del Clientes.

Fuente: Elaborado por el Autor

3.2.6.8 ESPECIFICACIÓN DE REQUISITOS DEL SOFTWARE (SRS)

Propósito

Especificación de requisitos software provee la determinación para un producto software, programa o conjuntos de aplicaciones que realizan procesos de un entorno específico.

Descripción

La SRS desempeña parte fundamental en el desarrollo del producto software y en el plan total del proyecto, en consecuencia, se definen esencialmente la funcionalidad del proyecto o en algunos casos para de un sistema para luego ser integrado. También se establecen los requisitos de rendimiento y funcionalidad con la pertinencia del producto. Software.

Otro elemento a considerar es la perspectiva del producto es este apartado se describe la interacción existente entre el software y sistemas más grandes.

La SRS puede incluir todas las necesidades que los interesados requieran, representantes de los proveedores, etc.

En la tabla 10 se determinan los literales tomados para la creación de este entregable se adapta al esquema ejemplo que proporciona la norma ISO/IEC/IEEE 29148:2018 en su literal 8.5.2.

Especificación de requisitos del Software (SRS)	3.1	8.5
Acrónimos y Abreviaturas		9.6.8
Construcción de Requisitos		5.2.4
Interfaces de Hardware	4.2 – 6.3	9.6.4.3
Perspectiva del Producto	9.26	9.6.4
Limitaciones	6.4	9.6.7

Tabla 6: Estructura del documento entregable Especificación del Software (SRS).

Fuente: Elaborado por el Autor

Consideraciones de uso

Responsable: Director de Desarrollo, Equipo de Desarrollo.

Técnicas recomendadas a Utilizar: ISO/IEC/IEEE 29148:2018, UML, C4, encuestas, praxis in situ (opcional - condicional), observación, entrevistas.

Descarga: para acceder al documento y realizar esta tarea acceder al repositorio:

[https://github.com/cDsants/M_I_R_N_P/blob/entregables_M_I_R_N_P/Especificacion%20de%20Requisitos%20del%20Software%20\(SRS\).xlsx](https://github.com/cDsants/M_I_R_N_P/blob/entregables_M_I_R_N_P/Especificacion%20de%20Requisitos%20del%20Software%20(SRS).xlsx)

Diagrama de Tarea:

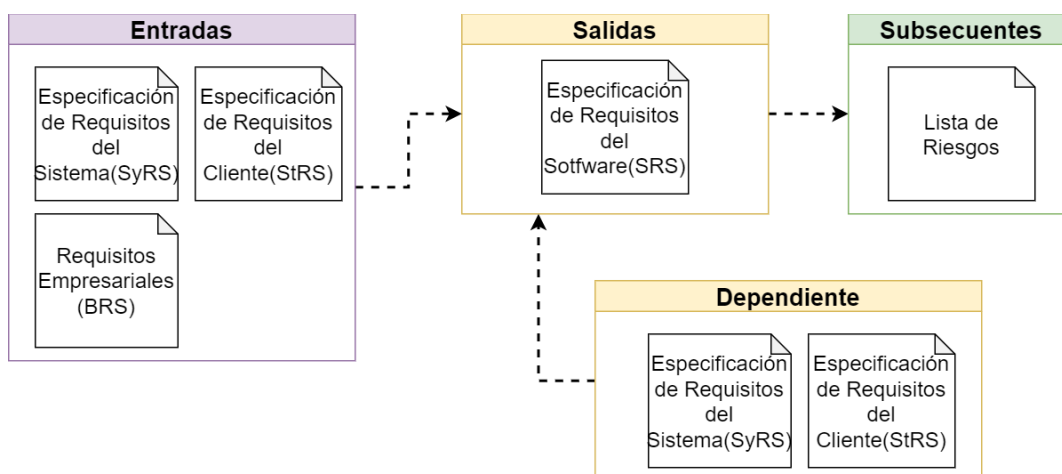


Figura 13: Fase de Análisis, Especificación de Requisitos del Software

Fuente: Elaborado por el Autor

3.2.6.9 LISTA DE RIESGOS.

Propósito

Identificar los posibles riesgos para el desarrollo del producto o sistema software.

Descripción

Las metodologías ágiles de desarrollo son actividades coordinadas con el objetivo del producto software en el menor tiempo, en algunas ocasiones no se planifica o determina

los riesgos que se pueden encontrar durante la ejecución de las tareas, como por ejemplo falta de compatibilidad, errores humanos, falta de documentación, requisitos no elicitados correctamente, deficiencia en la estimación de tiempo de entrega.

Después de tener todas las entradas para este documento se procede a sintetizar e identificar los posibles riesgos, el equipo de respuesta ante la respuesta a uno incidente.

En la tabla 11 se determinan los literales tomados para la creación de este entregable se adapta al esquema ejemplo que proporciona la norma ISO/IEC/IEEE 29148:2018 en su literal A.2.8.2.

	Guía Babok 2.0	ISO/IEC/IEEE 29148:2018
Especificación de requisitos del Software (SRS)	2.5.4.6	8.5
Acrónimos y Abreviaturas		9.6.8
Construcción de Requisitos		5.2.4
Interfaces de Hardware	4.2 – 6.3	9.6.4.3
Perspectiva del Producto	9.26	9.6.4
Limitaciones	6.4	9.6.7

Tabla 7: Estructura del documento entregable Lista de Riesgos.

Fuente: Elaborado por el Autor

Consideraciones de uso

Responsable: Director de Desarrollo, Equipo de Desarrollo.

Técnicas recomendadas a Utilizar: ISO/IEC/IEEE 29148:2018 (Gestión de Riesgos).

Descarga: para acceder al documento y realizar esta tarea acceder al repositorio:

[https://github.com/cDsants/MIRN/blob/entregables MIRN/ListaRiesgos.dOCX](https://github.com/cDsants/MIRN/blob/entregables/MIRN/ListaRiesgos.dOCX)

Diagrama de Tarea:

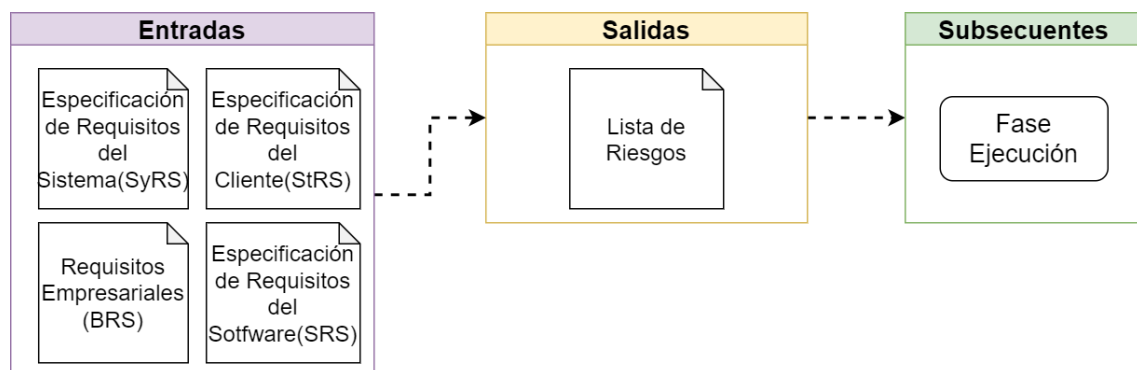


Figura 14: Fase de Análisis, Lista de Riesgos.

Fuente: Elaborado por el Autor

3.2.7 FASE 2 - EJECUCIÓN

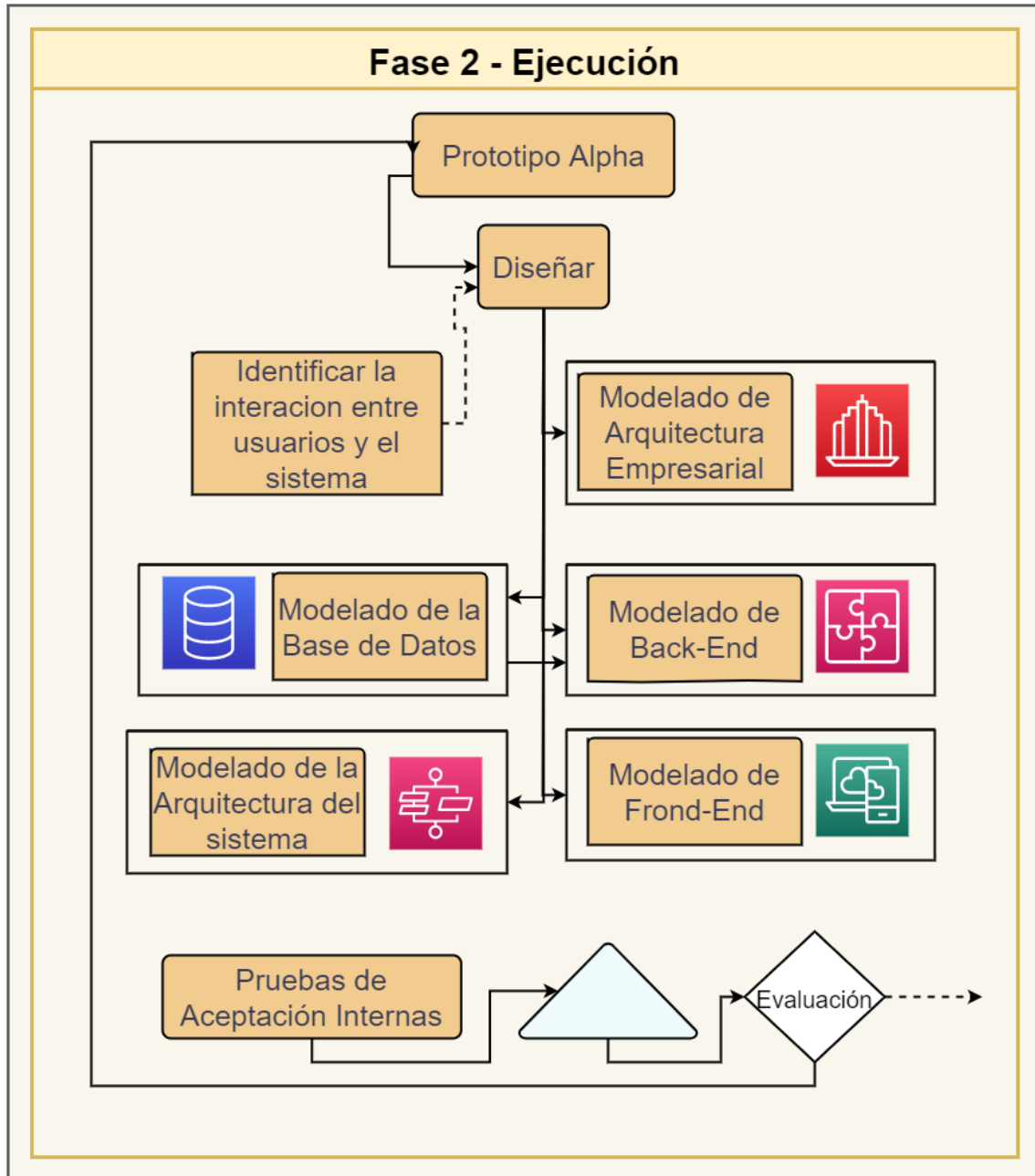


Figura 15: Fase de Ejecución, MIRNP

Fuente: Elaborado por el Autor

3.2.7.1 IDENTIFICAR INTERACCIÓN ENTRE USUARIOS Y EL SOFTWARE

Propósito

Establecer el diagrama de interacciones entre los usuarios y el software.

Descripción

Diseñar la Interacción permite comprender y definir la estructura y comportamiento de los nodos, módulos y métodos del sistema.

Consideraciones de uso

Responsable: Director de Desarrollo, Equipo de Desarrollo.

Técnicas recomendadas a Utilizar: UML (Diagrama de interacción, Diagramas de secuencia, Diagramas UML de comunicación).

Descarga: para acceder al documento y realizar esta tarea acceder al repositorio: https://github.com/cDsants/M_I_R_N_P/blob/entregables_M_I_R_N_P/Fase2-Ejecucion/InteraccionUsuarios.docx

Diagrama de Tarea:

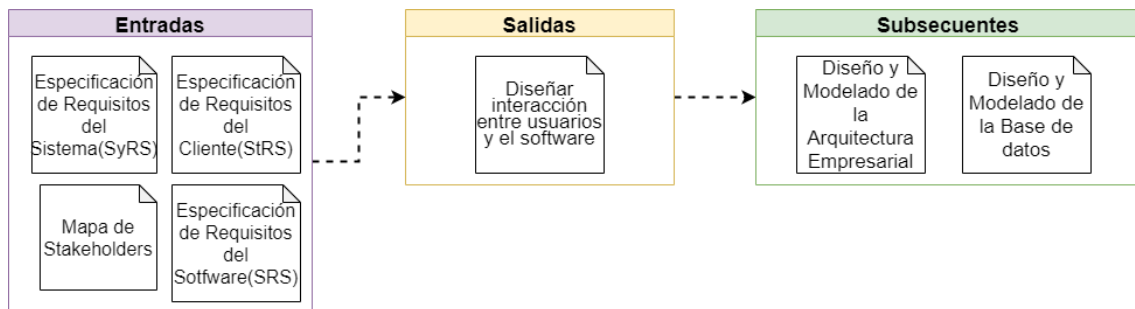


Figura 16: Interacción entre Usuarios y software

Fuente: Elaborado por el Autor

3.2.7.2 DISEÑO Y MODELADO DE LA ARQUITECTURA EMPRESARIAL

Propósito

Presentar la figura con el perfil de autenticidad que identifica el cumplimiento del requerimiento del cliente.

Descripción

El proveedor adapta el modelo de base de datos conforme lo solicita la empresa contratante para el servicio de sus clientes, como también se establece el prototipo de modelo a usar en la presentación de los datos.

Consideraciones de uso

Responsable: Analista de negocio, Director de Desarrollo.

Técnicas recomendadas a Utilizar: Togaf, Mapa de Capacidades del Negocio.

Descarga: para acceder al documento y realizar esta tarea acceder al repositorio: https://github.com/cDsants/M_I_R_N_P/blob/entregables_M_I_R_N_P/Fase2-Ejecucion/MODELADO_DE_ARQUITECTURA_EMPRESARIAL.docx

Diagrama de Tarea:

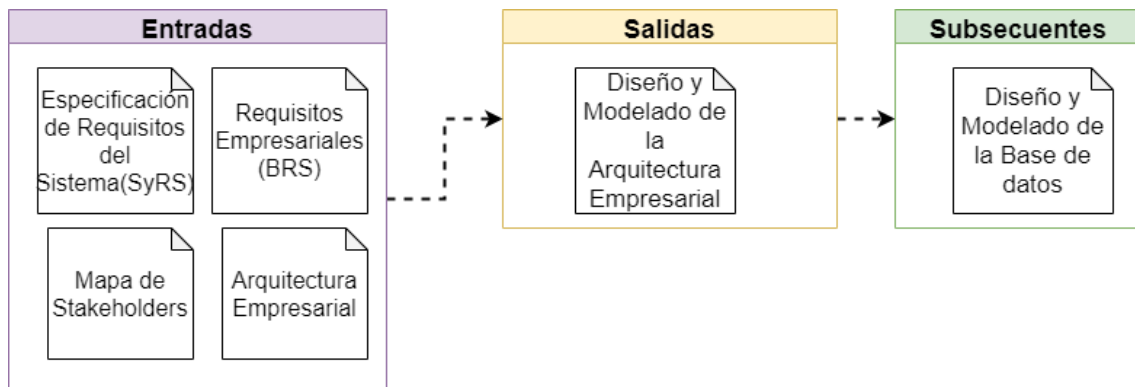


Figura 17: Modelado Arquitectura Empresarial

Fuente: Elaborado por el Autor

3.2.7.3 DISEÑO Y MODELADO DE BASE DE DATOS

Propósito

Graficar y construir a partir de las necesidades de la solución, El diseñador de la Base de datos determina el tipo, estructura y formato de la base de datos, cantidad de tablas y que datos deben almacenarse.

Descripción

El administrador de la empresa desarrolladora de software presenta la estructura lógica de acuerdo al tipo de requerimiento del cliente. Con la ayuda de los SyRS, StRS, SRS se describe y determina el tipo de base de datos a utilizar en la solución además de especificar los factores, reglas y limitaciones para la base de datos.

Consideraciones de uso

Responsable: Equipo de desarrollo.

Técnicas recomendadas a Utilizar: Modelado de datos, Diagrama entidad-relación, Modelado de datos relacional y no relacional NoSQL.

Descarga: para acceder al documento y realizar esta tarea acceder al repositorio:

https://github.com/cDsants/MIRNP/blob/entregables/MIRNP/Fase2-Ejecucion/D_Modelado_BD.docx

Diagrama de Tarea:

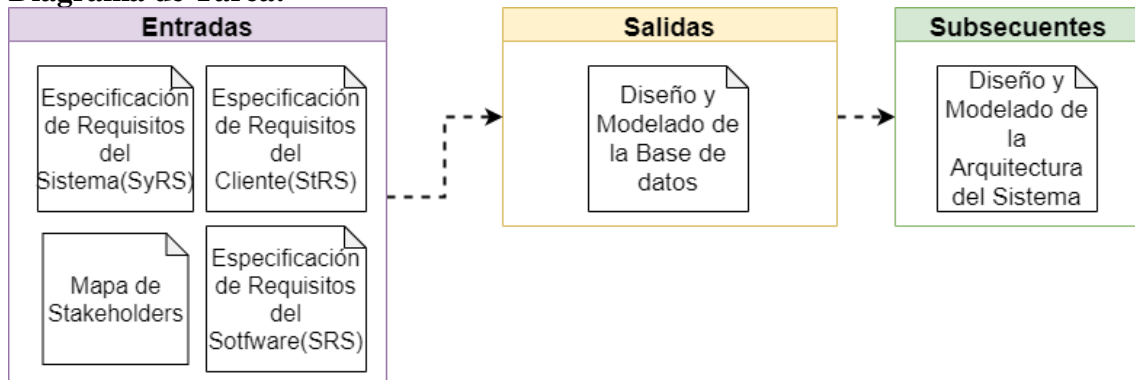


Figura 18: Modelado Base de datos
Fuente: Elaborado por el Autor

3.2.7.4 DISEÑO Y MODELADO DE LA ARQUITECTURA DEL SISTEMA.

Propósito

Establecer un modelo original con una arquitectura propia a las necesidades específicas del cliente.

Descripción

El diseño original de cada cliente contiene las particularidades de la producción del servicio o producto ofertado, por lo cual se plantea en esta etapa las características de la empresa, proveedores, clientes, servicios o productos, clientes potenciales. Esto está en estrecha correspondencia a la fase de análisis.

Se describe conceptualmente todas las vistas, estructuras y componentes internos y externos con los que la solución interactúa.

Consideraciones de uso

Responsable: Analista de negocio, Director de Desarrollo.

Técnicas recomendadas a Utilizar: Togaf, Mapa de Capacidades del Negocio, Diagramas UML, Diagramas C4, Patrones.

Descarga: para acceder al documento y realizar esta tarea acceder al repositorio:

https://github.com/cDsants/M_I_R_N_P/blob/entregables_M_I_R_N_P/Fase2-Ejecucion/D_Modelado_ArquiSistema.docx

Diagrama de Tarea:

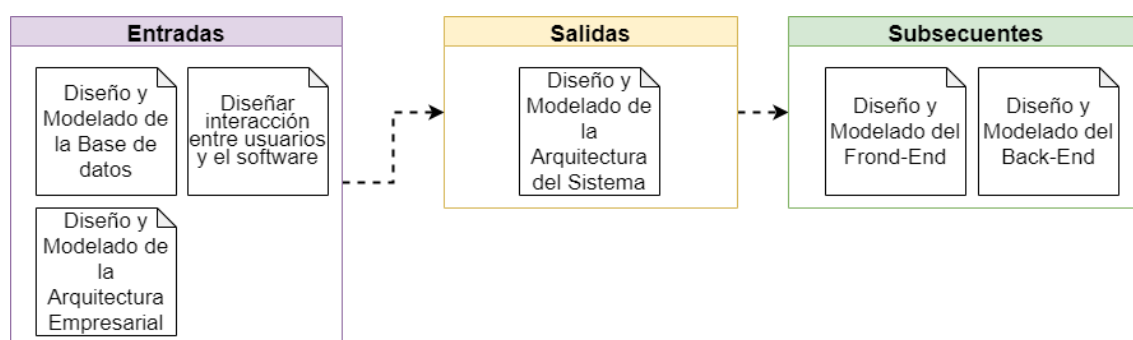


Figura 19: Modelado Arquitectura del Sistema

Fuente: Elaborado por el Autor

3.2.7.5 PROTOTIPAR DE INTERFAZ DEL SOFTWARE – FRONTEND.

Propósito

Dinamizar la presentación y explorar los aspectos interactivos del sistema, como son sus cualidades usabilidad, accesibilidad y su funcionalidad.

Descripción

Al prototipar la interfaz del usuario se crea una simulación interactiva y original de como funcionan los procesos del producto solicitado y validar que los requisitos están en

concordancia con su idea, en la presentación de datos está basada en el clásico y de uso común, front-end, debido que su presentación es directa y dinámica, permite probar, ajustar y perfeccionar los elementos con los que trabajan o interactúa el usuario y el producto o solución.

Consideraciones de uso

Responsable: Director de Desarrollo, Equipo de desarrollo.

Técnicas recomendadas a Utilizar: Mapa de Diseño UX, Patrones, Wireframe.

Descarga: para acceder al documento y realizar esta tarea acceder al repositorio: https://github.com/cDsants/M_I_R_N_P/blob/entregables_M_I_R_N_P/Fase2-Ejecucion/FRONTEND.docx

Diagrama de Tarea:

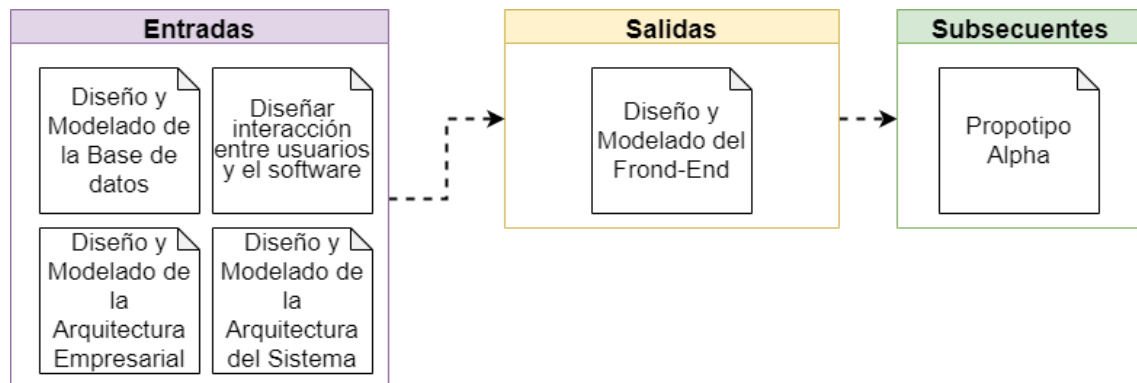


Figura 20: Prototipar la interfaz del software (Front_end)

Fuente: Elaborado por el Autor

3.2.7.6 PROTOTIPO LÓGICO DEL SOFTWARE – BACKEND.

Propósito

Presentar de forma lógica y sistemática los datos requeridos por el cliente, entre ellos los esenciales de la comercialización como proveedor, clientes y producto.

Descripción

El prototipado lógico del software o BACKEND es la capa de acceso a los datos además de acceder al servidor, a diferencia del Front End que permite que es la capa del software con la que interactúan el cliente.

El BackEnd posee secuencias lógicas de forma cronológicas y dinámica, más, sin embargo, tiene la particularidad de permitir un acceso desde cualquier dispositivo electrónico con acceso a la web o aplicación del sistema propuesto.

Consideraciones de uso

Responsable: Director de Desarrollo, Equipo de desarrollo.

Técnicas recomendadas a Utilizar: C4, Diagramas UML.

Descarga: para acceder al documento y realizar esta tarea acceder al repositorio: https://github.com/cDsants/M_I_R_N_P/blob/entregables_M_I_R_N_P/Fase2-Ejecucion/BACKEND.docx

Diagrama de Tarea:

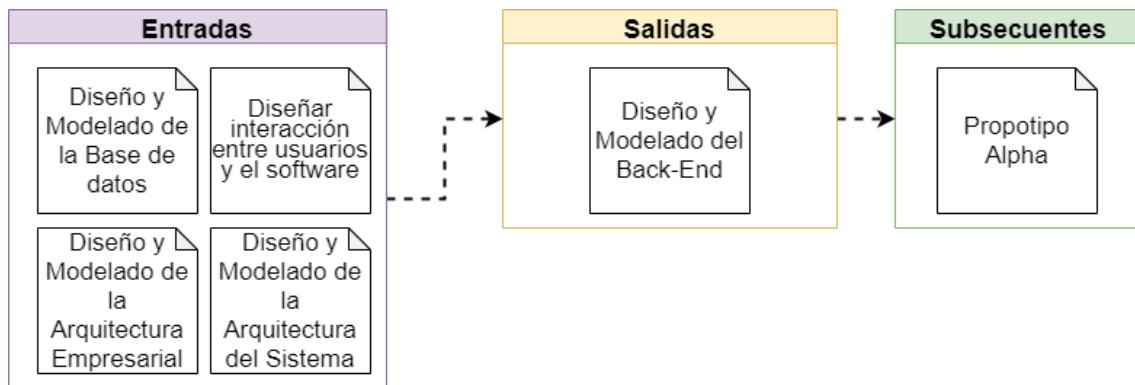


Figura 21: Prototipo lógico - Back_End

Fuente: Elaborado por el Autor

3.2.7.7 PROTOTIPO ALPHA

Propósito

Presentar la ejecución del procesamiento de los datos en un esquema salida lógico interactivo único para la solución ofertada al cliente.

Descripción

Se presenta el producto ofertado para verificar si cumple con los requisitos que solicita la empresa y que estén acorde a las expectativas del cliente. De tal manera se automatice el procesamiento de los datos por parte de la organización, dado que agiliza y dinamiza los procesos. Posterior a esta fase, se da por aprobado la ejecución y se da paso a la fase de control.

El prototipo Alpha es el entregable verificable de esta tarea. este es la primera versión del producto, carece de funcionalidad completa, sirve para establecer dirección al producto y eliminar aspectos, factores o características que no se hayan comprendido en la ingeniería de requisitos.

Consideraciones de uso

Responsable: Equipo de desarrollo, Interesados.

Diagrama de Tarea:

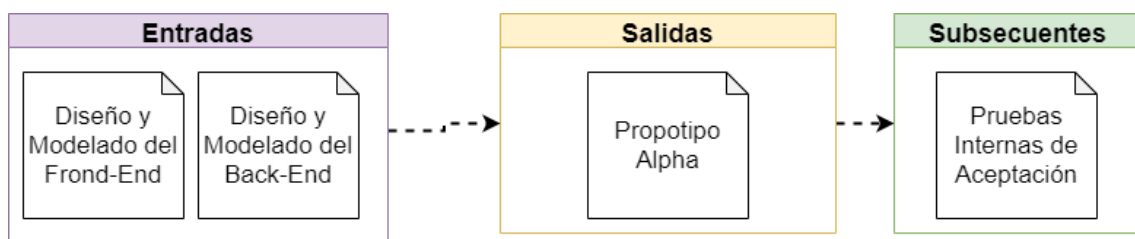


Figura 22: Prototipo Alpha

Fuente: Elaborado por el Autor

3.2.7.8 PRUEBAS DE ACEPTACIÓN INTERNAS.

Propósito

Cumplir con la necesidad del cliente interesado en adquirir el producto sistema/software.

Descripción

Consideraciones de uso

Responsable: Equipo de desarrollo.

Técnicas recomendadas a Utilizar: Modelado de datos, Diagrama entidad-relación, Modelado de datos NoSQL

Descarga: para acceder al documento y realizar esta tarea acceder al repositorio:

https://github.com/cDsants/M_I_R_N_P/blob/entregables_M_I_R_N_P/Fase2-Ejecucion/PRUEBAS_ACEPTACIÓN_INTERNAS.docx

Diagrama de Tarea:

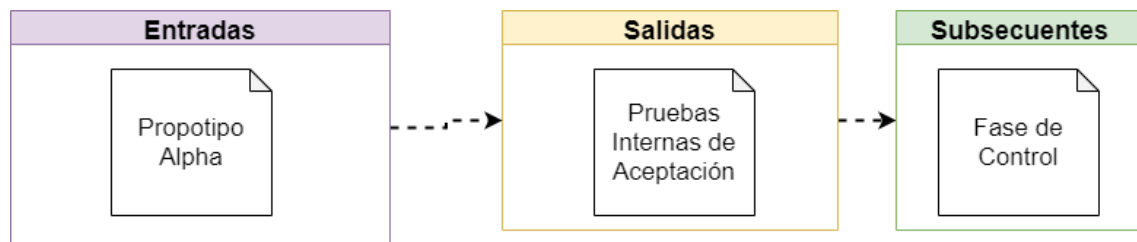


Figura 23: Pruebas Internas de Aceptación

Fuente: Elaborado por el Autor

3.2.8 FASE 3 – CONTROL

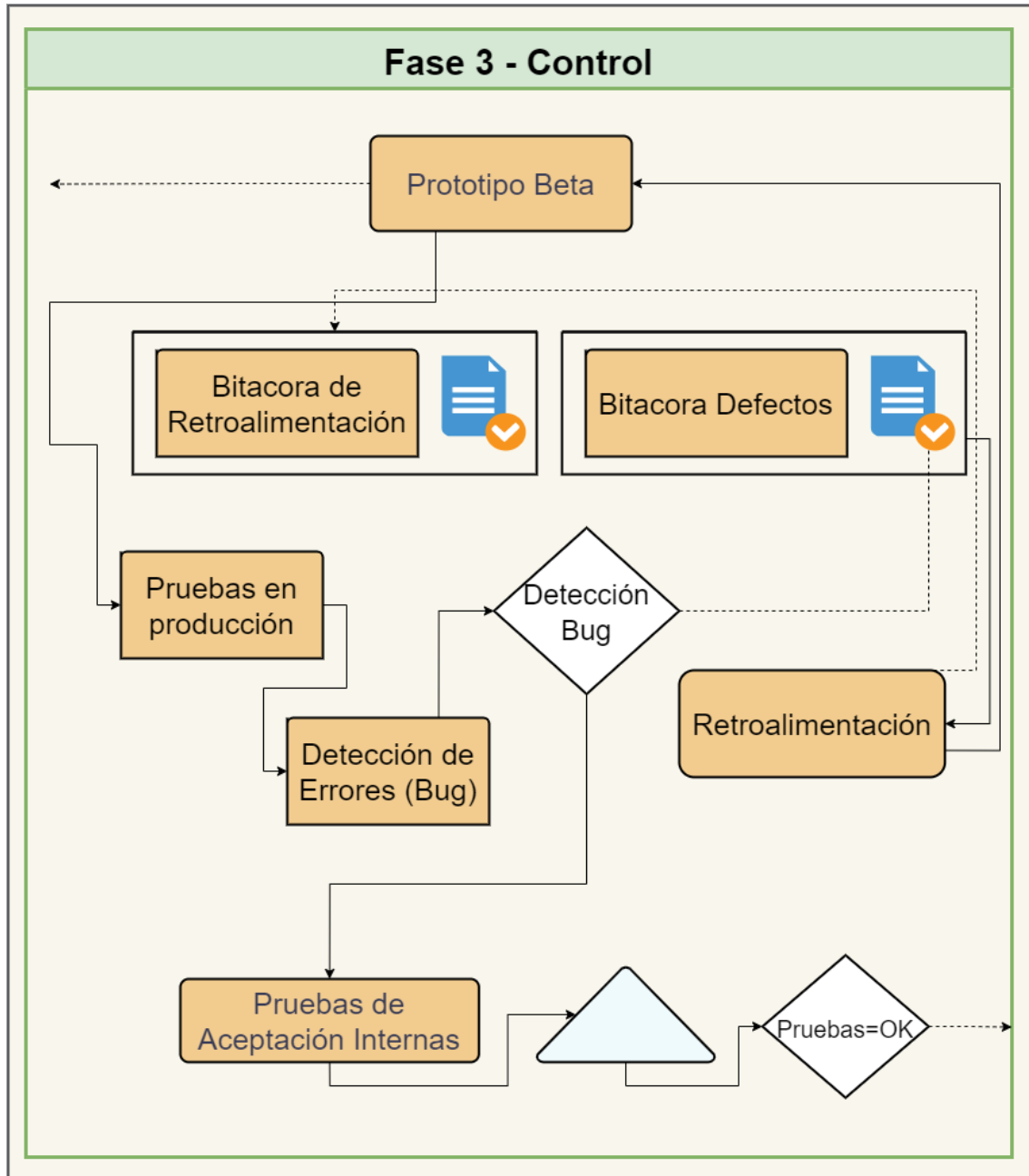


Figura 24: Fase de Control de MIRNP

Fuente: Elaborado por el Autor

3.2.8.1 BITÁCORA DE RETROALIMENTACIÓN.

Propósito

Brindar información estructural y cronológica del proceso de ingreso, ejecución y evaluación de la propuesta previo el diseño de software.

Descripción

Las bitácoras permiten observar los procesos utilizados al momento de la elaboración de la propuesta. Además, es un registro ordenado que puede ser esencial al momento de reevaluar o reconfigurar una actividad específica.

Consideraciones de uso

Responsable: Equipo de desarrollo.

Descarga: para acceder al documento y realizar esta tarea acceder al repositorio: https://github.com/cDsants/M_I_R_N_P/blob/entregables_M_I_R_N_P/Fase3-Control/BitacoraRetroalimentacion.docx

Diagrama de Tarea:

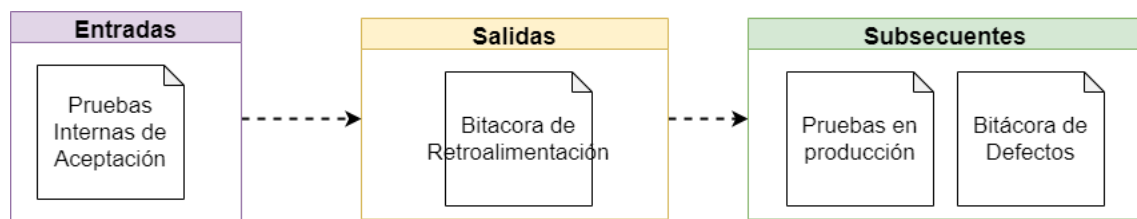


Figura 25: Bitácora de retroalimentación

Fuente: Elaborado por el Autor

3.2.8.2 PRUEBAS DE ACEPTACION INTERNAS

Propósito

Ejecutar las pruebas de calidad de software/sistema con las métricas de calidad establecidas en MIRNP

Descripción

Establecer cuáles serán las pruebas que se van a evaluar para el aseguramiento de calidad de del software para luego ser medidos con las métricas adaptadas y elaborados para MIRNP, literal 3.3 del presente texto.

Consideraciones de uso

Responsable: Director del desarrollo, Equipo de desarrollo.

Técnicas recomendadas a Utilizar: ISO 9126-4.

Descarga: para acceder al documento y realizar esta tarea acceder al repositorio: https://github.com/cDsants/M_I_R_N_P/blob/entregables_M_I_R_N_P/Fase3-Control/PRUEBAS_ACEPTACIÓN_INTERNAS.docx

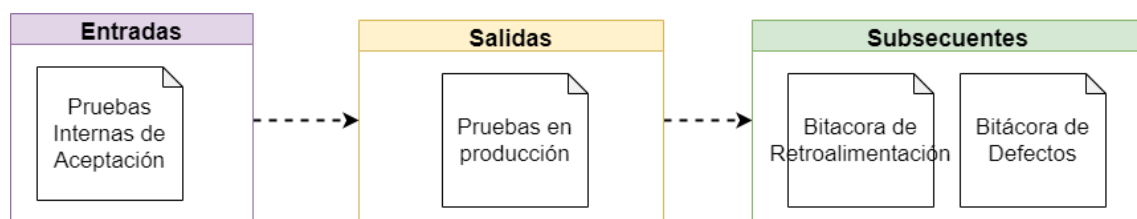


Figura 26: Pruebas en producción

Fuente: Elaborado por el Autor

3.2.8.3 BITÁCORA DE DEFECTOS.

Propósito

Generar un registro de los principales defectos a encontrarse durante el desarrollo de la ejecución del modelo propuesto o actualizado.

Descripción

Luego de aplicar varias pruebas del funcionamiento del sistema a proponerse, con el debido registro de actividades a la par se detallan los defectos encontrados durante y posterior a la evaluación de aceptación por los interesados. De tal manera que el software garantice el cumplimiento de los resultados obtenidos en la ejecución.

Consideraciones de uso

Responsable: Director del desarrollo, Equipo de desarrollo.

Técnicas recomendadas a Utilizar: ISO 9126-4.

Descarga: para acceder al documento y realizar esta tarea acceder al repositorio:

[https://github.com/cDsants/M_I_R_N_P/blob/entregables_M_I_R_N_P/Fase3-Control/BitacoraDefectos\(Bug\).docx](https://github.com/cDsants/M_I_R_N_P/blob/entregables_M_I_R_N_P/Fase3-Control/BitacoraDefectos(Bug).docx)

Diagrama de Tarea:

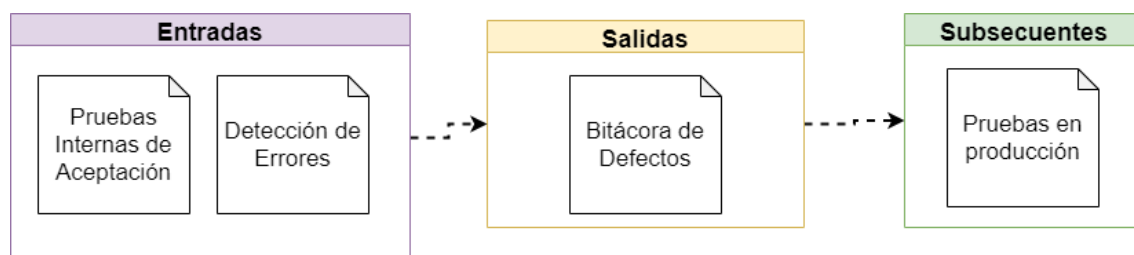


Figura 27: Bitácora de Defectos (Bug)

Fuente: Elaborado por el Autor

3.2.8.4 DETECCIÓN DE ERRORES EN EL CICLO DE VIDA SOFTWARE.

Propósito

Corregir los posibles errores existentes durante el diseño e implementación del software.

Descripción

Las constantes pruebas del sistema brindado tienen la intención de garantizar un buen funcionamiento del sistema propuesto, por tal razón, se toman los correctivos necesarios con la finalidad de mejorar el procesamiento de datos, presentación de interfaz o interactividad con los usuarios o clientes de la empresa contratante. Este proceso de pruebas se realiza y se documenta en el entregable de Bitácora de Defectos.

Consideraciones de uso

Responsable: Director del desarrollo, Equipo de desarrollo.

Técnicas recomendadas a Utilizar: ISO 9126-4.

Diagrama de Tarea:

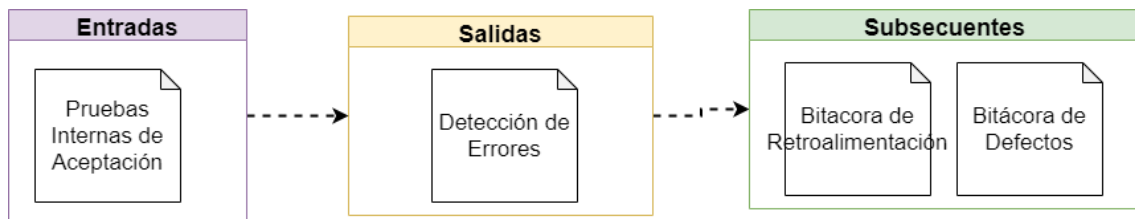


Figura 28: Proceso de detección de Errores

Fuente: Elaborado por el Autor

3.2.8.5 PROTOTIPO BETA

Propósito

Sustituir las fallas detectadas del software basadas en los defectos registrados de las bitácoras de control.

Descripción

Para el caso de una posible desestimación de la propuesta el prototipo Alpha se debe tener un plan de corrección de los errores consecuentes. En una nueva propuesta que garantiza el cumplimiento de los estándares de calidad y desarrollo en la funcionalidad del software.

Consideraciones de uso

Responsable: Director del desarrollo, Equipo de desarrollo.

Técnicas recomendadas a Utilizar: ISO 9126-4.

Diagrama de Tarea:

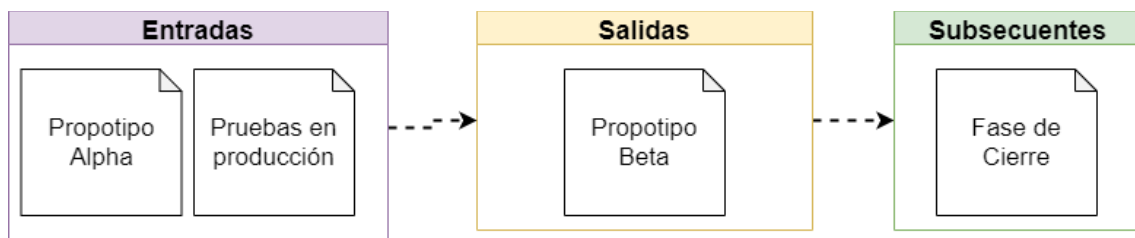


Figura 29: Proceso Prototipo Beta

Fuente: Elaborado por el Autor

3.2.9 FASE 4 – CIERRE O DESPLIEGUE

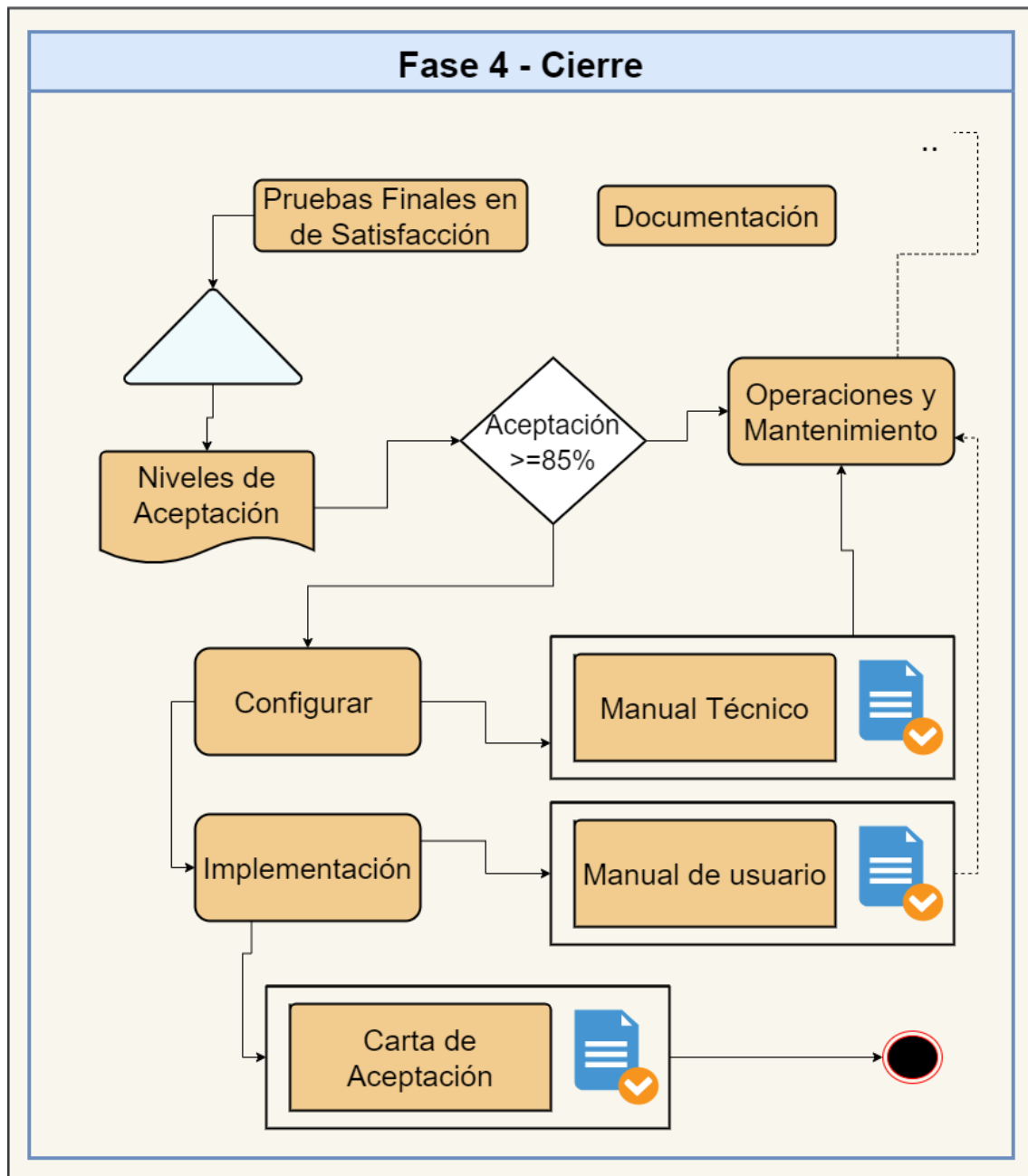


Figura 30: Fase 4 Cierre, MIRNP

Fuente: Elaborado por el Autor

3.2.9.1 PRUEBAS FINALES EN PRODUCCIÓN

Propósito

Evaluar y brindar el software solicitado por la empresa realizando pruebas de satisfacción.

Descripción

Se plantea el funcionamiento del sistema/software de usabilidad del cliente dependiendo la particularidad con la que se ha contratado a la empresa de diseño. Para estas pruebas crearon y adaptaron de la norma ISO 9126-4 métricas de calidad para el aseguramiento de la calidad.

Consideraciones de uso

Responsable: Director del desarrollo, Equipo de desarrollo.

Técnicas recomendadas a Utilizar: ISO 9126-4.

Diagrama de Tarea:

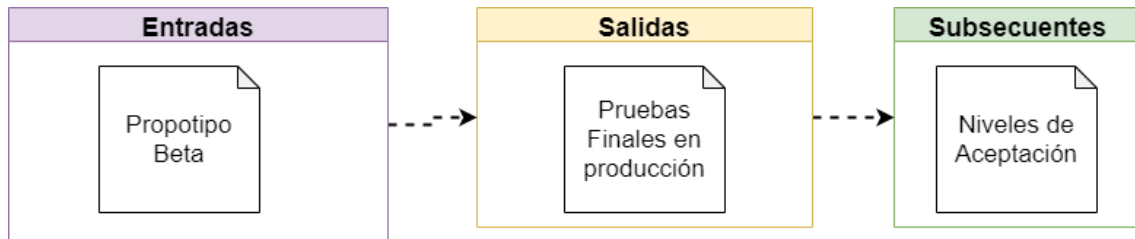


Figura 31: Pruebas Finales en producción

Fuente: Elaborado por el Autor

3.2.9.2 IMPLEMENTACIÓN.

Propósito

Implementar el software o sistema diseñado para la empresa solicitante, posterior a la evaluación de control y calidad de las normas ISO.

Descripción

Con la implementación del sistema contratado por el cliente se culmina la fase de cierre de la propuesta debido a la funcionalidad favorable que genera el prototipo beta. Dando como resultados un producto confiable y satisfactorio.

Consideraciones de uso

Responsable: Director del desarrollo, Equipo de desarrollo.

Técnicas recomendadas a Utilizar: ISO 9126-4.

Diagrama de Tarea:

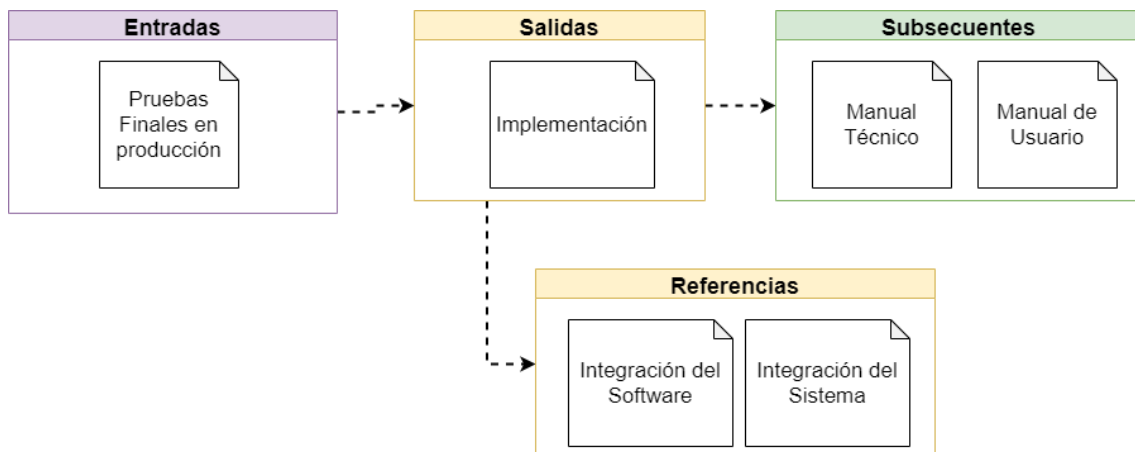


Figura 32: Proceso Implementación

Fuente: Elaborado por el Autor

3.2.9.3 MANUAL TÉCNICO.

Propósito

Guiar metodológicamente como funciona cada fase del sistema en el desarrollo de las actividades por las que implemento la empresa.

Descripción

El manual de ficha técnica consta con una información básica y específica para entender el funcionamiento del software, como también le provee al cliente las normas de acción en casos de presentar anomalías en el desenvolvimiento del sistema.

Consideraciones de uso

Responsable: Director del desarrollo, Equipo de desarrollo.

Técnicas recomendadas a Utilizar: ISO 9126-4.

Descarga: para acceder al documento y realizar esta tarea acceder al repositorio:

https://github.com/cDsants/M_I_R_N_P/blob/entregables_M_I_R_N_P/Fase4-Cierre/MANUAL_TECNICO.docx

Diagrama de Tarea:

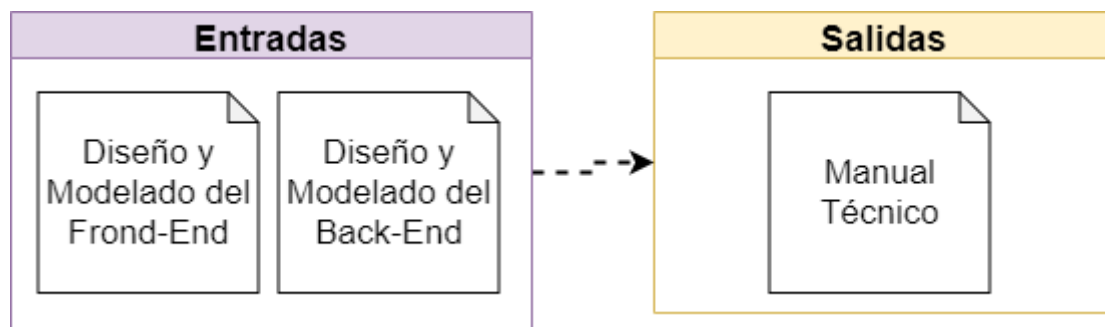


Figura 33: Manual Técnico
Fuente: Elaborado por el Autor

3.2.9.4 MANUAL DE USUARIO.

Propósito

Proveer de un manual de usuario con pasos instructivos del manejo y uso correcto del software de sistema requerido por el contratante.

Descripción

El manual del usuario cuenta con una guía ilustrativa que brinda una información detallada de cada componente del software.

Consideraciones de uso

Responsable: Director del desarrollo, Equipo de desarrollo.

Descarga: para acceder al documento y realizar esta tarea acceder al repositorio:
https://github.com/cDsants/M_I_R_N_P/blob/entregables_M_I_R_N_P/Fase4-Cierre/MANUAL_USUARIO.docx

Diagrama de Tarea:

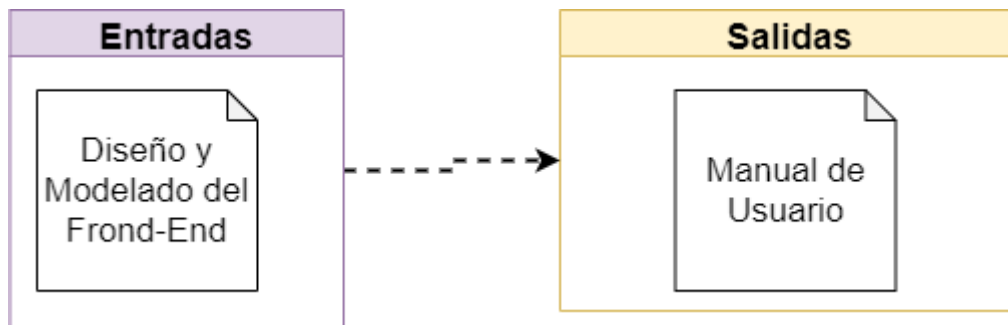


Figura 34: Manual de Usuario
Fuente: Elaborado por el Autor

3.2.9.5 CARTA DE ACEPTACIÓN DEL SOFTWARE.

Propósito

Establecer el acuerdo entre la empresa proveedora y el cliente que contrata el software calificado y adaptado a las normas y protocolos de calidad, como también de satisfacción del cliente.

Descripción

Una vez acordado las partes, se procede a formalizar el cierre contractual con especificaciones de cumplimiento bajo cualquier exigencia del cliente o la empresa proveedora del software. De tal manera que los resultados beneficien a ambas partes.

Consideraciones de uso

Responsable: Director del desarrollo, Equipo de desarrollo.

Descarga: para acceder al documento y realizar esta tarea acceder al repositorio:
https://github.com/cDsants/M_I_R_N_P/blob/entregables_M_I_R_N_P/Fase4-Cierre/CARTA_ACEPTACION.docx

Diagrama de Tarea:

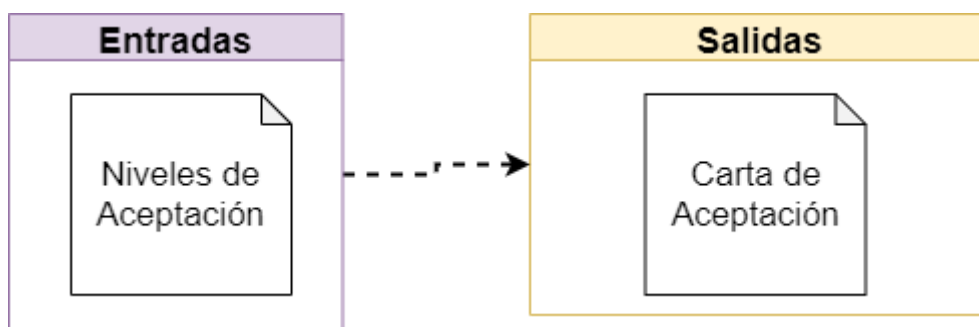


Figura 35: Carta de Aceptación

Fuente: Elaborado por el Autor

3.3 MÉTRICAS DE CALIDAD PARA EVALUACIÓN DE MIRNP

Desarrollar software que cumpla las necesidades y satisfaga los requisitos del problema es una tarea que a los equipos de desarrollo puede frustrar. Los sistemas, artefactos de software y aplicaciones pueden ser atractivas, seguras, de rendimiento alto. Sin embargo, en ocasiones no se realiza la evaluación de calidad por métricas. Es de valor invaluable que se realice o se cuantifique mediante métricas de calidad de software.

La MIRNP adaptó 07 métricas de calidad de software basándose en el estándar ISO/IEC 9126-4, para evaluar la satisfacción y calidad del software o sistema a implementar.[58]

3.3.1 Métrica de calidad MIRNP_Efe_01_Ent

La **MIRNP_01_Ent** es la primera métrica de calidad que se establece para la MIRNP, la función de esta es evaluar la cantidad de entregables desarrollados en completitud. En la tabla 12 se presenta en síntesis los parámetros para la evaluación.

Alias	MIRNP_Efe_01_Ent
Nombre de Métrica:	Entregable finalizado
Propósito de la Métrica:	Que los entregables de la MIRNP sean ejecutados.
Método de aplicación:	Prueba de usuario
Medición, fórmula y cálculo de elementos de datos	$X_e = A_e / B_e$ A_e = entregables ejecutados. B_e = total de entregables intentados.
Interpretación del valor medido:	$0 \leq X_e \leq 1$ más cerca de 1.0 resultado favorable.
Tipo de métrica:	X_e = count/count. A_e = count B_e = count.
Entrada a la medición:	Entregables de la MIRNP
Referencia ISO/IEC 12207 SLCP	6.6. Revisión conjunta
Público objetivo:	Equipo de desarrollo, Director de Desarrollo

Tabla 8: Métrica de calidad MIRNP_01_Ent

Fuente: Elaborada por el Autor.

3.3.2 Métrica de calidad MIRNP_Pro_02_Eficiencia

La métrica **MIRNP_Pro_02_Eficiencia** es de productividad y se encarga de evaluar la eficiencia que consume con relación al tiempo en lograr un proceso o tarea en el sistema/software. Esta métrica de calidad se la puede evaluar en la Fase 3 de la MIRNP sobre el entregable prototipo beta y cumplir con la entregable prueba de aceptación. En la tabla 13 se ilustra los parámetros a utilizar para realizar esta evaluación.

Alias	MIRNP_Pro_02_Eficiencia
-------	-------------------------

Nombre de Métrica:	Eficiencia de la tarea
Propósito de la Métrica:	¿Qué tan eficiente es para el usuario?
Método de aplicación:	Prueba de usuario
Medición, fórmula y cálculo de elementos de datos	ET = M1/T M1= Eficacia de la tarea. Tm=Tiempo de la tarea.
Interpretación del valor medido:	0 <= ET mayor el valor de ET mejor
Tipo de métrica:	Tm=time
Entrada a la medición:	Pruebas de aceptación internas
Referencia ISO/IEC 12207 SLCP	5.3. Pruebas de calificación
Público objetivo:	Equipo de desarrollo, Director de Desarrollo

Tabla 9: MIRNP_Pro_02_Eficiencia

Fuente: Elaborada por el Autor.

3.3.3 Métrica de calidad MIRNP_Seg_03_Averias

La métrica **MIRNP_Seg_03_Averias** es de seguridad en el ámbito de averías que pueda sufrir el producto software. En la MIRNP se debe evaluar esta métrica en la Fase 3 correspondientes a las pruebas en producción y detección de errores (Bug) con la finalidad de que las pruebas de aceptación internas sean las apropiadas y favorables. A continuación, se ilustra en la tabla 14 donde se establecen los parámetros necesarios para la evaluación.

Alias	MIRNP_Seg_03_Averias
Nombre de Métrica:	Averías en el software
Propósito de la Métrica:	Cuál es la incidencia de averías en las pruebas del producto software/sistema.
Método de aplicación:	Estadísticas de uso
Medición, fórmula y cálculo de elementos de datos	Xa = 1-Aa/Ba Aa= número de ocurrencias de corrupción en el software. Ba=número total de situaciones de uso.
Interpretación del valor medido:	0 <= Xa <= 1 más cerca de 1.0 resultado favorable.
Tipo de métrica:	Xa=count/count. Aa=count Ba=count.
Entrada a la medición:	Registro de supervisión de uso
Referencia ISO/IEC 12207 SLCP	5.4. Operación
Público objetivo:	Equipo de desarrollo, Director de Desarrollo, Interesados, Interface, Diseño

Tabla 10: Métrica de calidad MIRNP_Seg_03_Averias

Fuente: Elaborada por el Autor.

3.3.4 Métrica de calidad MIRNP_Satisfacion_04_SaCu

La métrica MIRNP_Satisfac_04_SaCu es de satisfacción y ayudan a evaluar que aptitudes tienen los usuarios con el uso del producto sistema/software en procesos, funciones o módulos específicos. También influye la percepción del usuario en eficiencia, productividad y seguridad en uso.

cabe indicar que esta prueba se la puede realizar a lo largo del ciclo de vida de MIRNP en la Fase 2, prototipo Alpha. Fase 3 en el prototipo beta y finalmente para medir el nivel de satisfacción del usuario en la Fase 4. En la tabla 15 se presenta en síntesis los parámetros para la evaluación.

Alias	MIRNP_Satisfacion_04_SaCu
Nombre de Métrica:	Satisfacción_Cuest_
Propósito de la Métrica:	Medir la satisfacción del usuario con las características solicitadas del software/sistema.
Método de aplicación:	Pruebas a usuarios
Medición, fórmula y cálculo de elementos de datos	$X_r = \sum(A_i)/nr$ A_i = respuesta a una pregunta. nr =número total de respuestas.
Interpretación del valor medido:	V de Aiken – más cerca de 1 mejor.
Tipo de métrica:	X_r =count. A_i =count
Entrada a la medición:	Registro de supervisión de uso - Registro de supervisión de usuarios
Referencia ISO/IEC 12207 SLCP	5.3 Pruebas de calificación 5.4. Operación 6.5. Validación
Público objetivo:	Equipo de desarrollo, Director de Desarrollo, Interesados, Interface, Diseño

Tabla 11: Métrica de calidad MIRNP_Satisfacion_04_SaCu

Fuente: Adaptada y Elaborada por el Autor.

3.3.5 Métrica de calidad MIRNP_Satisfac_05_Usabilidad

Esta métrica tiene como finalidad de utilizar el recurso de la observación para capturar la cantidad de funciones visuales que el usuario puede notar en el sistema/software en contraste a la cantidad de funciones evaluadas escogidas de los requisitos establecidos en los entregables StRS, SRS, SyRS. Esta métrica es de la dimensión de Usabilidad en sus grados de entendibilidad y facilidad de aprendizaje. Cabe indicar que esta prueba se la puede realizar a lo largo del ciclo de vida de MIRNP en la Fase 2, prototipo Alpha. Fase 3 en el prototipo beta y finalmente para medir el nivel de satisfacción del usuario en la Fase 4. A continuación en la tabla 16 se presentan los parámetros para realizar la evaluación de la métrica.

Alias	MIRNP_Satisfac_05_Usabilidad
Nombre de Métrica:	Usabilidad_funciones_detectadas
Propósito de la Métrica:	Medir la cantidad de funciones del sistema/software que el usuario detecta con facilidad.
Método de aplicación:	Pruebas a usuarios – Observación
Medición, fórmula y cálculo de elementos de datos	$Us = nF/tF$ nF = número de funciones evidentes por el usuario. tF = total de funciones evaluadas.
Interpretación del valor medido:	V de Aiken – más cerca de 1 mejor.
Tipo de métrica:	Us=count. nF=count tF=count.
Entrada a la medición:	StRS – SRS – prototipo Alpha – prototipo Beta -
Referencia ISO/IEC 12207 SLCP	5.3 Pruebas de calificación 5.4. Operación 6.6. Revisión conjunta 6.5. Validación
Público objetivo:	Equipo de desarrollo, Director de Desarrollo, Interesados, Interface, Diseño

Tabla 12: Métrica de calidad MIRNP_Satisfac_05_Usabilidad

Fuente: Adaptada y Elaborada por el Autor.

3.3.6 Métrica de calidad MIRNP_Efici_06_Rendimiento

Para la MIRNP se adapta esta métrica para calcular rendimiento del software/sistema como base para la obtención de los resultados se realizan simulaciones o calculo directo de ejecución del sistema, módulos o procesos a evaluar. Esta métrica debe o puede ser evaluada en la fase 3 de la metodología al realizar las pruebas de aceptación internas. A continuación, en la tabla 17 se presentan los parámetros para realizar la evaluación de la métrica

Alias	MIRNP_Efici_06_Rendimiento
Nombre de Métrica:	Respuestas en el tiempo
Propósito de la Métrica:	Medir la cantidad de tiempo que necesita el software/sistema para completar una tarea.
Método de aplicación:	Pruebas a usuarios - Estadísticas de uso
Medición, fórmula y cálculo de elementos de datos	tRS= tiempo (calculado y/o simulado)
Interpretación del valor medido:	Entre más cerca de 1 mejor.
Tipo de métrica:	tRS=time
Entrada a la medición:	Simulación o calculo tiempo de espera llamada al sistema/software.
Referencia ISO/IEC 12207 SLCP	5.4. Verificación 6.6. Revisión conjunta

	6.5. Validación
Público objetivo:	Equipo de desarrollo, Director de Desarrollo, Interesados, Interface, Diseño

Tabla 13: Métrica de calidad MIRNP_Efici_06_Rendimiento

Fuente: Adaptada y Elaborada por el Autor.

3.3.7 Métrica de calidad MIRNP_Mante_07_Cambiabilidad

Para la MIRNP se adapta y establece esta métrica para constatar la cantidad de cambios, actualizaciones en la especificaciones y módulos correctamente versionados y comentados. Es una métrica de la dimensión Mantenibilidad. En la MIRNP se recomienda evaluar en la Fase 3 al realizar las pruebas de aceptación internas y el proceso de retroalimentación. En la tabla 18 que se presenta a continuación se presenta los parámetros para realizar las pruebas de validación.

Alias	MIRNP_Mante_07_Cambiabilidad
Nombre de Métrica:	Registro_Cambios
Propósito de la Métrica:	Registrar las modificaciones, cambios, actualizaciones en la especificación y o módulos con comentarios y versionado en el código.
Método de aplicación:	Registro de la información sobre los incidentes
Medición, fórmula y cálculo de elementos de datos	$mC = nCf / tFm$ nCf= número de cambios registrados tFm= total de módulos o funciones modificados
Interpretación del valor medido:	$0 \leq X \leq 1$ Entre más cerca de 1, mayor cantidad de cambios. 0 indica control de cambios deficientes o pocos cambios y estabilidad alta.
Tipo de métrica:	mC=count nCf=count tFm=count
Entrada a la medición:	Bitácora de retroalimentación – Especificación de requisitos.
Referencia ISO/IEC 12207 SLCP	5.4. Verificación 6.6. Revisión conjunta
Público objetivo:	Equipo de desarrollo, Director de Desarrollo, Interesados, Interface, Diseño

Tabla 14: Métrica de calidad MIRNP_Mante_07_Cambiabilidad

Fuente: Adaptada y Elaborada por el Autor.