  
  
  
  
FACULTAD DE CIENCIAS EMPRESARIALES  
DEPARTAMENTO DE CIENCIAS DE LA COMPUTACIÓN Y TECNOLOGIAS   
DE LA INFORMACIÓN  
  
  
APLICACIÓN MÓVIL DE BÚSQUEDA DE RECORRIDOS DE BUSES WIM   
(“WHERE IS MY”)  
  
  
  
MEMORIA PARA OBTENER EL TITULO DE  
INGENIERIA CIVIL EN INFORMATICA  
  
  
AUTORES  
MATÍAS ANDRÉS ROBLES DÍAZ  
TAMARA VALENTINA SALGADO VILLALOBOS  
  
  
PROFESOR GUÍA  
ALFONSO RODRÍGUEZ RÍOS  
  
  
PROFESOR INFORMANTE

25 DE DICIEMBRE DE 2020

CHILLÁN - CHILE

Resumen

Este proyecto se presenta para dar conformidad a los requisitos exigidos por la Universidad del Bío- Bío en el proceso de titulación para la carrera de Ingeniería Civil en Informática. El proyecto se titula “Aplicación móvil de búsqueda de recorridos de buses WIM (*“Where is My”*)”.

Este proyecto posee como finalidad el crear un canal de comunicación directo entre las empresas de transporte público y los pobladores de las zonas rurales. Debido a que la información existente acerca de los recorridos de buses que se movilizan entre las diferentes ciudades y localidades no posee un lugar centralizado en donde sea posible la consulta de estos recorridos, además de no existir un lugar en donde las empresas puedan ofrecer sus servicios o dar información referente a estos, generando pérdidas a las empresas al no poder darse a conocer de manera efectiva y generando malestar o incomodidad en la población usuaria de estos medios de transporte al carecer de una información detalla y oportuna de los recorridos disponibles.

Es por esto, que el objetivo de este proyecto es la creación de un canal directo y centralizado a través de una aplicación móvil, en donde los usuarios de los medios de transporte rurales puedan consultar por los recorridos de su interés y a su vez las empresas puedan darse a conocer.

Finalmente, para llevar a cabo el desarrollo de este proyecto se utiliza la metodología ágil e incremental y también el uso de nuevas tecnologías como en el caso de NodeJs para desarrollar el Back-End y el Framework React Native para desarrollar el Front-End.

Abstract

Ídem al resumen, en Inglés.

Índice General

[1 Introducción 8](#_Toc295399549)

[2 Definicion de la empresa o institución 8](#_Toc295399550)

[2.1 Descripción de la empresa 8](#_Toc295399551)

[2.2 Descripción del área de estudio 8](#_Toc295399552)

[2.3 Descripción de la problemática 8](#_Toc295399553)

[3 Definición proyecto 8](#_Toc295399554)

[3.1 Objetivos del proyecto 8](#_Toc295399555)

[3.2 Ambiente de Ingeniería de Software 9](#_Toc295399556)

[3.3 Definiciones, Siglas y Abreviaciones 9](#_Toc295399557)

[4 Especificación de requerimientos de Software 9](#_Toc295399558)

[4.1 Alcances 9](#_Toc295399559)

[4.2 Objetivo del software 9](#_Toc295399560)

[4.3 Descripción Global del Producto 9](#_Toc295399561)

[4.3.1 Interfaz de usuario 9](#_Toc295399562)

[4.3.2 Interfaz De Hardware 10](#_Toc295399563)

[4.3.3 Interfaz Software 10](#_Toc295399564)

[4.4 Requerimientos Específicos 10](#_Toc295399565)

[4.4.1 Requerimientos Funcionales del sistema 10](#_Toc295399566)

[4.4.2 Interfaces externas de entrada 10](#_Toc295399567)

[4.4.3 Interfaces externas de Salida 11](#_Toc295399568)

[4.4.4 Atributos del producto 11](#_Toc295399569)

[5 Factibilidad 11](#_Toc295399570)

[5.1 Factibilidad técnica. 11](#_Toc295399571)

[5.2 Factibilidad operativa. 11](#_Toc295399572)

[5.3 Factibilidad económica. 12](#_Toc295399573)

[5.4 Conclusión de la factibilidad 12](#_Toc295399574)

[6 Análisis 12](#_Toc295399575)

[6.1 Diagrama de Flujo de Datos 12](#_Toc295399576)

[6.2 Casos de uso 12](#_Toc295399577)

[6.2.1 Actores 12](#_Toc295399578)

[6.2.2 Diagrama de Casos de Uso y descripción 13](#_Toc295399579)

[6.2.3 Especificación de los Caso de Uso 13](#_Toc295399580)

[6.3 Modelamiento de datos 13](#_Toc295399581)

[7 Diseño 13](#_Toc295399582)

[7.1 Diseño de Físico de la Base de datos 13](#_Toc295399583)

[7.2 Diseño de arquitectura funcional 14](#_Toc295399584)

[7.3 Diseño interfaz y navegación 14](#_Toc295399585)

[7.4 Especificación de módulos 15](#_Toc295399586)

[8 Pruebas 15](#_Toc295399587)

[8.1 Elementos de prueba 15](#_Toc295399588)

[8.2 Especificación de las pruebas 15](#_Toc295399589)

[8.3 Responsables de las pruebas 16](#_Toc295399590)

[8.4 Calendario de pruebas 16](#_Toc295399591)

[8.5 Conclusiones de Prueba 16](#_Toc295399592)

[9 Plan de Capacitación y entrenamiento 16](#_Toc295399593)

[10 Plan de Implantación y Puesta en marcha 16](#_Toc295399594)

[11 Resumen Esfuerzo requerido 17](#_Toc295399595)

[12 CONCLUSIONES 17](#_Toc295399596)

[13 Bibliografía 17](#_Toc295399597)

[14 ANEXO: PLANIFICACION INICIAL DEL proyecto 17](#_Toc295399598)

[14.1.1 Estimación inicial de tamaño 17](#_Toc295399599)

[14.1.2 Contabilización final del tamaño del Sw 17](#_Toc295399600)

[15 ANEXO: RESULTADOS DE ITERACIONES EN EL DESARROLLO 18](#_Toc295399601)

[16 ANEXO: MANUAL DE USUARIO 18](#_Toc295399602)

[17 ANEXO: ESPECIFICACION DE LAS PRUEBAS 18](#_Toc295399603)

[17.1 Pruebas de Unidad 18](#_Toc295399604)

[17.1.1 <nombre unidad> 18](#_Toc295399605)

[17.2 Sistema 18](#_Toc295399606)

[17.3 Aceptación 19](#_Toc295399607)

[18 ANEXO: DICCIONARIO DE DATOS DEL MODELO DE DATOS 19](#_Toc295399608)

[19 EJEMPLOS (QUITAR ESTE APARTADO) 19](#_Toc295399609)

[19.1 ISO/IEC 9126: Tecnología de Información – Evaluación del producto de software 19](#_Toc295399610)

[19.2 Esquema especificación de Interfaz 20](#_Toc295399611)

[19.3 Diagrama para representar la jerarquía de menú 22](#_Toc295399612)

[19.4 Árbol de descomposición funcional 22](#_Toc295399613)

[19.5 Estimación de tamaño de Sw: Punto Función 23](#_Toc295399614)

[19.6 Estimación de tamaño de Sw: Puntos de Casos de Uso 27](#_Toc295399615)

[19.7 Aspectos de Seguridad Informática a considerar en proyectos de Sw 28](#_Toc295399616)

Índice Tablas

Índice Figuras

# Introducción

Se presenta al lector cual es el propósito de este documento y se detalla el contenido de cada uno de sus capítulos.

# Definicion deL PUBLICO OBJETIVO

## Descripción del público objetivo

## Esta aplicación está principalmente dirigida a las personas pertenecientes a sectores rurales de la región de Ñuble que deben utilizar transporte público, así como también puede ser utilizada por personas de zonas urbanas que necesiten viajar a sectores rurales.

## Descripción de la problemática

## La región de Ñuble está constituida en su mayoría por sectores rurales. Estas zonas rurales poseen pequeños negocios, postas rurales, cajas vecinas, pero no poseen los grandes proveedores de servicios necesarios para mantener un estilo de vida promedio como el que se vive en pleno siglo XXI en nuestro país. Esta situación los obliga a viajar a las zonas urbanas en busca de estos servicios, así también, las personas que viven en las zonas urbanas y que necesitan mantener un contacto constante o parcial con algún residente de una zona rural, se ven en la obligación de utilizar algún medio de transporte que se dirija hacia estas zonas. Los medios de transporte presentes en nuestra región, rara vez son representados por una gran empresa o poseen una organización clara, muchas de estas son creadas, organizadas y dirigidas por los mismos vecinos residentes en estas zonas, quienes mayoritariamente carecen de una preparación como tal. El hecho de que muchas de las empresas de transporte sean reguladas por los mismos vecinos, genera problemas a la hora de hacer llegar la información referente al servicio, los precios, rutas de los buses o colectivos, vehículos utilizados y choferes presentes, que usualmente cambian sin que la población usuaria de este servicio se entere de manera oportuna, esto debido a la inexistencia de un canal de comunicación directa entre las empresas y los usuarios, generando dificultades y dudas, además de malestar en algunos casos. Esta situación hace parecer que el servicio no es óptimo y, en consecuencia, las empresas y los usuarios poseen pérdidas tanto en lo monetario como en el coste de tiempo. La inexistencia de un canal directo entre los usuarios y las empresas, dificulta la entrada de nuevos medios transporte y los usuarios al carecer de información actualizada y oportuna, no poseen en muchos casos más opciones que las recomendadas por sus propios vecinos y/o los mismos choferes de las líneas usuales, perjudicándolos de gran manera, por ejemplo, cuando algún residente de una zona rural no alcanza a subir en un bus o colectivo, este se ve obligado a esperar el próximo conocido, que usualmente comenzará su recorrido en varios minutos e incluso horas. Este tiempo de espera podría ser aprovechado al utilizar otra opción, si es que el usuario tuviera un lugar o documento que le mostrara más opciones de las que conoce y que estuvieran vigentes en el momento. Es importante señalar que, dentro del contexto previamente mencionado, es decir, en los sectores rurales presentes en nuestra región del Ñuble, se evidencia la precariedad de la señal telefónica y por ende el tráfico de datos móviles. Esta situación supone un reto a resolver por parte del proyecto y genera la necesidad de estudiar las tecnologías o estrategias utilizadas actualmente en la industria bajo problemáticas similares, con el fin de adoptarlas y/o modificarlas para lograr diseñar y desarrollar una solución factible dada estas condiciones.

Debe especificar claramente cuál es la situación actual de la actividad o proceso de negocio en estudio (cómo se efectúa en la actualidad). De la descripción de esta situación actual se identifica y especifica el problema que ha motivado la necesidad del sistema, lo cual definirá el objetivo del sistema. Si lo anterior no es coherente significa, simplemente, que el proyecto ha sido mal conceptualizado.

Opcionalmente, se pueden incluir los resultados de la aplicación de técnicas de análisis o estándares para el modelamiento de procesos de negocios tales como la notación BPMN a través de los diagramas de procesos de negocios, o UML 2.0 a través de los diagramas de actividad.

# Definición proyecto

## Objetivos del proyecto

### Objetivo General

Entregar un canal de información entre medios de transporte y población de la región de Ñuble, de una manera amigable, interactiva, oportuna y en tiempo real.

### Objetivos Específicos

* Investigar tecnologías que permitan el desarrollo multiplataforma con GPS.
* Desarrollar la aplicación.
* Contactar empresas que utilicen el software.
* Capacitar a las empresas en el uso de la app.
* Difundir la aplicación en los diferentes terminales.

Objetivos generales y específicos del proyecto, estos objetivos son distintos a los objetivos del software/sistema de Sw.

Los Objetivos del proyecto terminan con el proyecto y los objetivos del software se logran con el uso del software, es decir van más allá de la fecha de término del proyecto. Por ejemplo, un objetivo del proyecto puede comenzar como “*diseñar e implementar una solución a…”*

## Ambiente de Ingeniería de Software

Esta sección detalla la metodología a utilizar en el proceso de desarrollo de este proyecto, junto a la arquitectura de software, las técnicas y herramientas para el desarrollo.

### Metodología de desarrollo

Para este proyecto será utilizada una metodología de desarrollo ágil, iterativa e incremental, esto con la intención de ir probando las diferentes funcionalidades con usuarios de manera temprana y de esta forma generar correcciones oportunas, que permitan al momento del despliegue de la aplicación, reflejar el feedback otorgado por los usuarios.

### Herramientas

Para el desarrollo de esta aplicación, serán utilizados las siguientes tecnologías y lenguajes de programación:

## Base de Datos

## Firebase: Base de datos no relacional organizada en forma de árbol JSON. Esta base de datos creada por Google permite subscribirse en tiempo real, devolviendo información de la BDD si esta posee cambios.

## Entorno de desarrollo

## Git: Software que permite llevar un control de versiones. Utilizado para llevar un registro de los cambios en archivos presentes en un equipo, con el objetivo de coordinar los esfuerzos que varias personas realizan sobre estos archivos.

## Visual Studio Code: Editor de código multilenguaje desarrollado por Microsoft. Este editor de código posee integración con Git lo que permite mantener un control más rápido e interactivo de las versiones del producto.

## Framework

## React Native: Bajo el slogan de React Native “learn once, write everywhere” este framework permite la creación de aplicaciones móviles para la mayoría de las plataformas con un único código. Basado en JavaScript con un enfoque orientado a componentes, este permite una programación rápida y simple de aplicaciones móviles utilizando componentes nativos ya sean estos de iOS o Android.

## Lenguajes

## CSS: Cascading Style Sheets es un lenguaje de estilos utilizado para describir la presentación de documentos HTML o XML. Este se encarga de describir cómo debe ser renderizado un elemento.

## JavaScript: Lenguaje de programación ligero e interpretado open source. Utilizado en desarrollo web para interfaces de usuario e igualmente para desarrollo en ambientes externos al navegador como lo es Node.js.

### Patrones de diseño

Justificación breve de:

* metodología de desarrollo,
* las técnicas y notaciones,
* estándares de documentación, producto o proceso
* herramientas de apoyo al desarrollo de software que serán utilizadas.

Se pide indicar el propósito de cada elemento dentro del desarrollo del Sw. (para qué será utilizado). No se requiere que explique la historia o un manual de cada elemento.

## Definiciones, Siglas y Abreviaciones

Este ítem se incluye la definición de las siglas, abreviaciones, conceptos técnicos o de negocio que son necesarios para el buen entendimiento de este documento.

# Especificación de requerimientos de Software

Adaptación basada en *IEEE Software requirements Specifications* *Std 830-1998.*

## Alcances

Este ítem del estándar considera la descripción de las características de este producto de software que lo diferencian de otros. Se debe explicar en términos de lo que hará el producto y si es necesario que no hará.

## Objetivo del software

### Objetivo general del software

Brindar un canal directo de comunicación entre empresas de transporte rural y la población usuaria de los medios de transporte rural.

### Objetivos específicos del software

* Manejar dos perfiles. Con el fin de facilitar el uso en la plataforma. Los perfiles son: Empresa y Pasajero
* Registrar usuarios nuevos en la aplicación sean pasajeros o empresas.
* Almacenar los datos básicos de los recorridos como, por ejemplo; origen, destino, horarios, empresa, precio, entre otros.
* Listar los recorridos disponibles, dados un punto de origen, un punto de destino, una fecha y hora.
* Permitir el seguimiento a un recorrido para obtener noticias referentes a este en cualquier momento.
* Disponer la opción de publicar noticias para las empresas que así lo deseen y/o necesiten.

Se describen los objetivos que debe cumplir el software en forma general y específica. Debería señalarse en el objetivo global y correspondientes específicos los siguientes elementos o aspectos:

***INFORMACION*** *que considera /almacena / gestiona /maneja /etc-el* ***PROCESO*** *que apoya/realiza- y el* ***RESULTADO*** *que se logra.*

Ejemplo: *El sistema* ***manejará****información sobre el* ***proceso productivo*** *que permita una* ***planificación integral del mismo y logra un uso optimo de los recursos utilizados en el proceso.***

## Descripción Global del Producto

### Interfaz de usuario

NO TE ENTENDI NI V…

Se indican las características lógicas de cada interfaz entre el software y los usuarios. No se refiere al diseño de la interfaz sino a los requerimientos que existen en la empresa respecto a la interfaces de sw.

Por ejemplo puede incluir características de configuración como formato de pantalla, o layout de informes, uso de colores, iconografía, usos de teclas programables, etc.

Se indican también todos los aspectos de optimización, que indique las forma como el software debe y no debe aparecer al usuario.

### Interfaz De Hardware

Especifica las características lógicas de cada interfaz entre el software y los periféricos hardware, por ejemplo:

* Sensor XWS, N° Interrupción, puerto

Estas interfaces se especifican especialmente cuando el sw interactúa con hw. específico que requiere, para manejarlos correctamente, configurar conexiones, puertos o interrupciones y evitar colisiones con otros dispositivos.

### Interfaz Software

Se especifica el uso de otros productos software (por ejemplo un sistema de administración de datos, sistema operativo o paquetes matemáticos), e interfaces con otros sistemas (p.e., en las relaciones entre un sistema de cuentas por cobrar y un sistema de contabilidad general). Para cada producto software con el cual existe alguna relación o interfaz, se detalla:

* Nombre;
* Abreviación;
* Número especificación o Versión;
* Fuente.

### Interfaces de comunicación

Se indican las interfaces para la comunicación tales como el/los protocolos de comunicación.

## Requerimientos Específicos

### Requerimientos Funcionales del sistema

La lista de los requerimientos funcionales específicos. Se recomienda el uso de la forma verbal en infinitivo para denotar las acciones que el software debe realizar. Los requerimientos deben ser enumerados para facilitar su seguimiento.

En la descripción de cada requerimiento se incluyen condiciones o restricciones del requerimiento, por ejemplo “*los registros de los clientes pueden ser eliminados si y sólo si el cliente no ha efectuado ninguna compra en los 5 últimos años*”.

Los requerimientos pueden ser agrupados por distintos criterios, por ejemplo tipo de usuario o módulo (otras organizaciones se encuentran en el anexo del estándar *IEEE Std 830-1998)*.

|  |  |  |
| --- | --- | --- |
| Id | Nombre | Descripción |
|  |  |  |
|  |  |  |
|  |  |  |
|  |  |  |

### Interfaces externas de entrada

Cada interfaz de entrada indica todos los grupos de datos que serán ingresados al sistema independiente del medio de ingreso.

|  |  |  |
| --- | --- | --- |
| Identificador | Nombre del ítem. | Detalle de Datos contenidos en ítem |
| DE\_01 | Datos del proveedor | NOMBRE, RUT, GIRO, DIRECCION,TELEFONO |

### Interfaces externas de Salida

Se especifica cada salida del sistema, indicando en cada caso el formato o medio de salida.

|  |  |  |  |
| --- | --- | --- | --- |
| Identificador | Nombre del ítem. | Detalle de Datos contenidos en ítem | Medio Salida |
| IS\_01 | Informe de los proveedores | NOMBRE, RUT, CODIGO,GIRO,DIRECCION,TELEFONO | Archivo XLS  Impresora  Pantalla |

### Atributos del producto

Tomando como referencia el modelo ISO/IEC 9126 (ver sección 19.1, página 19), se indican los atributos de calidad del producto software que se desarrolla. Considere que no es necesario que todos los atributos sean aplicables al producto, solo indique los que se consideran críticos. Defina en términos claros como se concretan cada uno de los atributos de calidad de forma que sean fáciles de comprobar.

Por ejemplo:

* USABILIDAD- OPERABILIDAD. Que los mensajes de error sean claros para el usuario. Es decir que en el 100% de los mensajes de error, relacionados con el ingreso y procesamiento de datos, se indique claramente el error, la causa y la solución.
* EFICIENCIA- TIEMPO DE EJECUCIÓN/RESPUESTA. El sistema debe garantizar un tiempo de carga inferior a X segundos considerando una conexión de red de velocidad estándar de XYZ Mbyte/s y 10 usuarios conectados (Se pueden indicar consideraciones respecto al número de transacciones y tareas, usuarios o cantidad de datos soportados en un periodo de tiempo, en condiciones normales o en un peak).
* FUNCIONALIDAD-SEGURIDAD. El sistema debe mantener un control de acceso a la funcionalidad a través de login –password establecido según los perfiles definidos para los usuarios del sistema. El sistema debe exigir que los datos de login-password sean renovados cada 15 días.

# Factibilidad

## Factibilidad técnica.

Los recursos necesarios para el desarrollo de esta aplicación son mostrados a continuación:

### Software necesario

En tabla a continuación se muestran las herramientas en el ámbito de Software necesarias para el desarrollo de la aplicación:

|  |  |
| --- | --- |
| **Software o Herramienta** | Costo |
| Firebase (base de datos) | $0 |
| React native | $0 |
| Visual Studio Code | $0 |
| Bizagi Modeler | $0 |
| yEd | $0 |
| Libre Office | $0 |
| GitHub | $0 |
| NodeJs | $0 |
| Google Chrome | $0 |
| Docker | $0 |
| Total | **$0** |

Como se puede apreciar el costo del Software necesario para el desarrollo de la aplicación es nulo, lo que supone un punto a favor para el proyecto, debido a que estas tecnologías son públicas y gratuitas.

### Hardware necesario

A continuación, se muestran las herramientas en el ámbito de Hardware necesarias para el desarrollo de la aplicación.

* **Equipos para desarrollo:**

En las siguientes tablas se muestran los requisitos mínimos y los requisitos óptimos necesarios para dos equipos que serán utilizados en el desarrollo de la aplicación:

|  |  |
| --- | --- |
| **Requisitos mínimos de equipos para desarrollo** | |
| Procesador | Intel Core i5 7300HQ |
| RAM | 8GB |
| Almacenamiento | 256GB SSD |
| Sistema Operativo | Windows 10 |

|  |  |
| --- | --- |
| **Requisitos óptimos de equipos para desarrollo** | |
| Procesador | Intel Core i9 |
| RAM | 16GB |
| Almacenamiento | 1TB SSD |
| Sistema Operativo | macOS Catalina |

* **Servidores:**

En la siguiente tabla se muestran las características mínimas necesarias para los servidores:

|  |  |
| --- | --- |
| **Requisitos óptimos de equipos para desarrollo** | |
| Procesador | Intel Core i3 |
| RAM | 4GB |
| Almacenamiento | 500GB SSD |
| Transferencia | 20TB |

### Cuentas de desarrollador

En la siguiente tabla se muestran las cuentas de desarrollador necesarias para la publicación de la aplicación en las tiendas de aplicaciones más populares:

|  |  |
| --- | --- |
| **Cuentas de Desarrollador** | |
| Google Developer Account | $19.250 pago único |
| Apple Developer Account | $77.000 pago único |

### Requisitos mínimos para los dispositivos de los usuarios finales

En la siguiente tabla se muestran los requisitos mínimos necesarios para la utilización de la aplicación:

|  |  |
| --- | --- |
| **Requisitos mínimos de equipos usuarios finales** | |
| RAM | 1GB |
| Almacenamiento | 1GB de memoria libre |
| Sistema Operativo | Android 4.1 |

|  |  |
| --- | --- |
| **Requisitos mínimos de equipos usuarios finales** | |
| RAM | 1GB |
| Almacenamiento | 1GB de memoria libre |
| Sistema Operativo | iOS 10.0 |

Identificar y especificar dispositivos, equipamiento y software (versión y tipo de licencia) necesarios para el desarrollo y operación del sistema de información o aplicación de software.

Además se especifica si existen las capacidades técnicas del equipo de trabajo para abordar un proyecto como el que se propone. Estas capacidades incluyen conocimientos y habilidades en los métodos, técnicas y herramientas utilizadas en el desarrollo, así como en la tecnología inherente al software a desarrollar, por ejemplo tecnología móvil, biométrica, etc. De lo anterior se puede derivar la contratación de capacitación o directamente externalizar el desarrollo. En ambos casos existe incidencia en la factibilidad económica.

## Factibilidad operativa.

Esta aplicación representa una solución muy atractiva y positiva para la comunidad rural debido a los vacíos presentes en la información de los medios de transporte, además de mejorar los tiempos de respuesta por parte de la comunidad frente a eventuales problemas en las líneas de transporte.

La principal dificultad operacional podría ser vista en la población más longeva debido a la poca afinidad con las plataformas digitales, junto a esto supone una barrera operativa más, la capacidad de las redes presentes en las zonas rurales, en donde es posible apreciar en algunas zonas una conexión 4G constante y unos metros más adelante una inexistencia de la red. Contrastando con lo anterior, las tecnologías utilizadas vendrían a solucionar esta problemática, ya que la capacidad de adaptación y reacción que otorga “React Native” permitiría a la aplicación contar con un bajo consumo de red y en el caso de perderse la conexión, al ser esta reestablecida, retomar el estado de la aplicación, actualizando solamente la información que haya cambiado dentro del tiempo en que la conexión no se encontraba establecida sin perder las interacciones del usuario. Otro atributo de la tecnología y que beneficia a la factibilidad operacional son los pocos recursos que esta necesita para su funcionamiento, debido a que combina la potencia otorgada por JavaScript con la fluidez de los lenguajes nativos.

Finalmente, el estudio de factibilidad operacional determina que el proyecto posee un factor de bajo riesgo operacional, haciendo de este un proyecto operacionalmente factible.

Establecer los impactos (positivos y/o negativos) que la implementación del sistema de información o aplicación de software implicará en aspectos relacionados con la institucionalidad, los procesos, los actores, los recursos o cualquier aspecto relacionado con la operación de la organización.

## Factibilidad económica.

### Costo de desarrollo

* **Arquitecto de software:** Para el diseño se requerirá de un arquitecto, en promedio un arquitecto recibe un sueldo de $1.251.000, el costo por hora es de $6.950, esto calculado por 180 hrs. Mensuales.
* **Analista Programador:** Para la etapa de análisis del proyecto se necesita de un analista, este cargo tiene un sueldo promedio de $871.200 por 180 hrs. de trabajo, por lo que el costo por hora es de $4.840.
* **Programador:** El programador se hará cargo del desarrollo e implementación de la aplicación el costo de cada hora de trabajo es de $5.200 calculado por 180hrs mensuales con un sueldo promedio de $936.000.
* **Quality Assurance:** Para asegurar la calidad del producto y la realización de pruebas se requiere de un Quality Assurance, el cual recibe un sueldo de $639.000 en promedio por 180 hrs. Trabajadas mensualmente. El costo por hora de este trabajador sería de $3.550.

|  |  |  |  |
| --- | --- | --- | --- |
| **Actividad** | **Tiempo (horas)** | **Costo x Hora** | **Total** |
| Análisis | 40 hrs | $4.840 | $193.600 |
| Diseño | 90 hrs | $6.950 | $625.500 |
| Desarrollo | 85 hrs | $5.200 | $442.000 |
| Pruebas | 30 hrs | $3.550 | $106.500 |
| Despliegue | 20 hrs | $5.200 | $104.000 |
| **Total** | **265 hrs** | **$25.740** | **$1.471.600** |

### Costo de operación

* Técnico en computación e informática: Para mantener tanto la aplicación web como la aplicación móvil, se requiere de un técnico, el sueldo promedio es de $475.000 por lo que su costo por hora es de $2.638, se requiere del técnico por 2 hora semanal, esto es $21.104 mensuales y $126.624 anuales.
* Dominio para la web: Se debe considerar el costo de un dominio web para la aplicación, este tiene un valor aproximado de $12.529.
* Servidor: Un plan básico de hosting varía entre $40.000 y $61.000 anuales, por lo que en promedio el servidor tiene un valor de $50.500 anuales.

|  |  |
| --- | --- |
| **Costos de Operación** | |
| Mantenimiento | $253.248 |
| Dominio Web | $12.529 |
| Servidor | $50.500 |
| **Total** | **$316.277** |

### Beneficios tangibles

Los beneficios tangibles percibidos en este proyecto corresponden a las mensualidades que deben ser canceladas por parte de las empresas que deseen mostrar los datos de sus medios de transporte. Estos cobros hacia las empresas son realizados de manera mensual y los valores se detallan en la siguiente tabla:

|  |  |
| --- | --- |
| **Tarifas por medio de transporte** | |
| Coste Servicio Buses | $3.000 / mensual por bus |
| Coste Servicio Colectivos | $1.500 / mensual por colectivo |

Para el cálculo de los ingresos se consideraron diferentes factores y con ello se realizaron las siguientes estimaciones:

* Para el primer año se estima poseer 12 buses y 4 colectivos asociados a la aplicación. Estos números se deben a que se consideraron 4 empresas que decidieran probar con 1 bus cada una. Esto para 2 terminales rurales presentes en Chillán y 1 presente en San Carlos. Considerando los montos mencionados en la tabla anterior, obtenemos $432.000 por un año en el que los buses se mantengan participando en la aplicación y si añadimos a los colectivos, que suman en su totalidad $72.000, obtenemos un total para el primer año de $504.000.
* Para el segundo año se estima un incremento en la cantidad de asociados correspondiente a 16 buses y 12 colectivos lo que supone un total de $792.000.
* En el tercer año se considera una expansión del territorio cubierto por la aplicación por lo que los asociados aumentan a un total de 22 buses y 18 colectivos, percibiendo como ingresos $1.116.000.
* En el 4to periodo de funcionamiento se espera contar con 26 buses y 22 colectivos, esto considerando nuevos participantes y pérdidas de asociados. En su totalidad esto generaría $1.350.000.
* En el último año de proyección se estima un alcance de 30 buses y 26 colectivos dando un total de $1.584.000.

A continuación, se muestra una tabla resumen con los beneficios tangibles anuales percibidos por medio de transporte.

|  |  |
| --- | --- |
| **Beneficios** | |
| Coste Servicio Buses | $36.000 / anual por bus |
| Coste Servicio Colectivos | $18.000 / anual por colectivo |
| **Total** | **$54.000 anuales** |

### Beneficios Intangibles

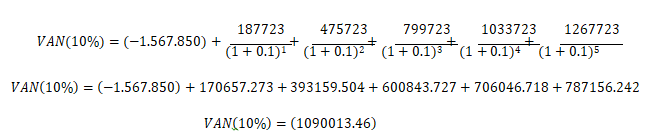
Los beneficios intangibles percibidos en este proyecto corresponden a los siguientes:

* Reconocimiento por parte de los pasajeros: La utilización de la aplicación por parte de las empresas supone un distintivo de la competencia, permitiendo la captación de más “clientes” o pasajeros.
* Disminución en los tiempos de espera: Gracias a que la aplicación permite a las líneas de buses y colectivos publicar actualizaciones del estado de sus máquinas, esto reduce el tiempo de espera por parte de los usuarios en el eventual caso del desperfecto de una maquina en una línea y permite tomar medidas por parte de los pasajeros, quienes pueden buscar otro medio de transporte disponible.

### Cálculo de VAN

Para poder analizar la rentabilidad de un proyecto que no ha sido llevado a cabo, resulta necesario el utilizar herramientas de estimación. Una herramienta existente es el valor actual neto o VAN, esta herramienta permite, al calcular los costos y ganancias (especificadas mediante un estudio de mercado), el estimar los beneficios obtenidos en una supuesta implementación del proyecto, permitiendo así a los stakeholders decidir si apoyar o resistirse a una oferta. A continuación, se presenta la tabla correspondiente al VAN de este anteproyecto de título:

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
|  | **Año 0** | **Año 1** | **Año 2** | **Año 3** | **Año 4** | **Año 5** |
| (+)  Ingresos |  |  |  |  |  |  |
| Beneficios |  | $504.000 | $792.000 | $1.116.000 | $1.350.000 | $1.584.000 |
| (-) Costos |  |  |  |  |  |  |
| Costos de Operación |  | $316.277 | $316.277 | $316.277 | $316.277 | $316.277 |
| (-)  Inversión |  |  |  |  |  |  |
| Costo de desarrollo | $1.567.850 | 0 | 0 | 0 | 0 | 0 |
| **Total** | $1.567.850 | $187.723 | $475.723 | $799.723 | $1.033.723 | $1.267.723 |



El resultado del VAN al ser $1.090.013 y este valor ser positivo, esto indica que el proyecto posee factibilidad financiera y es positivo llevar a cabo su ejecución.

### Enfatizar en los beneficios tangibles e intangibles que se esperan con el proyecto. La cuantificación de los beneficios, ya sean ahorros o ingresos, debe quedar respaldada por la fuente de dicha información y la forma como fueron calculados/estimados los beneficios.

En los costos se incluyen los costos hardware, software, u otras inversiones necesarias para la implantación del sistema. El costo de mano de obra (desarrollo) se considera un ahorro.

**En caso de ser requerida la estimación del Valor actual neto**, considere:

* Vida útil estimada del proyecto: Flujo de gastos del proyecto (gastos de cada año a lo largo de su vida útil) incluyendo la inversión inicial como un gasto (año 0)
* Flujo de ingresos anuales (Ingresos de cada año a lo largo de la vida útil del proyecto), considerando:

## Conclusión de la factibilidad

Breve conclusión de la factibilidad del desarrollo del Sw.

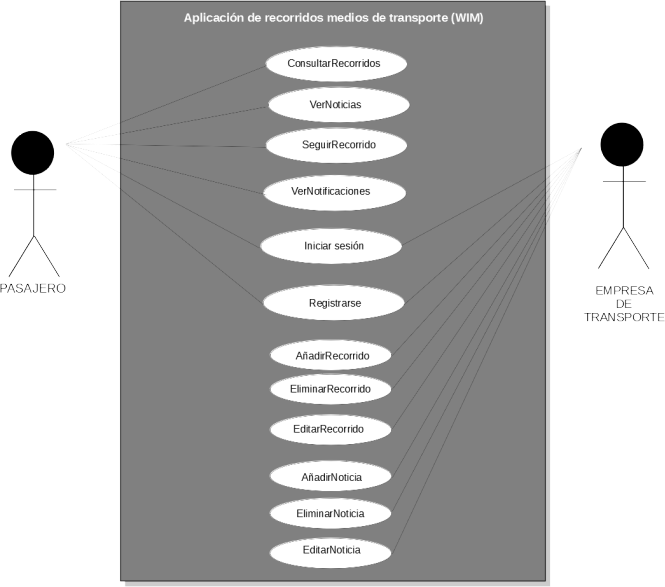
# Análisis

## Procesos de Negocios futuros

Descripciones de los procesos de negocio (UML/BPMN) como punto de partida del modelamiento del software. Debería existir una nueva “vista” del proceso de negocio que solucione de mejor forma el problema, por ejemplo, el e-commerce hace que el proceso de negocio sea totalmente distinto al que existía en la empresa.

## Diagrama de casos de uso

Un diagrama de casos de usos es un modelo que nos permite representar las maneras en las que interactúan los usuarios finales con un sistema, en este caso la aplicación desarrollada en este informe, WIM:



Cada uno de los requerimientos funcionales deben ser representados a través de 1 o más CU. Recuerde que esta técnica no admite DESCOMPOSICIÓN. Este ítem es EXCLUYENTE al ítem 6.1.

### Actores

En este modelo es posible apreciar los dos participantes que corresponden a:

* Pasajero: corresponde a la persona que utiliza la aplicación con la intención de encontrar un medio de transporte.
* Empresa de transporte: corresponde a quien ofrece un recorrido de un medio de transporte a través de la aplicación.

Por cada actor se debe describir:

* Su rol o funciones dentro de la empresa
* Nivel de conocimientos técnicos requeridos
* Nivel privilegio en el sistema y las funcionalidades del software a las cuales tiene acceso

### Casos de Uso y descripción

Aquí se presenta el diagrama de los casos de uso para mostrar la interacción entre los actores y los casos de uso. Además del diagrama se debe describir en forma textual (palabras) la forma como los actores harán uso del sistema. Esta descripción es general y no debe indicar el detalle de la ejecución de cada caso de uso.

### Especificación de los Casos de Uso

Para cada caso de uso (ovalo) del diagrama deben describir el ítem (6.3.3.1)

#### Caso de Uso: <nombre caso de uso>

* Descripción: Descripción general del Caso de Uso
* Pre-Condiciones: Se listan las condiciones que deben cumplirse para que ocurra el caso de Uso 1
* Flujo de Eventos Básicos: Descripción de la secuencia de acciones del caso de uso para clarificar en lenguaje natural lo que el sistema hace cuando el caso de uso es comenzado por un actor y cómo el sistema interactúa con los actores

|  |  |
| --- | --- |
| Al actor | El sistema |
| 1 | 2 |
| 3 | 4 |

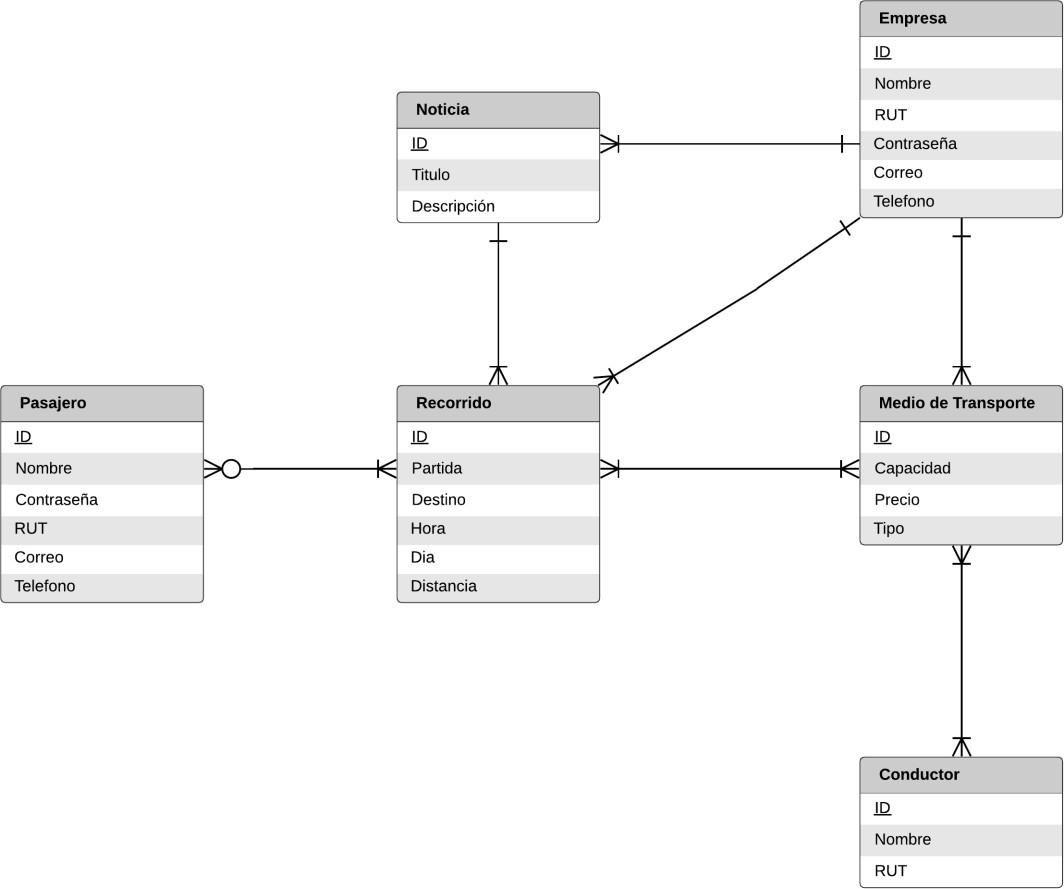
* Flujo de Eventos Alternativo: Se describe cada uno de los flujos alternativos que el caso de uso puede tener.

|  |  |
| --- | --- |
| Al actor | El sistema |
| 2(a) | 5 |
|  |  |

* Post-Condiciones: Lista de posibles condiciones o estados de la base de datos que pueden o deben cumplirse al terminar el Caso de Uso.

## Modelamiento de datos

Un modelo entidad-relación nos permite modelar de una manera simple y gráfica las entidades participantes en una problemática, organización, etc. permitiendo observar sus relaciones y de esta manera ayudando a comprender de mejor manera una problemática o situación.



Las relaciones expuestas en este modelo son descritas a continuación:

* Pasajero-Recorrido: Un pasajero puede buscar 1 a muchos Recorridos.
* Recorrido-Pasajero: Un recorrido puede ser buscado por 0 a muchos pasajeros. Se considera el mínimo como 0 debido a que una empresa puede agregar un recorrido y que nadie consulte por él.
* Recorrido-Medio de Transporte: Un recorrido puede ser realizado por 1 a muchos medios de transporte.
* Recorrido–Empresa: Un recorrido es creado poseído por una única empresa.
* Recorrido-Noticia: Un recorrido puede poseer de 0 a muchas noticias.
* Medio de Transporte-Recorrido: Un medio de transporte puede realizar 1 a muchos recorridos.
* Medio de Transporte-Conductor: Un medio de transporte puede ser conducido por 1 a muchos conductores.
* Medio de Transporte-Empresa: Un medio de transporte es poseído por 1 única empresa. (no se considera la venta de una máquina, debido a que no se necesita llevar un control histórico de los dueños de la máquina)
* Conductor-Medio de Transporte: Un conductor puede conducir 1 o muchos medios de transporte.
* Empresa-Medio de Transporte: Una empresa puede poseer 1 a muchos medios de transporte.
* Empresa-Recorrido: Una empresa puede poseer 1 a muchos recorridos.
* Empresa-Noticia: Una empresa puede publicar de 1 a muchas noticias.
* Noticia-Empresa: Una noticia es publicada por 1 única empresa.
* Noticia-Recorrido: Una noticia es poseída por un único recorrido.

En este ítem se incluye una introducción al modelo y el diagrama. Esta introducción es una explicación, en términos de la empresa, de las entidades o clases y relaciones más representativas del software.

Para el modelamiento se puede utilizar modelos E-R o de clases. Recuerde respetar una codificación para nombrar distintos elementos del modelo.

# Diseño

## Diseño de Físico de la Base de datos

El modelo físico de la Base de datos es la representación de **la organización y estructura de la base de datos** que será utilizada por el Sw, por lo tanto debe ser "consistente" con el modelo especificado en el ítem 6.4.

El diagrama de la Base de datos debe ser incorporado aquí. Dentro del diseño físico de la BD se deben especificar los procedimientos almacenados que podrán ser desarrollados para optimizar el uso de los datos.

## Diseño de arquitectura funcional

Su objetivo es realizar un diseño de la arquitectura funcional del sistema.

Para éste punto se deben especificar las interrelaciones entre todos los módulos del sw (propios y aquellos utilizados desde librerías) y la dependencia de ellos. Para realizar esta actividad se puede construir un Árbol de descomposición funcional, como se muestra en el ejemplo para un sistema de biblioteca en la página 22 de éste documento. Es importante que considere que los últimos niveles de descomposición del árbol corresponderá a los módulos que serán codificados (por ejemplo, librerías, procedimientos - funciones), por lo tanto es importante el nivel de detalle al que se llegue sin olvidar "*implementar módulos reutilizables, maximizar la cohesión y minimizar el acoplamiento*"

Los primeros niveles de descomposición de éste árbol debe ser consistente con el Diagrama de Casos de Uso o el Diagrama de flujos de datos, según corresponda.

## Diseño interfaz y navegación

El diseño de la interfaz de usuario debe considerar un diseño estándar que será respetado en todas las pantallas. En el diseño se considera la organización y el aspecto de la interfaz. El aspecto considera muchos elementos, entre ellos, los colores, imágenes de fondo, uso de iconos entre otros.

La organización de una pantalla considera la ubicación de cada uno de los tipos de elementos de la interfaz, considerando por ejemplo las siguientes áreas: (ver ítem 19.2, página 20)

* De ingresos de datos
* De Botones de opción general
* De botones de opciones específicas a la ventana
* De Menús
* De títulos
* De Barras de Herramientas
* De pie de página
* De Encabezados
* De Logos

El diseño de menú/ navegación considera las opciones / medios que tendrá el usuario para acceder a la funcionalidad del Sw.,

Debe considerar:

* Nombre de ítem y opciones representativas para el usuario
* Organización/ jerarquía representativas para el usuario
* Facilidad de acceso a opciones relacionadas

La jerarquía de menú solo representa los anidamientos y agrupaciones de las opciones de menú y el mapa de navegación representa las opciones que tendrá el usuario para "navegar / recorrer" dentro de las distintas opciones (ver Diagrama para representar la jerarquía de menú, página 22).

## Especificación de módulos

Cada uno de los Procesos del **último nivel** de descomposición del **diseño arquitectónico** funcional deberá corresponder a los módulos de programas que serán construidos en la codificación, por lo tanto deben ser especificados a través del siguiente formato.

Los **módulos de programa creados para esta aplicación** se describen como sigue:

|  |  |  |  |
| --- | --- | --- | --- |
| **N° Módulo**: xxxxx | | **Nombre Módulo**: xxxxxxxxxxxxxxxxxxx | |
| **Parámetros de entrada** | | **Parámetros de Salida** | |
| **Nombre:** | **Tipo de dato:** | **Nombre:** | **Tipo de dato:** |
|  |  |  |  |
|  |  |  |  |

Los módulos de programa utilizados desde librerías externas se describen como sigue:

* Nombre
* Objetivo
* Parámetros

# Pruebas

Adaptación basada en *IEEE Software Test Documentation* Std 829-1998

## Elementos de prueba

Componentes, módulos o sistemas que serán probados. Cada uno de estos elementos se describe brevemente.

## Especificación de las pruebas

Indicar las características que serán probadas, por ejemplo funcionalidad, desempeño, resistencia, interfaz y navegación o seguridad.

* Características a probar. Por ejemplo: funcionalidad, desempeño, resistencia, interfaz y navegación o seguridad.
* Nivel de prueba. Por ejemplo: Unidad, Integración, Sistema, Aceptación
* Objetivo de la Prueba. Que se espera encontrar o detectar
* Enfoque para la definición de casos de prueba. Por ejemplo: Caja negra, caja blanca
* Técnicas para la definición de casos de prueba. Por ejemplo: Técnicas para la definición de casos de prueba (valores límites, particiones, complejidad ciclomática, etc. )
* Actividades de prueba. Procedimiento necesario para ejecutar las pruebas, incluye la preparación o configuración de hardware o software, la ejecución de las pruebas según el orden requerido, la acciones después de ejecutar las pruebas.
* Criterios de cumplimiento. Criterio a cumplir para dar por terminada y superada la prueba.

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| Características a probar | Nivel de prueba | Objetivo de la Prueba | Enfoque para la definición de casos de prueba | Técnicas para la definición de casos de prueba | Actividades de prueba | Criterios de cumplimiento |
|  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |

## Responsables de las pruebas

Detallar los responsables de la ejecución de las distintas pruebas que serán realizadas, ya sean por elementos o por niveles.

## Calendario de pruebas

Calendarización de las distintas actividades de prueba que serán realizadas, ya sean por elementos, niveles o características.

## Detalle de las pruebas

Incluir el detalle de las pruebas según las tablas de anexo 17.

## Conclusiones de Prueba

Concluya respecto al proceso y énfasis de las pruebas realizadas, así como en los resultados obtenidos.

# Plan de Capacitación y entrenamiento

El plan de capacitación debe indicar:

* Usuarios a capacitar
* Tipo de capacitación o entrenamiento
* Funcionalidad o aspectos que serán abordados
* Responsable
* Tiempo estimado hrs
* Calendario. Programación en el tiempo de la capacitación (carta Gantt)
* Recursos requeridos /utilizados

# Plan de Implantación y Puesta en marcha

* Justificación del tipo de puesta en marcha
* El Calendario incluye la programación en el tiempo de la puesta en marcha (carta Gantt) y la funcionalidad que será incorporada en cada etapa.
* Indicaciones de seguridad, respaldos, mantención, planes de contingencia

# Resumen Esfuerzo requerido

El final de este documento se debe indicar las horas destinadas en realizar cada una de las fases del desarrollo del software, las horas corresponden a la suma de las horas gastadas por cada integrante y del equipo en conjunto.

|  |  |
| --- | --- |
| Actividades/fases | N° Horas |
|  |  |
|  |  |
|  |  |
|  |  |
| TOTAL |  |

Comentar los resultados con los datos obtenidos en la sección 14.1.1. En cuanto a la cantidad de líneas de código y el esfuerzo estimado.

# Conclusiones

En primera instancia el alumno debe hacer la contrastación de los objetivos del proyecto y del sistema planteados y alcanzado al final del proyecto.

Se planean conclusiones respecto al ajuste de las herramientas, lenguajes o metodologías utilizadas y la planificación inicial del proyecto.

Para terminar se incluyen conclusiones generales del proyecto desde los puntos de vista:

* Académico
* Personal

# Bibliografía

Formato de referencias y bibliografía según los estándares de biblioteca.

# ANEXO: PLANIFICACION INICIAL DEL proyecto

Carta Gantt u otra herramienta de calendarización con las actividades que serán llevadas a cabo en función de la metodología de desarrollo elegida. Considera actividades desarrolladas por los desarrolladores y los usuarios o clientes.

En caso de metodologías incrementales o evolutivas, se debe especificar la funcionalidad que será abordada en cada iteración o incremento

### Estimación inicial de tamaño

Estimación de Tamaño del software aplicando técnicas basadas en PF o Casos de Uso.

Consulte los valores de la industria utilizados para cuantificar el tiempo (horas de esfuerzo de desarrollo) necesario para implementar 1 Punto de Caso de Uso o 1 Punto de función.

### Contabilización final del tamaño del Sw

* Contabilizar la cantidad de líneas de código implementadas en su software. Especifique junto a los valores la forma como fue calculado el valor (si se consideran líneas en blanco, comentarios o todas las líneas, si se consideran funciones o componentes reutilizados)
* Estimar equivalencia de PF o PCU estimados inicialmente con Líneas de código contabilizadas al final.
* Estimar el esfuerzo horas hombre, considerando las horas de trabajo reales dedicadas al proyecto por cada fase del desarrollo.

# ANEXO: RESULTADOS DE ITERACIONES EN EL DESARROLLO

Si se ha seguido un método/modelo de desarrollo iterativo y/o incremental es necesario adjuntar los hallazgos o resultados obtenidos de las iteraciones.

Es decir se debe indicar el contenido o funcionalidad de la iteración o incremento y los comentarios, correcciones u observaciones de los usuarios.

# ANEXO: MANUAL DE USUARIO

Según se requiera.

# ANEXO: ESPECIFICACION DE LAS PRUEBAS

## Pruebas de Unidad

El siguiente ítem se repite para cada unidad o módulo independiente.

### <nombre unidad>

En este ítem se especifican:

* Las configuración Hw, Sw, SO o de comunicaciones que son necesarias para la prueba.
* Pre condiciones de las pruebas. Por ejemplo en la prueba del módulo de “registra venta” se requiere que existan productos con stock disponible ingresados a la BD.

|  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| ID Caso  De Prueba | Características a Probar | Datos de Entrada | | | | | | | Salida esperada | Salida  Obtenida | Éxito / Fracaso | Observaciones |
| D1 |  |  |  |  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |  |  |  |  |  |  |

## Sistema

En este ítem se especifican:

* Condiciones de la prueba. Esta prueba debe ser ejecutada en un ambiente lo más parecido al que utiliza el usuario.
* Las configuración Hw, Sw, SO, comunicaciones que son necesarias para la prueba.

|  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| Id | Descripción Requerimiento Funcional | Entrada | | | | | | Salida esperada | Salida  Obtenida | Evaluación | |
| D1 |  |  |  |  |  | Éxito / Fracaso | Criticidad en caso Fracaso |
|  |  |  |  |  |  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |  |  |  |  |  |

## Aceptación

Prueba alfa realizada junto al usuario, prueba beta realizada por el usuario sin asistencia del desarrollador.

|  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| Id | Descripción Requerimiento Funcional | Entrada | | | | | | Salida esperada | Salida  Obtenida | Evaluación | |
| D1 |  |  |  |  |  | Éxito / Fracaso | Criticidad en caso Fracaso |
|  |  |  |  |  |  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |  |  |  |  |  |

# ANEXO: DICCIONARIO DE DATOS DEL MODELO DE DATOS

El diccionario completo se incluye como anexo no obstante las tablas principales son descritas en este punto.

# EJEMPLOS (QUITAR ESTE APARTADO)

## ISO/IEC 9126: Tecnología de Información – Evaluación del producto de software

ISO/IEC 9126: Tecnología de Información – Evaluación del producto de software – Características de calidad y Guías para su uso. La ISO 9126 provee un marco de trabajo para la evaluación de la calidad del software, el cual puede ser aplicable a todo tipo de software. Este estándar define seis características que describen la calidad del software:

* Funcionalidad: conjunto de atributos que se sostienen sobre la existencia de un conjunto de funciones y sus propiedades específicas. Las funciones son aquellas que satisfacen necesidades implícitas y establecidas.
* Fiabilidad: conjunto de atributos que se sostienen sobre la capacidad del software para mantener su nivel de rendimiento bajo condiciones establecidas para un período de tiempo establecido.
* Usabilidad: conjunto de atributos que se sostienen sobre el esfuerzo necesario para el uso, y sobre la evaluación individual de tal uso, por un conjunto de usuarios implícitos o establecidos.
* Eficiencia: conjunto de atributos que se sostienen sobre la relación entre el nivel de rendimiento del software y la cantidad de recursos usados, bajo condiciones establecidas.
* Mantenibilidad: conjunto de atributos que se sostienen sobre el esfuerzo necesario para realizar modificaciones especificadas.
* Portabilidad: conjunto de atributos que se sostienen sobre la habilidad del software para ser transferido desde un entorno a otro.

Para cada característica se sugiere un conjunto de subcaracterísticas de calidad, las que se definen a continuación:



## Esquema especificación de Interfaz



1. Menú. Incluye opciones como……
2. Barra de herramientas. Incluye iconos como……
3. Imagen CORPORATIVA
4. Título de ventana con contexto
5. Despliegue e ingreso de datos
6. Botones de optimización / navegación (BUSCAR, INGRESAR NUEVO, VER DETALLES, entre otros)
7. Botones de opción general (GUARDAR, ACEPTAR, CACELAR, CERRAR)
8. Pié de página, sistema, fecha, hora , día, entre otros)

## Diagrama para representar la jerarquía de menú



## Árbol de descomposición funcional



## Estimación de tamaño de Sw: Punto Función

**Identificar las transacciones**

Lista de chequeo para identificar las transacciones (EI, EO, EQ) y datos (ILF, EIF). Cada una de las preguntas debe ser contestada como SI.

Entradas Externas

1. Entran datos desde exterior de la aplicación

2. Existen datos en algún fichero lógico interno que son actualizados

3. El proceso es la unidad mínima de actividad que tiene sentido para el usuario

4. El proceso es completo y deja al sistema en un estado consistente

5. Para el proceso subyacente se debe de cumplir alguna de las siguientes reglas ( A o B)

A La lógica del proceso es exclusiva de esta entrada, o la primera vez que la contamos

B Los datos elementales son diferentes de otras entradas

Salidas Externas

1. El proceso envía datos o información al exterior de la aplicación

2. El proceso es la unidad mínima de actividad que tiene sentido para el usuario

3. El proceso es completo y deja al sistema en un estado consistente

4. Para el proceso subyacente se debe de cumplir alguna de las siguientes reglas ( A o B)

A La lógica del proceso es exclusiva de esta salida (o la primera vez)

B Los datos elementales son diferentes de otras salida

Peticiones o Consultas Externas

1. Una petición atraviesa la frontera del sistema

2. El proceso envía datos o información al exterior de la aplicación

3. Se recuperan datos

4. No se calculan datos derivados para enviar al exterior

5. El proceso (entrada/salida) es la unidad mínima de actividad que tiene sentido para el usuario

6. El proceso es completo y deja al sistema en un estado consistente

7. El proceso no actualiza ningún Fichero Lógico Interno

8. Para el proceso subyacente se debe de cumplir alguna de las siguientes reglas ( A o B)

A La lógica del proceso en su parte de entrada o salida, es distinta del de otras consulta del sistema (o la primera vez)

B Los datos elementales de la entrada o salida son diferentes de otras consultas

**Identificar** **Archivos**

Archivos de Interfaz Interna

1. Se trata de una agrupación de datos lógica o identificable desde el punto de vista del usuario y satisface un requerimiento especifico del usuario

2. La agrupación de datos es mantenida por procesos de la aplicación en estudio

3. La agrupación de datos es mantenida mediante un proceso elemental de la aplicación

4. La agrupación de datos no ha sido contada como un fichero de interfaz externo

Archivos de Interfaz Externa

1. Se trata de una agrupación de datos lógica o identificable desde el punto de vista del usuario y satisface un requerimiento especifico del usuario

2. La agrupación de datos es referenciada, y externa, a la aplicación en estudio

3. La agrupación de datos no es mantenida mediante la aplicación en estudio

4. La agrupación de datos ha sido contada como un fichero lógico Interno en otra aplicación

5. La agrupación de datos no ha sido contada como un fichero lógico Interno de la aplicación en estudio

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| ID Tipo | Descripción | Complejidad | | | Total |
| Simple | Promedio | Compleja |
| EI | Entradas Externas | \*3= | \*4= | \*6= |  |
| EO | Salidas Externas | \*4= | \*5= | \*7= |  |
| EQ | Consultas Externas | \*3= | \*4= | \*6= |  |
| ILF | Archivos Lógicos Internos | \*7= | \*10= | \*15= |  |
| EIF | Archivo de Interfaz Externos | \*3= | \*7= | \*10= |  |
| Total de Puntos de Función sin ajustar (brutos) FC | | | | |  |

* Data Element Type (DET): es un campo único (no repetitivo) reconocible por el usuario
* File Type Referenced (FTR): es un tipo de archivo al que se hace referencia en una transacción; tiene que ser un ILF o EIF
* Record Element Type (RET): es un subconjunto de campos de un archivo, reconocible como tal por el usuario

**DIFICULTAD EO y EQ**

|  |  |  |  |
| --- | --- | --- | --- |
|  | Número de Campos o Atributos de la Salida DET | | |
| FTR | 1-5 DET | 6-19 DET | 20 + DET |
| 0 ó 1 FTR | BAJA | BAJA | MEDIA |
| 2 ó 3 FTR | BAJA | MEDIA | ALTA |
| 4 + FTR | MEDIA | ALTA | ALTA |

**DIFICULTAD EI**

|  |  |  |  |
| --- | --- | --- | --- |
|  | Número de Campos o Atributos de la Entrada DET | | |
| FTR | 1-4 DET | 5-15 DET | 16 + DET |
| 0 ó 1 FTR | BAJA | BAJA | MEDIA |
| 2 -3FTR | BAJA | MEDIA | ALTA |
| 3 + FTR | MEDIA | ALTA | ALTA |

**DIFICULTAD ILF y EIF**

|  |  |  |  |
| --- | --- | --- | --- |
|  | Número de Campos o Atributos DET | | |
| RET | 1-19 Atributos | 20-50 Atributos | 51 + Atributos |
| 1 RET | BAJA | BAJA | MEDIA |
| 2 a 5 RET | BAJA | MEDIA | ALTA |
| 6 o más RET | MEDIA | ALTA | ALTA |

**COMPLEXITY ADJUSTEMENT**

FC1: Comunicación de datos

0: Sistema aislado del exterior

1: Batch, usa periféricos E o S remotos

2: Batch, usa periféricos E y S remotos

3: Captura de datos en línea o teleproceso que pasa los datos o sistema de consulta

4: Varios teleprocesos con mismo protocolo

5: Varios protocolos. Sistema Abierto y con interfaces de todo tipo al exterior.

FC2: Proceso distribuido

0: Sistema totalmente centralizado

1: Sistema realiza procesos en un equipo, salidas usadas vía Sw por otros equipos

2: Sistema captura, los trata en otro

3: Proceso distribuido, transacciones en una sola dirección.

4: idem, transferencia en ambas direcciones.

5: procesos cooperantes ejecutándose en distintos equipos.

FC3: Objetivos de rendimiento

0: Rendimiento normal (no se da énfasis)

1: Se indican requisitos, no medida especial.

2: Crítico en algunos momentos. Procesos acabados antes de la próxima sesión de trabajo.

3: Tiempo de respuesta es crítico.

4: ... en diseño hacer análisis de rendimiento en tiempo respuesta o cantidad oper./hora

5: .. Uso herramientas para alcanzar el rendimiento demandado por el usuario

FC4: Conf. explotación usada intensamente por otros sistemas

0: No se indican restricciones

1: Existen las restricciones usuales

2: Características de seguridad o tiempos.

3: Restricciones en algún procesador

4: El Sw deberá funcionar con restricciones de uso en algún procesador.

5: Restricciones especiales para aplicación en los componentes distribuidos del sistema

FC5: Tasa de transacciones

0: No se prevén picos

1: Se prevén picos poco frecuentes (mensual)

2: Se prevén picos semanales

3: Se prevén horas punta, diarias

4: Tasa de trans. tan elevada que en diseño se hace análisis de rendimiento

5: Análisis de rendimiento en diseño, implementación e instalación.

FC6: Entrada de datos en línea

0: Todo es Batch

1: 1%<entradas interactivas <7%

2: 8%<entradas interactivas <15%

3: 16%<entradas interactivas <23%

4: 24%<entradas interactivas <30%

5: Entradas interactivas >30%

FC7: Eficiencia con el usuario final

0: No se da énfasis al tema

1: 1 a 3 de los factores

2: 4 a 5 de los factores

3: 6 o más factores, sin requerir eficiencia

4: ... con requerimientos que implican estudio de los factores humanos en el diseño

5: … se demandan prototipos y herramientas para verificar que se alcanzaran los objetivos

Eficiencia del usuario con Menús, Uso de ratón, Ayudas "en línea", Movimiento automático del cursor, Efectos de Scroll, Teclas de función predefinidas, Lanzamiento de procesos Batch desde las transacciones "en línea", Selección mediante cursor de datos de la pantalla, Pantallas con muchos colores y efectos, Ventanas de "pop-up", Aplicación bilingüe (cuenta por cuatro), Aplicación Multilingüe (más de dos, cuenta por seis).

FC8: Actualizaciones en línea

0: No hay

1: De 1 a 3 ficheros con información de control. Cantidad baja y ficheros recuperables

2: ... pero con 4 o más ficheros de control

3: Actualización de ficheros importantes

4: ... esencial la protección ante pérdidas

5: Gran cantidad de actualizaciones interactivas. Sistemas de recuperación muy automatizados

FC9: Lógica de proceso interno compleja

0: Ninguna de las características

1: 1 Característica

2: 2 Características

...

5: Las 5 características

Lógica de Proceso Interno Compleja. La complejidad interna en un proceso está en función de las siguientes características: Especificados algoritmos matemáticos complejos, Proceso con lógica compleja, Especificado muchas excepciones, consecuencia de transacciones incompletas, que deberán tratarse, Manejar múltiples dispositivos de entrada/salida, Se incorporaran sistemas de seguridad y control.

FC10: Reusabilidad del código

0: No se prevé

1: Reutilizar código en la misma aplicación

2: Menos de un 10% de la aplicación tiene en cuenta las necesidades de + de 1 usuario

3: El 10 % o más ...

4: Aplicación preparada para ser reutilizable. Nivel de código

5: Aplicación preparada para ser reutilizable. Por medio de parámetros

FC11: Contempla conversión e instalación

0: No se requiere conversión.

1: Se solicita facilidad de instalación

2: Se solicitan procesos de conversión e instalación, no importantes para el proyecto

3: ... si son importantes

4: 2, y herramientas conversión e instalación

5: 3, y herramientas conversión e instalación. Sistema crítico para la empresa

FC12: Facilidad de operación

0: Nada, en todo caso, back-up

1 a 4: Suma de ítems

 Arranque, back-up y recuperación

 Idem, sin intervención operador ( X2 )

 Minimizar necesidad de disp. externos almac.

 Minimiza necesidad de manejar papel

5: Sistema automático sin intervención humana

FC13: Instalaciones múltiples

0: 1 solo lugar

1: Múltiples lugares, mismo Hw y Sw

2: En diseño se tiene en cuenta el caso (1)

3: En diseño se tiene en cuenta múltiples entornos Hw y Sw

4: Se documenta y planea para (1) y (2)

5: Idem, para (3)

FC14: Facilidad de cambios

0: No se especifica nada

1: Un ítem de valor 1

2: Ítems por valor 2

3: ...

5: Ítems por valor 5

Ítems a tener en cuenta:

• Consultas flexibles del usuario:

• Simples con condiciones. lógicas and/or que implican un único fichero lógico

• Medias con cond. lógicas sobre más de 1 F.L. (X2)

• Complejas con condiciones lógicas complejas que afectan a varios F.L. (X3)

• Parámetros de la aplic. con tablas ajenas al código:

• El cambio se hace efectivo al arrancar el sistema

• El cambio es interactivo (X2)

Processing Complexity Adjustement PCA

PCA = 0.65 + (0.01\*PC)=

Function Point Measure FP

FP = FC \* PCA =

## Estimación de tamaño de Sw: Puntos de Casos de Uso

• Clasificar Actores

• Clasificar casos de uso

• Factores técnicos

• Factores del entorno

• Calcular puntos de Casos de uso

|  |  |  |  |
| --- | --- | --- | --- |
| Tipo de caso de uso | |  |  |
| 5 | Simple | Menos de 5 clases 5 | 3 transacciones o menos |
| 10 | Medio | 5 a 10 clases 10 | 4 a 7 transacciones |
| 15 | Complejo | Más de 10 clases 18 | Más de 7 transacciones |

|  |  |  |
| --- | --- | --- |
| Tipo de actor D | |  |
| 1 | Simple | Otro sistema que interactúa con el sistema a desarrollar mediante una interfaz de programación (API). |
| 2 | Medio | Otro sistema interactuando a través de un protocolo (ej. TCP/IP) o una persona interactuando a través de una interfaz en modo texto |
| 3 | Complejo | Una persona que interactúa con el sistema mediante una interfaz gráfica (GUI). |

Calcular UUCP (Unadjusted Use Case Point)

UUCP= UAW+UUCW

Calcular TCF (Technical Complexity Factor)

TCF=0.6+(0.01\*TFactor)

Calcular EF (Environmental Factor)

EF=1.4+(-0.03\*EFactor)

UCP = UUCP \* TCF \* EF

Calculate TCF (Technical Complexity Factor)

|  |  |
| --- | --- |
| Technical Factor | Multiplier |
| Distributed System | 2 |
| Application performance objectives, in either response or throughput | 1 |
| End-user efficiency (on-line) | 1 |
| Complex internal processing | 1 |
| Reusability, the code must be able to reuse in other applications | 1 |
| Installation ease | 0,5 |
| Operational ease, usability | 0,5 |
| Portability | 2 |
| Changeability | 1 |
| Concurrency | 1 |
| Special security features | 1 |
| Provide direct access for third parties | 1 |
| Special user training facilities | 1 |

|  |  |
| --- | --- |
| Descripción | Valor |
| Irrelevante De | 0 a 2. |
| Medio De | 3 a 4. |
| Esencial | 5 |

|  |  |
| --- | --- |
| Environmental Factor | Multiplier |
| Familiar with Objectory + RUP | 1,5 |
| Application experience | 0,5 |
| Object Oriented experience | 1 |
| Analyst capability | 0,5 |
| Motivation | 1 |
| Stable requirements | 2 |
| Par time workers | -1 |
| Difficult programming language | -1 |

Level of Effort. Schneider and Winters, proponen que: Si la suma entre (el número de factores de entorno (F1 a F6) superiores a 3 y el número de factores de entorno (F7 a F8) inferiores a 3).

* es menor o igual a 2 entonces LOE=20,
* es 3 o 4 LOE=28.
* es mayor a 4 reconsiderar el proyecto. Por ejemplo reducir los riesgos relacionados con los factores de entorno.

## Aspectos de Seguridad Informática a considerar en proyectos de Sw

Seguridad: Decisiones y acciones tendientes a llevar los riesgos a un nivel aceptado (Eliminarlos, reducirlos o asumirlos)

La seguridad apunta a proteger: Los recursos existentes y los procesos que apuntan a la generación de nuevos recursos

|  |  |  |
| --- | --- | --- |
| AREAS DE RIESGO | ASPECTOS A ASEGURAR | MEDIDAS DE SEGURIDAD |
| PROYECTO | Sustentabilidad económica | Estudio de Factibilidad Económica: VAN y TIR |
| HARDWARE | Integridad física del HW | Seguros; resguardos para evitar destrucción, mantención |
| Operatividad del HW | Verificar configuración y compatibilidades de SW básico |
| PERSONAS DEL EQUIPO DE DESARROLLO | Calidad de la gestión Informática | Seguir indicaciones de propuesta CobiT |
| Calidad de la Concepción del Sistema | Formalizar problema-requerimientos empresa-requerimientos TIC-Restricciones |
| Calidad del Diseño Lógico | Asegurar calidad de interfaces apropiadas para usuario |
| Formalizar aprobación de E/S por Usuario |
| Calidad del Diseño Físico | Documentar rigurosamente aspectos técnicos del sistema |
| Procedimiento de creación del ambiente de desarrollo |
| Apropiada arquitectura de HW, módulos de SW y Datos |
| Calidad de la Construcción de Sistema | Control de concordancia con el diseño |
| Calidad de la operación del Sistema | Existencia de instrucciones de operación adecuadas |
| PRODUCTO DE SOFTWARE | Calidad: Confiabilidad-Legalidad-Disponibilidad  Efectividad-Eficiencia-Flexibilidad y otros | Seguimiento para verificar la calidad |
| Acciones frente a fallas de software | Procedimientos de mantención y correctivos ágiles |
| CONTINUIDAD DEL SISTEMA EN OPERACIÓN | Continuidad frente a catástrofe con pérdida de integridad lógica y/o Física del Sistema (programas y datos) | Respaldos automatizados de Sistema completo |
| Procedimiento de restauración automatizado de Sistema |
| Disponer de log histórico en sistemas transaccionales |
| Reprocesos Batch en sistemas transaccionales |

El riesgo (probabilidad de falla) de cada aspecto se puede establecer de las siguientes formas

1-En base a estadísticas de fallas para sistemas antiguos

2-En base a investigación de literatura disponible

3-En base a criterios lógicos