**UNIVERSIDAD ANDRES BELLO**

**FACULTAD DE INGENIERÍA**

**INGENIERÍA EN COMPUTACIÓN E INFORMÁTICA**



Tarja Móvil

Seminario de Título 1

**Alejandro Adam Concha**

**VIÑA DEL MAR – CHILE**

**Marzo 2019**

Contenidos

[1. Introducción 8](#_Toc535112997)

[2. Fundamentación del Problema 11](#_Toc535112998)

[2.1. Análisis de la situación actual. 11](#_Toc535112999)

[2.2. Análisis de la problemática. 12](#_Toc535113000)

[2.2.1. Técnica de los cinco por qué. 12](#_Toc535113001)

[2.2.2. Objetivos. 13](#_Toc535113002)

[2.2.3. Objetivo general. 13](#_Toc535113003)

[2.2.4. Objetivos específicos. 13](#_Toc535113004)

[2.2.5. Diagrama de alto nivel. 14](#_Toc535113005)

[2.3. Arquitectura de la solución. 15](#_Toc535113006)

[2.4. Solución ideal. 16](#_Toc535113007)

[2.5. Limitaciones. 16](#_Toc535113008)

[2.6. Restricciones. 16](#_Toc535113009)

[2.7. Alternativas de solución. 17](#_Toc535113010)

[2.7.1. Procedimientos manuales. 17](#_Toc535113011)

[2.7.2. Cambios en procedimientos actuales. 17](#_Toc535113012)

[2.7.3. Alternativas disponibles en el mercado. 17](#_Toc535113013)

[2.8. Solución propuesta. 18](#_Toc535113014)

[2.9. Factibilidades. 18](#_Toc535113015)

[2.9.1. Factibilidad operativa 18](#_Toc535113016)

[2.9.2. Factibilidad técnica 19](#_Toc535113017)

[2.9.3. Factibilidad Económica 19](#_Toc535113018)

[2.10. Costos asociados 19](#_Toc535113019)

[2.10.1. Costo Recursos Humanos 19](#_Toc535113020)

[2.10.2. Costo Hardware 20](#_Toc535113021)

[2.10.3. Costo Software 20](#_Toc535113022)

[2.10.4. Costo de capacitación 21](#_Toc535113023)

[2.10.5. Costo Total 21](#_Toc535113024)

[2.11. Factibilidad Legal 22](#_Toc535113025)

[2.12. Diseño de alto nivel. 23](#_Toc535113026)

[2.13. Requerimientos de alto nivel. 24](#_Toc535113027)

[2.13.1. Requerimientos Sistema: 24](#_Toc535113028)

[2.13.2. Requerimientos Funcionales: 24](#_Toc535113029)

[2.13.3. Requerimientos No Funcionales: 24](#_Toc535113030)

[2.13.4. Requerimientos de Dominio: 25](#_Toc535113031)

[3. Planificación del proyecto. 27](#_Toc535113032)

[3.1. Metodología de desarrollo. 27](#_Toc535113033)

[3.1.1. Ventajas 28](#_Toc535113034)

[3.1.2. Inconvenientes 28](#_Toc535113035)

[3.1.3. Roles dentro del proyecto 28](#_Toc535113036)

[3.1.4. Planificación del proyecto. 28](#_Toc535113037)

[3.2. Gestión de la configuración. 29](#_Toc535113038)

[3.2.1. Etapas 29](#_Toc535113039)

[3.2.2. Proponer cambios 30](#_Toc535113040)

[Implementar los cambios 33](#_Toc535113041)

[1.1 Participantes 35](#_Toc535113042)

[1.1.1 Representante de SCM 35](#_Toc535113043)

[1.1.2 Desarrolladores 36](#_Toc535113044)

[1.1.3 CCB 36](#_Toc535113045)

[1.1.4 Unidad de SQA 36](#_Toc535113046)

[1.1.5 Jefe de proyecto 36](#_Toc535113047)

[3.3. Gestión de cambios. 39](#_Toc535113048)

[3.4. Gestión de riesgos. 39](#_Toc535113049)

[3.6. Entorno de desarrollo. 41](#_Toc535113050)

[4. Resultados 46](#_Toc535113051)

[4.1. Análisis de resultados. 46](#_Toc535113052)

[4.1.1. Hito 2 46](#_Toc535113053)

[4.1.2. Planificación del proyecto 46](#_Toc535113054)

[4.1.3. Problemas y Riesgos 46](#_Toc535113055)

[4.1.4. Resultado 47](#_Toc535113056)

[4.1.5. Lecciones aprendidas 48](#_Toc535113057)

[4.1.6. Hito3 49](#_Toc535113058)

[4.1.7. Requerimiento Hito3 49](#_Toc535113059)

[4.1.8. Planificación Hito3 49](#_Toc535113060)

[4.1.9. Resultados 49](#_Toc535113061)

[4.1.10. Lecciones aprendidas 51](#_Toc535113062)

[4.2. Situación futura. 51](#_Toc535113063)

[4.3. Resultados esperados. 52](#_Toc535113064)

[2 5. Conclusiones. 55](#_Toc535113065)

[2.1 5.1. Conclusión. 55](#_Toc535113066)

[2.2 5.2. Problemas abiertos. 55](#_Toc535113067)

[2.3 5.3. Trabajo futuro. 55](#_Toc535113068)

Resumen

El presente documento que tiene por finalidad entregar una propuesta de solución a la problemática que presenta la empresa SAAM Extraportuarios S.A., en relación al proceso de registro del estado de los contenedores y de la carga que ingresa o sale de los terminales extraportuarios. Proceso que es llamado tarja. (1)

Saam Extraportuario S.A. bajo las siglas AEP realiza el proceso Tarja en forma manual, desde la planificación, ejecución y elaboración del informe tarja.

Como todo proceso que es llevado en forma manual, está sujeto a diversas dificultades, tales como lentitud en el proceso, errores en la toma de estados, omisión de registro fotográficos y con la intervención de un gran número de personas en especial para la elaboración del informe tarja.

Dado a lo anterior, se propone desarrollar una aplicación que automatice el proceso tarja, logre disminuir los tiempos de ejecución, permita llevar trazabilidad, y una en un solo documento el estado de la carga y contenedor con un registro fotográfico completo, agregando valor al servicio que se entrega a los distintos Forwarders.

Capítulo 1: Introducción

# Introducción

SAAM es una empresa multinacional de origen chileno que presta servicios al comercio internacional a través de sus tres divisiones de negocios: Terminales Portuarios, Remolcadores y Logística. Con más de 50 años de experiencia, SAAM está presente en 13 países del Norte, Centro y Sur de América, generando empleo a más de 8 mil trabajadores.

Actualmente somos uno de los principales operadores portuarios de América y líder en servicios de remolcadores en el continente y cuarto a nivel mundial. En los distintos mercados donde opera, SAAM está asociado a operadores locales y globales estratégicos. Entre ellos destacan SSA Marine, el mayor operador de terminales en Estados Unidos; SMIT, segundo mayor operador de remolcadores del mundo y filial del grupo holandés Boskalis y American Airlines.

SAAM constituye el principal activo de Sociedad Matriz SAAM S.A., sociedad anónima abierta constituida el año 2011 y cuyas acciones se cotizan en la Bolsa de Comercio de Santiago de Chile, siendo parte del IPSA que reúne a las 40 principales empresas del país.

Hoy SAAM es líder en la prestación de servicios integrados a los Navieros, así como a Exportadores e Importadores, en todo el proceso de movilización de carga.

En el año 1989 Se crea Almacén Extraportuario de SAAM (Primer Terminal Extraportuario fuera de un puerto), primera empresa privada en Chile que participa en la prestación de Servicios en Zona Primaria, brindando servicios de logística para todo tipo de carga.

En el año 2000 Se crea Saam Extraportuarios S.A. con giro único para entregar un mejor servicio a nuestros clientes. Se crea SAAM Extraportuarios SAI, nueva sucursal ubicada a un costado del acceso sur del puerto de San Antonio.

En el año 2004 Se crea SAAM Extraportuarios Placilla, nueva sucursal, la cual cuenta con un almacén de última generación, especialmente construido para el almacenamiento de vinos y licores. Adicionalmente cuenta con un moderno Centro de Transferencia para servicios de consolidado y desconsolidado de Fruta fresca de importación, exportación y tránsito a terceros países.

Los múltiples acuerdos comerciales que Chile a tomado especialmente con países del Asia nos ha permitido experimentar un fuerte crecimiento en las exportaciones e importaciones, La OEC (the Observatory of Economic Complexity) establece a Chile como la 41° mayor economía de exportación y como la 44º importador más grande en el mundo.

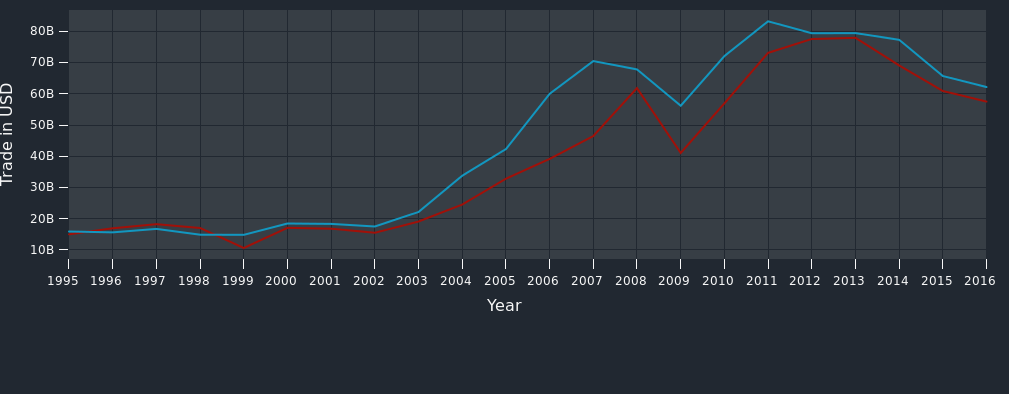


Figura 1.0 Situación Exportación e importación

Este fuerte crecimiento ha permitido que ingresen nuevos actores al servicio extraportuario tales como Agunsa, Puerto Columbo, Sitrans, Texval, SEAPORT, El Sauce, entre otros.

La incorporación de estos nuevos actores autorizados por la Aduana de Chile, ha obligado a modernizar e introducir mayor tecnología a los procesos y servicios que actualmente se prestan. Ya que una de las formas de aumentar la participación en el mercado es incorporando tecnología en los servicios complementarios. El primero en introducir dicha tecnología fue SAAM quien incorporó a su sistema corporativo (Torpedo) el estado de los contenedores tanto en el ingreso como en la salida llamado reporte de carga, luego incorpora la tarja electrónica, que consiste en registrar el estado de la carga al momento de la apertura de la unidad (contenedor) y cierre de este. Proceso netamente documental que está ligado al movimiento de carga y permanencia dentro de los terminales.

Estos informes fueron rápidamente adoptados por los nuevos participantes como exigencia Aduanera, lo que generó migración de algunos Forwarders.

SAAM dentro de los servicios que ha considerados críticos se encuentra el proceso Tarja que es realizado en faena en forma manual. Este proceso consiste en registrar el estado de la carga acompañado de un registro fotográfico entregando como resultado un informe al cliente Forwarder.

Para esta ello, se ha solicitado generar una propuesta de solución consistente en un sistema informático que permita automatizar el proceso tarja, con el fin de integrar en forma rápida, un informe con el mayor detalle del estado de la carga asociada a un consignatario el que será entregado al cliente Forwarder.

Capítulo 2: Fundamentación del problema

# Fundamentación del Problema

# Análisis de la situación actual.

SAAM Extraportuario en los últimos cinco años ha ido experimentando un fuerte crecimiento en los servicios de consolidado, desconsolidado y Despacho de carga bajo acuerdo comercial con los principales Forwarders del país. Bajo estos acuerdos se establece mantener mayores controles apuntando a la trazabilidad y a la seguridad entregando información precisa y oportuna de los servicios contratos.

Cada servicio es planificado el día anterior a la faena, en donde se establecen las unidades(contenedores), la cuadrilla que trabaja por cada unidad, se indica número de personas que se requieren (tarjador, paletizador, grúa, etc). Al día siguiente antes del inicio de faena se imprime todos los documentos asociados (Manifiesto) a la tarja, las que son distribuidas a los tarjadores. Siguiendo la planificación se ubica la cuadrilla e inicia la actividad asignada, el trabajador registra en su planilla todos los movimientos de carga y lo acompaña con fotografías de la mercancía y de la marca (etiqueta) una vez terminado, el funcionario AEP recepciona la documentación, las fotos y prepara el informe tarja que es enviado vía correo al cliente Forwearder.

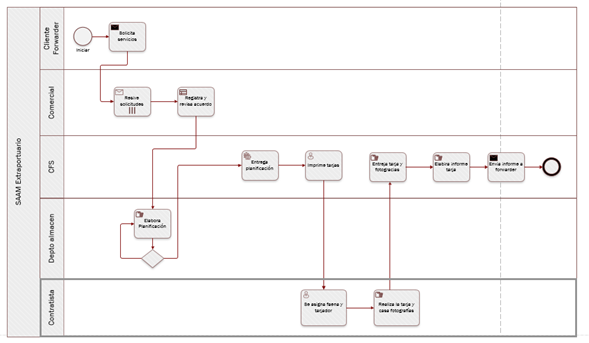


Figura 2.1.1 Situación actual

Para entender las actividades que se realizan es necesario revisar algunos conceptos.

Freight Forwarder (FF o embarcador) es un agente que le presta sus servicios expertos a los exportadores e importadores, gracias a su conocimiento cabal sobre las reglas y regulaciones en esta materia, tanto del país de origen como de los países de destino, así como de los métodos de envío y los documentos relacionados con el comercio exterior.

Consolidado: Actividad que permite agrupar diferentes embarques (carga) de uno o varios consignatarios, para ser transportados bajo un solo documento de transporte.

Desconsolidado: Actividad que permite desagrupar embarques consolidados en un mismo documento de transporte u otro equivalente y que vienen destinados a diferentes consignatarios, presentando cada embarque individual con su respectivo documento de transporte hijo. Procede en el ingreso de mercancías al territorio aduanero.

Tarja Documentos cuyo principal propósito es poder registrar el estado de la carga en el instante en que esta es transferida, consolidada o desconsolidada, recepcionada o entregada en patio, o cuando entra o sale de los recintos portuarios.

# Análisis de la problemática.

# Técnica de los cinco por qué.

Para la identificación de la causa raíz utilice la técnica de los 5 ¿por qué?

1. ¿Por qué se generan retrasos en la entrega del informe tarja al cliente Forwarder?
   1. Porque el proceso tarja es lento, tanto en la faena como en la generación de los informes que son enviados al cliente.
2. ¿por qué es lento el proceso tarja?
   1. Porque es un proceso manual, desde la planificación hasta la generación del informe Tarja, lo que genera retrasos en el término de la faena aumentó los costos operacionales.
3. ¿Por qué se generan aumento en los costos de operación?
   1. Porque se extiende la jornada de trabajo para cumplir con lo planificado para que se revisen y elaboran el documento informe tarja. Lo que lleva a cometer errores de digitacion, falta de imágenes, etc.
4. ¿Por qué es importante disminuir los errores?
   1. Porque, el informe tarja es un documento oficial que se usa como medio de prueba, que el cliente usa para cobro de las responsabilidades.
5. ¿Por qué es importante mantener conforme al cliente forwarder?
   1. Porque hay acuerdos comerciales establecidos.

**Problemática**

El alto número de tarjas que se realizan diariamente hace que se retrase el informe tarja, ya que se debe recopilar los datos para generar el informe que suele entregarse con errores.

# Objetivo general.

Desarrollar una aplicación móvil que permita automatizar el proceso tarja con el fin de obtener el informe tarja en el menor tiempo posible.

# Objetivos específicos.

* OE1: Contar con un sistema tarja que permita planificar las actividades diarias.
* OE2: Contar con un sistema móvil asincrónico que no dependa de la red wifi.
* OE3: El sistema permitirá realizar la planificación diaria de las actividades tarja
* OE4: El sistema tarja registrará el estado de la carga con fotografías asociada.
* OE5: Disminuir los tiempos de faena por cada tarja
* OE6: Disminuir los tiempos de entrega del informe tarja
* OE7: Disminuir los errores en el documento informe tarja



Figura 2.3.2 Métrica

# Diagrama de alto nivel.

A continuación, se presenta el diagrama de alto nivel de la solución sistema tarja electrónica para SAAM Extraportuarios S.A.

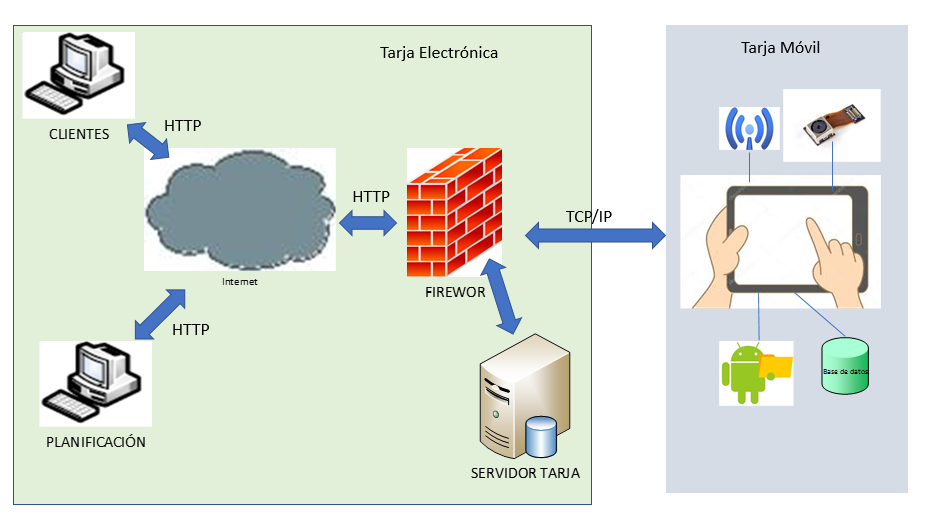


Figura 2.4.1 Diagrama de Alto Nivel

# Arquitectura de la solución.

En los inicios de la informática, la programación se consideraba un arte y se desarrollaba como tal, debido a la dificultad que entrañaba para la mayoría de las personas, pero con el tiempo se han ido descubriendo y desarrollando formas y guías generales, con base a las cuales se puedan resolver los problemas. A estas, se les ha denominado Arquitectura de Software, porque, semejanza de los planos de un edificio o construcción, estas indican la estructura, funcionamiento e interacción entre las partes del software. (2)

Para el desarrollo de este proyecto se ocupará la arquitectura de 3 capas, la cual es una especialización de la arquitectura cliente-servidor donde la carga se divide en tres partes con un reparto claro de funciones: una capa para la presentación, otra para el procesamiento y otra para el almacenamiento.

1. Capa de presentación: presenta el sistema al usuario, le comunica la información y captura la información del usuario en un mínimo de proceso. Esta capa se comunica únicamente con la capa de negocio. También es conocida como interfaz gráfica y debe tener la característica de ser entendible y fácil de usar para el usuario.
2. Capa de negocio: es donde residen los programas que se ejecutan, se reciben las peticiones del usuario y se envían las respuestas tras el proceso. Se denomina capa de negocio porque es aquí donde se establecen todas las reglas que deben cumplirse. Esta capa se comunica con la capa de presentación, para recibir las solicitudes y presentar los resultados, y con la capa de datos, para solicitar al gestor de base de datos para almacenar o recuperar datos de él. Tambien se consideran aquí los programas de aplicación.
3. Capa de datos: es donde residen los datos y es la encargada de acceder a los mismos. Está formada por uno o más gestores de bases de datos que realizan todo el almacenamiento de datos, reciben solicitudes de almacenamiento o recuperación de información desde la capa de negocio.

La ventaja principal de esta arquitectura es que el desarrollo se puede llevar a cabo en varios niveles y, en caso de que sobrevenga algún cambio, sólo se ataca al nivel requerido sin tener que revisar entre código mezclado.

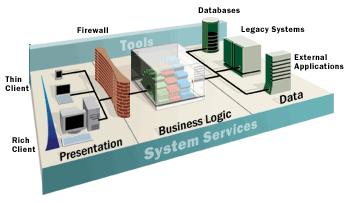


Figura 2.5.1 Diagrama tres capas

# Solución ideal.

El alcance de la solución propone que este sistema automatizar la operación tarja en todos los terminales de Saam Extraportuarios, disminuyendo los tiempos y entregando un valor agregado al servicio que se entrega.

# Limitaciones.

* El proyecto contará con una aplicación web que permita planificar las tarjas,
* Solo tendrán acceso las personas autorizadas.
* El proyecto contará con una aplicación móvil que permita realizar la faena tarja consolidado, tarja desconsolidado y tarja despacho.
* Lo equipos móviles se conectarán a la red wifi interna dedicada para este fin
* Los equipos móviles deben funcionar en forma asincrónica.
* Los equipos móviles estar bajo plataforma android 5.5 o superior, debe contar con cámara incorporada de 8 megapíxeles o superior.
* Los equipos serán IP66 o superior. Para uso exterior.
* Los equipos deben ser de uso industrial.

# Restricciones.

* La aplicación deberá mantener sesión de usuario.
* Los usuarios no tendrán acceso a internet
* El sistema deberá almacenar por un periodo de 3 meses las tarjas generadas.

# Alternativas de solución.

# Procedimientos manuales.

Como primera alternativa, se considera continuar con el proceso manual, aumentar la dotación creando nuevas cuadrillas con el fin de disminuir los tiempos en la faena.

Para la generación del informe tarja se establecería un equipo dedicado a este fin, los cuales recogerían los datos entregados por los tarjadores con las fotografías asociadas, generando el documento informe tarja con ello, disminuimos los tiempos de entrega.

Para garantizar, que el informe esté correcto, se modificará el proceso actual para incorporar una función de control informe quien revisará antes de ser despachado al cliente forwarder.

# Cambios en procedimientos actuales.

Si bien es cierto existen procedimientos para la actividad tarja, estos se deberán modificar para ajustarlos a las nuevas necesidades.

Se debe modificar los procedimientos de creación de informe tarja ya que, se debe contratar personal con dedicación exclusiva para la recolección y generación del informe.

Se debe incorporar un nuevo cargo supervisor tarja con su respectivo procedimiento el que permitirá supervisar la elaboración del informe tarja para luego revisar y autorizar el informe antes de ser despachado al cliente evitando los errores.

# Alternativas disponibles en el mercado.

No se ha encontrado un software en el mercado nacional e internacional que cubra la necesidad de los terminales extraportuarios zona primaria del país, si bien es cierto hay sistemas que permite llevar registro del estado de las cargas, están más bien ligados al área logística, aeropuertos y puertos en general.

Las empresas ligadas a este rubro han desarrollado sus propios sistemas los que son revisados y autorizados por la autoridad competente la Aduana de Chile. Ejemplo de ello, es Puerto Columbo que lanzó el sistema de tarja electrónica. En el año 2017 (http://www.dycsa.cl/extraportuario/)

# Solución propuesta.

1. La solución propuesta consiste en desarrollar la solución Tarja en dos ambientes distintos. Por un lado construir un ambiente web donde el personal AEP, realice la planificación de las actividades diarias (consolidado, desconsolidado, despacho), administre y mantenga los recursos (usuarios, terminales, nave, puertos, gruas, tarjadores, etc) y permita que los clientes autorizados revise el estado de las trajas y pueda descargar el informe tarja. Por otro lado se requiere una aplicación móvil asincrónica que permite visualizar lo planificado por tarja electrónica, asociado a un terminal, a un tipo de faena (consolidado o desconsolidado), para luego tomar una unidad y realizar la tarja, que consiste en tomar registro de los estados del contenedor y de su carga asociada a un consignatario con registro fotográfico las que serán almacenadas en una base de datos local, una vez finalizada realizar una sincronización enviarlas como archivo plano a un servidor central, las que serán disponibilizadas al cliente forwarder para su descarga.

# Factibilidades.

Después de evaluar las distintas alternativas es necesario realizar un estudio de factibilidad considerando aspectos relevantes como el acceso a la tecnología, así como los costos, beneficios y el grado de aceptación que la propuesta genera en la empresa.

Este análisis permitió determinar las posibilidades de diseñar el sistema propuesto y su puesta en marcha, los aspectos tomados en cuenta para este estudio fueron clasificados en cuatro áreas que se describen a continuación:

# Factibilidad operativa

La factibilidad operativa permite predecir, si se pondrá en marcha el sistema propuesto, aprovechando los beneficios que ofrece a todos los usuarios involucrados, ya sea los que interactúan directamente con él, cómo también aquellos que recibirán información producida por el mismo sistema.

Por otra parte, el correcto funcionamiento de la aplicación siempre estará supeditado a la capacidad de los empleados para utilizarlo. La necesidad y el deseo de utilizar una herramienta que sirva de apoyo al flujo de trabajo permitió identificar las necesidades de manera más amigable y sencilla. Basándose en las entrevistas, reuniones y seguimientos durante el flujo de trabajo, la aplicación que se va a desarrollar es factible operacionalmente.

# Factibilidad técnica

La factibilidad técnica consistió en realizar una evaluación de la tecnología existente en la empresa y la posibilidad de hacer uso de estos en la implementación de la solución. La empresa cuenta con computadores con las características necesarias para el uso del sistema, servidores con motor base de datos requerido y con sus licencias asociadas que son suficientes para el desarrollo del sistema. Si bien es cierto, la empresa cuenta con dispositivos móviles, para este proyecto se ha evaluado distintos equipos optando por adquirir Tablet de tipo industrial IP68 de marca Explore M60. Las características de esta Tablet permiten contar con un dispositivo portátil Android robusto. La combinación de las tecnologías inalámbricas de wifi, GPS, Bluetooth, datos y voz LTE y NFC proporciona un pase de acceso total a los sistemas de generación de información IoT. Su cámara de alta resolución y su lector de códigos de barra 1D/2D Imagen de Área SR (Standard Range) permiten obtener resultados de alta calidad con alta precisión de lectura. También cuenta con una batería de larga vida útil, reemplazable por el usuario, que dura hasta 22 horas.

# Factibilidad Económica

La Factibilidad Económica es de vital importancia, debido a que la empresa debe poseer

los recursos necesarios para la implementación del sistema. El siguiente estudio permite

visualizar si se puede solventar y financiar los recursos necesarios del proyecto.

# Costos asociados

# Costo Recursos Humanos

Se considera para este proyecto dos desarrolladores, con dedicación parcial de 3 horas día por un periodo de estimado para finalizar la aplicación de 10 meses.

El salario estimado por hora salario por hora es de $4.923 pesos.

Considerando que el sistema está siendo desarrollado por dos personas, de lunes a viernes trabajando en promedio 3 horas diarias, se estima que el salario mensual para ambos desarrolladores es de $ 590.760 mil pesos.

|  |  |
| --- | --- |
| **Salario** | **$295,380** |
| Meses | 10 |
| Cantidad desarrolladores | 2 |
| **Costo Total RRHH** | **$5,907,600** |

Tabla 2.8.1.1. Costo Personal.

# Costo Hardware

Considerando que la empresa cuenta con todos los equipos necesarios para el funcionamiento del sistema, el único costo en hardware identificado es el de los dispositivos móviles. Xplore M60

|  |  |
| --- | --- |
| **Valor Tablet Xplore M60** | **$872,417** |
| cantidad dispositivos | 10 |
| **Costo Total equipos** | **$8,724,170** |

Tabla 2.8.1.2. Costo Hardware

# Costo Software

Los costos de los programas utilizados para la realización del proyecto se especifican en

la tabla 7.0

|  |  |  |  |
| --- | --- | --- | --- |
| **Software** | **Tipo** | **Licencias** | **Costo** |
| Windows 7 | Sistema Operativo | disponible | 0 |
| Microsoft Office | Herramientas de apoyo | disponible | 0 |
| Visual Estudio 2013 | Entorno desarrollo | disponible | 0 |
| SQL Server 2016 | Software de Gestión de Base de Datos | disponible | 0 |
| **Total Software** |  |  | **0** |

Tabla 2.8.1.3. Costo Software

# Costo de capacitación

La capacitación consistirá en dos sesiones de 45 min. Con un costo total de $100.000 pesos por seción. Esta tiene como objetivo fortalecer los conocimientos en desarrollo en aplicaciones móviles a los desarrolladores de SAAM

# Costo Total

Los costos finales para la realización del proyecto están dados por:

CT = Costo Personal + Costo de Hardware + Costo de Software + Costo de Capacitación

|  |  |
| --- | --- |
| **Concepto** | **Valor** |
| Costo Personal | $5,907,600 |
| Costo Hardware | $8,724,170 |
| Costo Software | $0 |
| Costo Capacitación | $200,000 |
| **Total Costos** | **$14,731,770** |

Tabla 2.8.1.4. Costo Total aprobado

Dada la estructura organizativa de la empresa, en donde la gerencia de comunicaciones y sistemas entrega el servicio de desarrollo y de infraestructura, estos costos serán incorporados al proyecto el que será activado a 3 años para el área logística.

# Factibilidad Legal

El objetivo del estudio de factibilidad legal es el poder verificar que en el desarrollo de un sistema no incurra en infracciones, violaciones y otros delitos impidan la puesta en marcha de sistema.

Para el desarrollo del presente proyecto no existen trabas legales que impidan el buen desempeño y funcionamiento del software, puesto que no se incurren en infracciones a las

leyes vigentes hoy en día, las cuales se especifican a continuación:

Ley Nº 19.223, Relativa a delitos Informáticos que detalla solo cuatro artículos que se

describen a continuación:

Artículo 1º

El que maliciosamente destruya o inutilice un sistema de tratamiento de información o sus partes o componentes, o impida, obstaculice o modifique su funcionamiento, sufrirá la pena de presidio menor en su grado medio a máximo.

Si como consecuencia de estas conductas se afectarán los datos contenidos en el sistema, se aplicará la pena señalada en el inciso anterior, en su grado máximo.

Artículo 2º

El que con el ánimo de apoderarse, usar o conocer indebidamente de la información contenida en un sistema de tratamiento de la misma, lo intercepte, interfiera o acceda a él, será castigado con presidio menor en su grado mínimo a medio.

Artículo 3°

El que maliciosamente altere, dañe o destruya los datos contenidos en un sistema de tratamiento de información, será castigado con presidio menor en su grado medio.

Artículo 4°

El que maliciosamente revele o difunda los datos contenidos en un sistema de información, sufrirá la pena de presidio menor en su grado medio. Si quien incurre en estas conductas es el responsable del sistema de información, la pena se aumentará en un grado.

# Diseño de alto nivel.

El diagrama de despliegue es un tipo de diagrama utilizado para modelar la disposición físico de los artefactos software en nodos en objetos de despliegue.

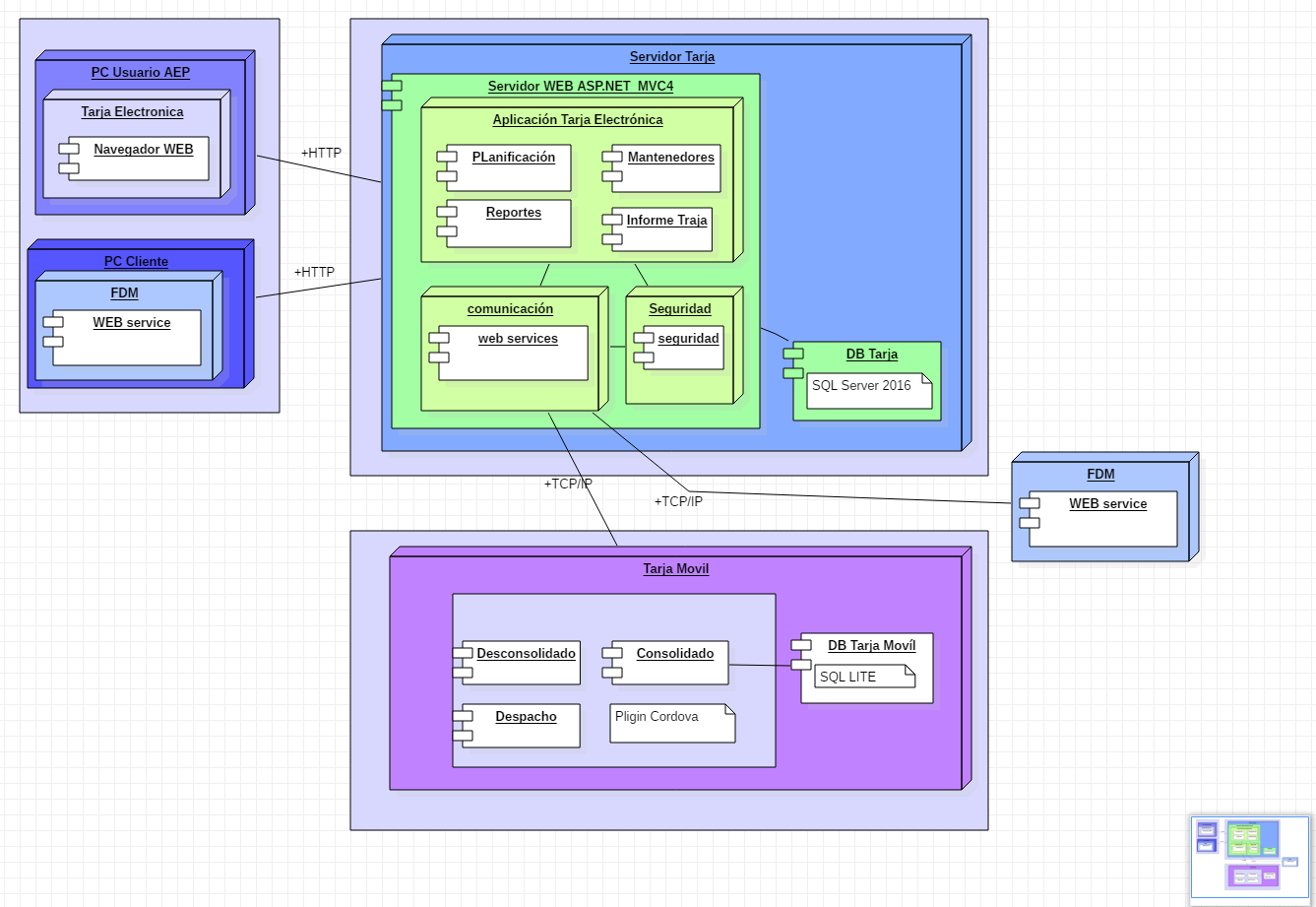


Figura 2.11. Diagrama de Alto Nivel

# Requerimientos de alto nivel.

# Requerimientos Sistema:

* RS1. Permitir acceder al sistema web desde internet aun cuando no este en la red corporativa.
* RS3. Permitir el uso del sistema móvil aun cuando no esté conectado a la red WIFI esta debe ser asincrónica.
* RS4. Múltiples usuarios pueden realizar diversas operaciones en una misma instancia de tiempo.
* RS5. Permitir modificación datos Ingresado.
* RS6. Permitir eliminar datos Ingresado.
* RS7. Permitir la realización de búsquedas de datos.
* RS8. El sistema deberá operar en todo momento excepto cuando se encuentre en mantención.
* RS9. El sistema debe permitir sacar fotografías

# Requerimientos Funcionales:

* RF1. El sistema debe solicitar contraseña de ingreso para acceder al sistema.
* RF2. El sistema debe manejar perfiles de usuarios para limitar acceso entre los distintos tipos de usuarios.
* RF3. El sistema debe usar validaciones para el correcto ingreso de la información antes de ser almacenada en la base de datos.
* RF4. El sistema de manejar mantenedores.
* RF5. El usuario debe poder acceder a una aplicación en terreno para poder hacer el ingreso de los datos.
* RF6. El sistema debe manejar distintos estados, para el correcto seguimiento.

# Requerimientos No Funcionales:

* RNF1. Escalabilidad: El sistema debe poder ser modificado para agregar nuevas funciones si se requiere a futuro por la empresa modificando el código de fuente.
* RNF2. Accesibilidad: El sistema no debe ser complejo de utilizar por los distintos tipos usuarios.
* RNF3. Disponibilidad: El sistema debe estar operativo y funcional cada vez que un usuario lo disponga.
* RNF4. Seguridad: El sistema debe respaldar los datos de manera confiable en la base de datos.
* RNF5. Rendimiento: El sistema debe ser rápido al procesar las órdenes que se le indiquen y obtener una rápida respuesta ante las instrucciones asignadas.
* RNF6. Portabilidad: El sistema debe tener la capacidad para ser instalado y desinstalado de forma exitosa en un entorno determinado.

Capítulo 3: Planificación del proyecto

# Planificación del proyecto.

# Metodología de desarrollo.

El modelo incremental combina elementos del modelo en cascada con la filosofía interactiva de construcción de prototipos. Se basa en la filosofía de construir incrementando las funcionalidades del programa. Este modelo aplica secuencias lineales de forma escalonada mientras progresa el tiempo en el calendario. Cada secuencia lineal produce un incremento del software.

Información: Ingeniería del Software: Un enfoque Práctico, Pressman Roger S. 2005.

Diseñado por Marlady Ortiz

Cuando se utiliza un modelo incremental, el primer incremento es a menudo un producto esencial, sólo con los requisitos básicos. Este modelo se centra en la entrega de un producto operativo con cada incremento. Los primeros incrementos son versiones incompletas del producto final, pero proporcionan al usuario la funcionalidad que precisa y también una plataforma para la evaluación.

# Ventajas

Entre las ventajas que puede proporcionar un modelo de este tipo encontramos las siguientes:

Mediante este modelo se genera software operativo de forma rápida y en etapas tempranas del ciclo de vida del software.

Es un modelo más flexible, por lo que se reduce el coste en el cambio de alcance y requisitos.

Es más fácil probar y depurar en una iteración más pequeña.

Es más fácil gestionar riesgos.

Cada iteración es un hito gestionado fácilmente

# Inconvenientes

Para el uso de este modelo se requiere una experiencia importante para definir los incrementos y distribuir en ellos las tareas de forma proporcionada. Entre los inconvenientes que aparecen en el uso de este modelo podemos destacar los siguientes:

Cada fase de una iteración es rígida y no se superponen con otras.

Pueden surgir problemas referidos a la arquitectura del sistema porque no todos los requisitos se han reunido, ya que se supone que todos ellos se han definido al inicio

# Roles dentro del proyecto

Product Owner: Profesor Patricio Castillo, quién estará a cargo de las siguientes funciones dentro del proyecto:

* Enviar material necesario para la realización del proyecto.
* Aconsejar y evaluar durante el avance del proyecto.
* Priorizar las tareas.
* Aceptar o rechazar resultados del trabajo.

Master: Alejandro Adam Concha, la cual velará que durante la realización del proyecto se sigan los valores y principios de la metodología ágil y guiar al equipo para encontrar la solución a la que apunta el proyecto.

Equipo: conformados por dos programadores de SAAM y Alejandro Adam quien es el encargado y el responsable del trabajo a realizar en cada iteración del proyecto.

# Planificación del proyecto.

# Product Backlog

A continuación, se presenta las historias de usuario que permitirá confeccionar el product backlog para el proyecto



Figura 3.1.4.1 Planificación Carta Gantt

# Product Backlog

Según Las historias de usuario descritas en el product backlog, se realizará el desglose de tareas que corresponderán a cada iteración realizar

# Planificación del proyecto

A continuación, se presenta la planificación del proyecto desde el punto de vista de la duración de los Sprint.

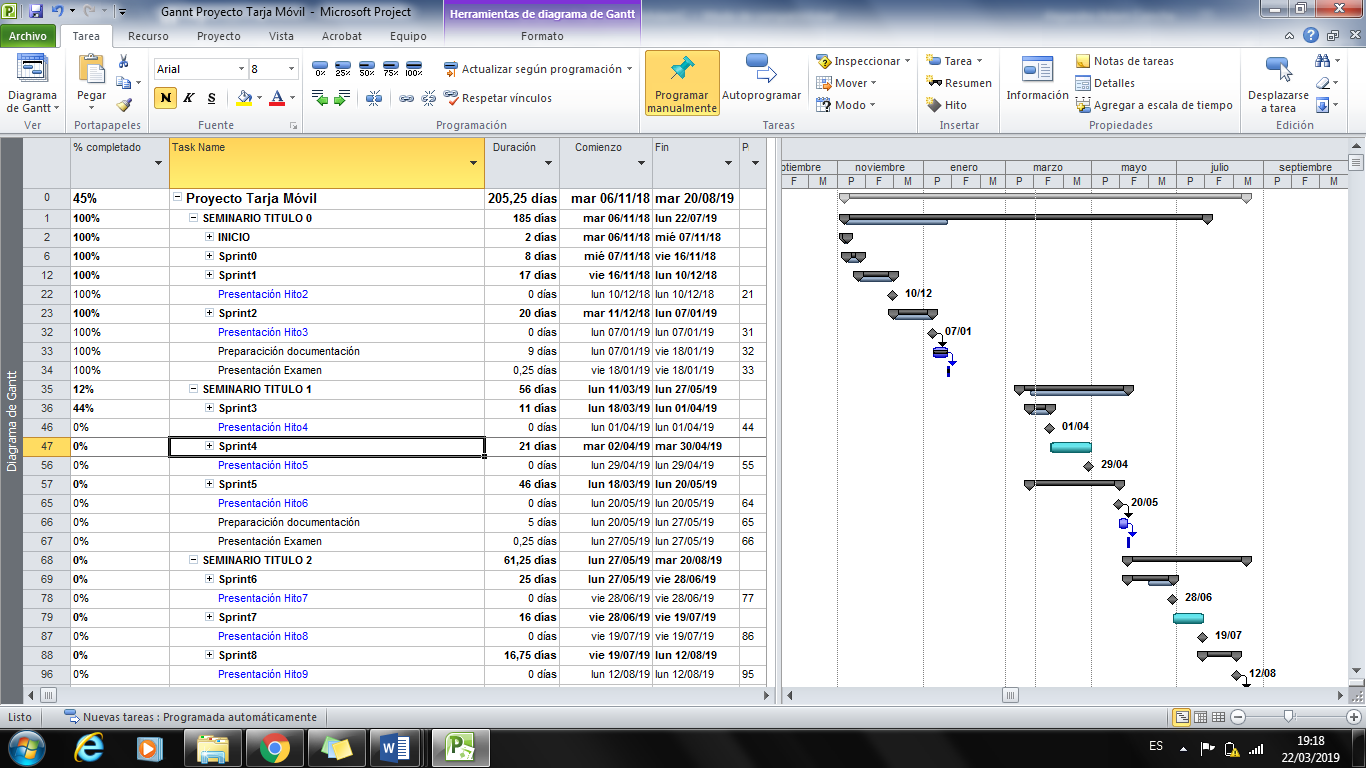


Figura 3.1.4.2 Planificación Carta Gantt

# Plan general de Pruebas

El plan de pruebas se encuentran en los anexos del proyecto.

# Responsables de las Pruebas

Las pruebas serán realizadas por usuario asignado por SAAM Extraportuarios y por Product Owner: Profesor Patricio Castillo quien validara las funcionalidades.

# Entorno de Pruebas

# Ambiente:

Las pruebas de se desarrollará de la siguiente manera:

1. En dependencia de SAAM Extraportuarios donde se realiza la faena tarja. Por personal SAAM.
2. Desde alojamiento externo donde se realizarán las pruebas por parte de nuestro Product Ower.

# Hardware:

Servidor Tarja: Servidor de propiedad de SAAM, en donde se aloja la base de datos SQL Server 2016 y el ambiente web en servidor de aplicaciones

# Software

**Pc escritorio.**

* Windows 7 o superior
* Código fuente de la aplicación, con editor de texto para las aplicaciones
* Datos de pruebas según especificación de etapa de testing.

**Equipo Movil.**

* Android 8.0
* APK instalado en el equipo.

# Entorno de desarrollo

Se desarrolla en las dependencias de SAAM S.A. departamento de desarrollo, donde se proporcionarán los equipos y servicios, al igual que los ambientes respectivos bajo normas de seguridad.

# Gestión de la configuración.

El control de cambios (Change Control, CC) es la columna vertebral. En términos generales, es un proceso sistemático para evaluar, coordinar y decidir sobre los cambios propuestos, así como también, para monitorear la implantación e incorporación de aquellas modificaciones aprobadas a los baselines y la documentación asociada. El CC asegura que los cambios sean propuestos, justificados, evaluados, coordinados, aprobados o rechazados, documentados e incorporados a un nuevo baseline.

De acuerdo a esto es necesario contar con un proceso que garantice que solamente los cambios aprobados sean implementados en un baseline. Para ello el CC se componen de cuatro actividades principales: proponer cambios, evaluar los cambios propuestos, aprobar/rechazar los cambios propuestos, e implementar los cambios aprobados.

El proceso de CC es aplicable a todas las etapas del ciclo de vida del software. Sin embargo, tiene sentido sólo a partir del establecimiento formal de la identificación de la configuración.

# Etapas

A continuación, se describen las etapas del proceso de control de cambios en términos de sus objetivos, criterios de entrada/salida y de sus actividades. Además, se enuncian los participantes de cada etapa y sus responsabilidades durante ella.

**Proponer cambios**.

**Objetivo**

El objetivo de esta etapa es identificar los problemas en la configuración, informarlos y dar inicio al proceso de CC.

Participantes

Desarrollador, Stakeholders.

Criterios de entrada

1. Identificación e informe de un problema en la configuración.

Actividades

Identificación de un problema.

El proceso de CC comienza con la detección de un problema. Inicialmente, el problema debe ser analizado informalmente por uno o más desarrolladores para establecer las causas y determinar posibles acciones correctivas.

Realizado lo anterior, quien detectó el problema debe informar mediante una Petición de cambios dirigida al responsable de la gestión de Cambio

Observación: La detección de problemas, generalmente, ocurre durante las revisiones, auditorías, pruebas o en el transcurso de la operación del producto.

Proponer cambios.

El responsable recibe la petición de cambio, revisando en primera instancia su claridad y la validez del problema. Si determina que es incompleta, la devolverá a su origen. En caso contrario, le asignará un identificador único, con el propósito de monitorearla, y registrará su información en un archivo (manual o electrónico) o base de datos para el monitoreo de las peticiones de cambio.

La petición de cambios debe satisfacer como mínimo las siguientes preguntas:

* ¿La solución es clara, es decir, la solución propuesta puede ser implementada por alguien ajeno al sistema?
* ¿La solución es consistente, no introduce conflictos con otros CI?
* ¿La solución asegura resolver los problemas detectados?
* ¿La solución es completa?
* ¿Se identifican los costos y la calendarización requerida para establecer las acciones correctivas?

Criterios de salida

1. La petición de cambio es aprobada en primera instancia por el responsable de SCM.
2. Evaluar los cambios propuestos

Objetivo

El propósito de esta etapa es hacer una evaluación de los cambios solicitados para verificar la validez de la petición de cambio.

Participantes

Representante de SCM, desarrolladores

Criterios de entrada

1. La petición de cambio es aprobada en primera instancia por el responsable de SCM.

Actividades

1. Distribución de la petición de cambio entre un subconjunto de los desarrolladores.

El responsable de SCM al interior del proyecto debe distribuir la petición de cambio entre un conjunto de desarrolladores para una segunda evaluación.

Estos desarrolladores, preferentemente, deben ser personas con una amplia visión del proyecto, pues ello da mayor objetividad y globalidad a su evaluación.

1. Los desarrolladores analizan la petición de cambios.

Los desarrolladores deben analizar las modificaciones propuestas en términos de su impacto sobre los requerimientos, la funcionalidad, interfaz, utilidad, costos y planificación del sistema, y también, sobre la confiabilidad, mantenibilidad, transportabilidad y eficiencia del software.

Los productos de este análisis son la descripción de los cambios por realizar para implementar la petición de cambios, los CI’s y la documentación afectada, y los recursos necesarios.

3. El responsable establece el paquete de cambios.

Una vez que el responsable recibe la información del análisis realizado por los desarrolladores, conforma el paquete de cambios. Este contiene a la petición de cambios y los resultados de la evaluación realizada por los desarrolladores durante la presente etapa.

Criterios de salida

1. Se estableció y completó el paquete de cambios.

Aprobar y/o rechazar los cambios

Objetivo

El objetivo de esta etapa es dar una resolución a las peticiones de cambio.

Participantes

CCB, representante de SCM

Criterios de entrada

1. El paquete de cambios se encuentra disponible.

Actividades

1. El representante de SCM, al interior del proyecto, distribuye entre los miembros del CCB el paquete de cambios.

El CCB es el responsable de tomar la decisión final sobre la petición de cambios. Por ello el representante de SCM debe entregar a sus miembros el paquete de cambio con la debida anticipación a la reunión de evaluación.

2. El CCB se reúne para resolver la aprobación o rechazo de la petición de cambio.

Básicamente, el CCB se reúne para decidir, sobre la base de la información contenida en el paquete de cambios, la aprobación o rechazo de la petición. Su respuesta puede ser la aprobación, rechazo o una solicitud de mayor información y/o de un análisis adicional.

Si se autoriza la petición, ésta es enviada al CMO para dar curso a las acciones respectivas. De ser denegada, es devuelta a su origen junto a las razones expuestas por el CCB. Y en la última circunstancia, se envía el paquete de cambios al grupo de desarrolladores, que estuvo a cargo de su análisis, junto a las consultas del CCB.

Criterios de salida

1. El CCB a aprobado o rechazado la petición.

En el caso de aprobación se pasa a la siguiente etapa. En el caso contrario el proceso se cierra.

### Implementar los cambios

Objetivo

El propósito de esta etapa es verificar la implementación de las acciones correctivas aprobadas pro el CCB.

Participantes

Representante de SCM, desarrolladores, bibliotecario, grupo de SQA.

Criterios de entrada

1. La petición de cambio fue aprobada por el CCB.

Actividades

1. Notificación del cambio.

Una vez que el cambio ha sido autorizado, se pone en marcha su implantación mediante una notificación de cambio que detalla los cambios por realizar, las restricciones y los criterios para su posterior revisión y auditoría. Esto debe ser llevado a cabo por el responsable de SCM.

Una copia de la notificación es enviada al bibliotecario y otra a los desarrolladores a los cuales se les asignó la tare de implementar las acciones correctivas definidas en la petición de cambio.

2. Autorización de acceso a los CI de las baselines establecidas.

Al despacharse la notificación, se autoriza el acceso a los CIs contenidos en la biblioteca del software a los desarrolladores responsables de realizar los cambios aprobados. Los cambios son hechos sobre una copia del CI.

Es responsabilidad del bibliotecario habilitar correctamente dicho acceso.

3. Revisión y aprobación de los cambios realizados.

Cuando los cambios hayan sido completados, debe llevarse a cabo una revisión sobre el nuevo CI. Esta revisión y el monitoreo de la adecuada implementación de los cambios es responsabilidad de SQA.

Cuando las actividades de SQA sobre los CIs afectados hayan concluido, se originará una nueva versión de los CIs y estos volverán a la biblioteca. Por último, la nueva versión será incluida en la baseline correspondiente y distribuida en la organización.

Criterios de salida

1. Los cambios han sido completados y aprobados.
2. Una nueva versión de los CIs afectados a ha sido incorporada a la biblioteca del software

## Participantes

Definición de los roles y responsabilidades de los participantes del proceso de control de cambios.

**Representante de SCM**

La unidad de SCM debe nombrar a uno de sus miembros como el responsable del proceso de control de cambios para cada proyecto. Este representante de SCM al interior del proyecto deberá responsabilizarse de:

* Establecer y documentar el proceso de control de cambios.
* Gestionar y coordinar el proceso de control de cambios.
* Mantener registros de las actividades del proceso.
* Preparar y facilitar la evaluación de la petición de cambio.
* Interactuar con el CCB para la resolución de la petición de cambio.
* Asegurar el contenido de la biblioteca del software, al informar al bibliotecario sobre el estado de las actividades del control de cambios.
* Garantizar la correcta implementación de los cambios aprobados.
* Mantener informados al jefe de proyectos, a los desarrolladores y a la unidad de SCM sobre el estado de los cambios.

**Desarrolladores**

El término desarrollador es utilizado para referenciar a cualquier miembro de un proyecto. En relación con el proceso de control de cambio, cada desarrollador debe:

Informar sobre los problemas detectados, mediante la petición da cambio, al representante de SCM.

* Definir en la petición las causas y posibles soluciones al problema detectado.
* Facilitar cualquier información que se requiera para documentar o explicar el problema detectado.
* Participar en la implementación de los cambios aprobados que le hayan sido asignados.
* Coordinar la correcta incorporación de los cambios a los CIs con el representante de SCM.

### CCB

El comité de control de la configuración (Configuration Control Board, CCB) tiene bajo su responsabilidad la revisión de las peticiones de cambio a los componentes del software y la aprobación de éstas.

### Unidad de SQA

Dentro de las actividades de SQA se encuentra el monitoreo de las actividades de SCM, por ello un representante de SQA tiene que:

* Verificar el correcto establecimiento del proceso de control de cambio.
* Corroborar la adherencia del proceso establecido con el proceso y los estándares definidos.
* Revisar la implementación de los cambios aprobados, antes de que los CIs afectados por ellos ingresen a la biblioteca del software.
* Participar como miembro del CCB de ser necesario.

**Jefe de proyecto**

A pesar de no ser miembro activo del proceso de control de cambio, el jefe de proyecto debe facilitar sus actividades, difundirlas entre los desarrolladores y, de ser necesario, debe participar como miembro del CCB.

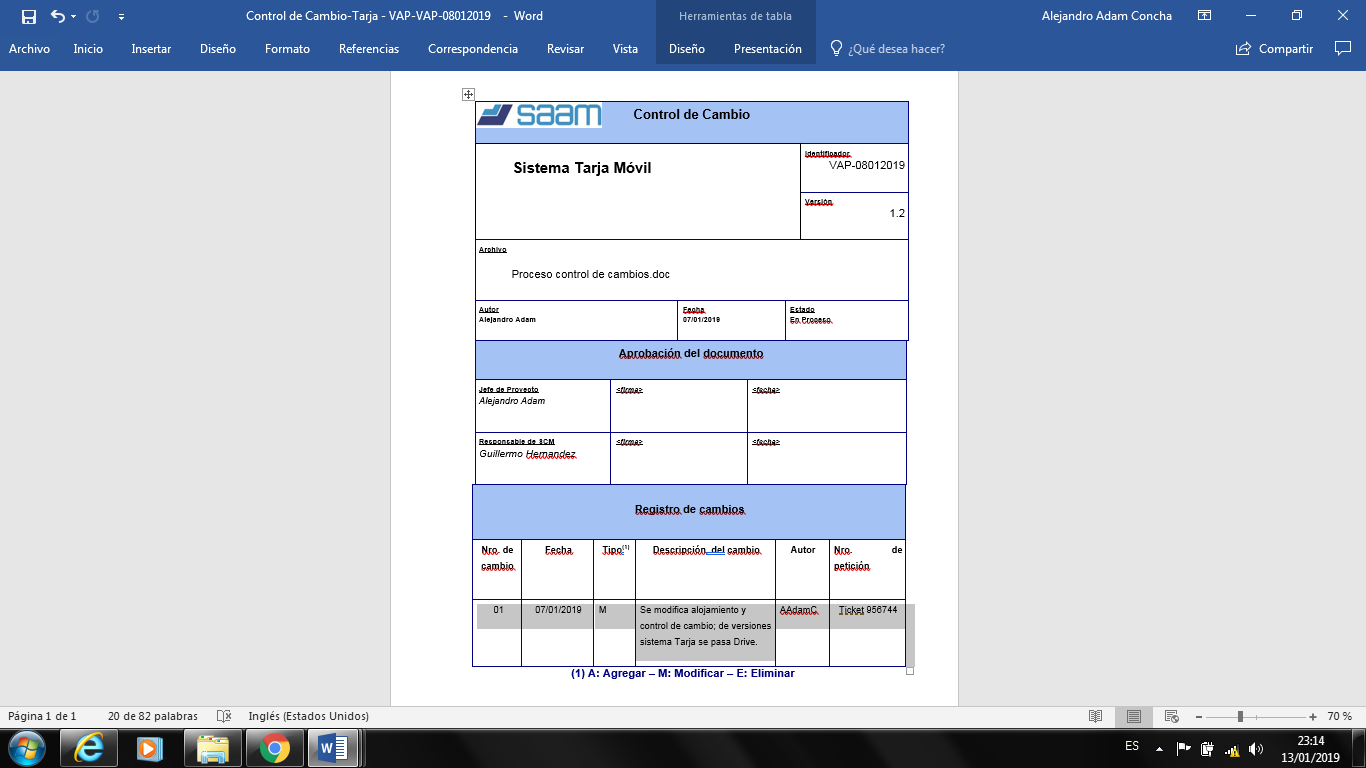


Figura 3.2 Control de Cambio

# Gestión de cambios.

Para el control de cambios se utilizará el almacenaje en GitHab

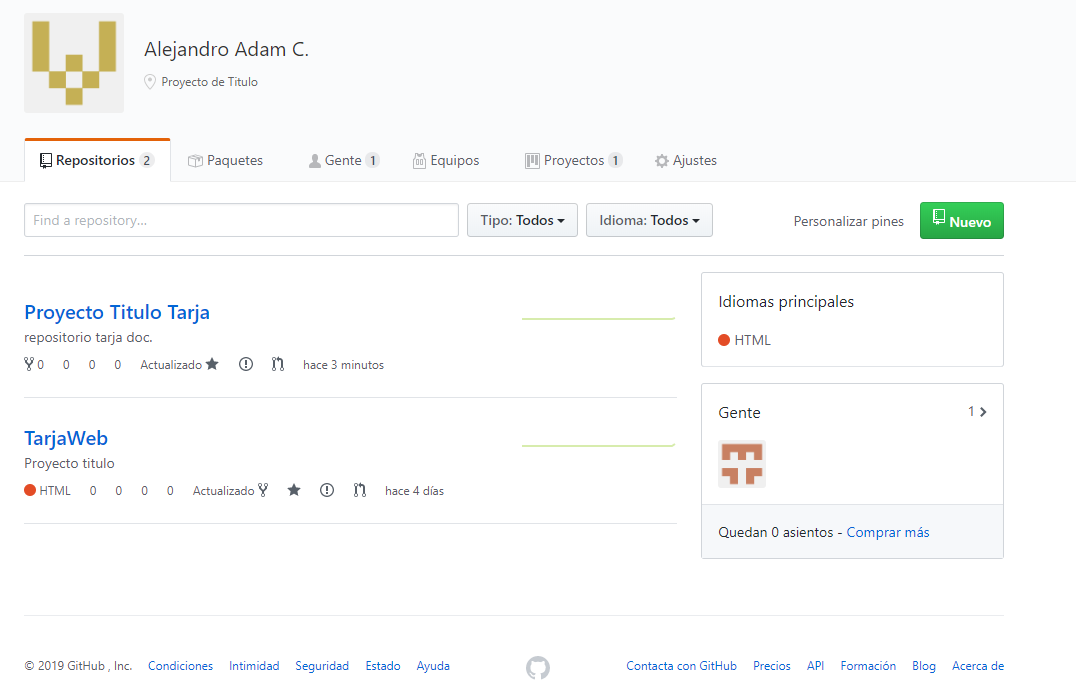


Figura 3.3Control del Cambio

# Gestión de riesgos.

Para la valoración de riesgos se utilizará un análisis cuantitativo mediante el método de Joan Peib, se verá reflejado en la siguiente figura, donde se representan los niveles de acuerdo con la probabilidad e impacto de cada riesgo.

En el análisis de riesgo buscamos cualquier situación adversa que pueda llegar a afectar al curso normal del proyecto asignándole una probabilidad de ocurrencia y un plan de mitigación asociada, siendo necesario en algunos casos en los que la probabilidad y el grado en el que afecta al proyecto sean demasiado altos la generación de planes de contingencia para minimizar los efectos sobre el mismo.

Para administrar los riesgos asociados al proyecto, se efectuarán las siguientes actividades:

* Identificación de los riesgos, identificar los riesgos de proyecto, negocio y producto.
* Análisis de riesgos, evaluación de la probabilidad y consecuencia de los riesgos.
* Planificación de riesgos, elaboración de planes para minimizar o evitar los efectos del riesgo.
* Monitoreo de riesgos, monitorear los riesgos durante todo el proyecto.

Cada riesgo conlleva un plan de mitigación asociado el cual permite minimizar la ocurrencia de dicho riesgo, sin embargo, los planes de contingencia serán solo realizados para aquellos riesgos que tengan un alto grado de ocurrencia y que además tengan una incidencia en el proyecto que puedan afectar en forma seria.

Para la evaluación del riesgo nos basaremos en la siguiente tabla de riesgo

Donde:

Probabilidad de ocurrencia: 1-bajo, 2-medio, 3-alto

Impacto o Severidad del Riesgo: 1-bajo, 2-medio, 3-alto

Exposición al Riesgo: Probabilidad de Ocurrencia versus el Impacto

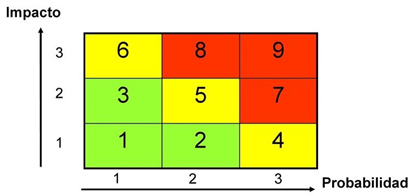


Figura 3.4.1 Tabla valorización de riesgo



Figura 3.4.2 Tabla de riesgo

# Entorno de desarrollo.

**Herramientas de desarrollo   
Lenguaje de programación.**  
A lo largo del proyecto y debido a los distintos módulos que se espera implementar, se utilizarán distintos lenguajes de programación que se especificarán a continuación:

**C#**

Es un lenguaje de programación orientado a objetos desarrollado y estandarizado por Microsoft como parte de su plataforma .NET.  
Su sintaxis básica deriva de C/C++ y utiliza el modelo de objetos de la plataforma.NET el cual es similar al de Java aunque incluye mejoras derivadas de otros lenguajes (entre ellos Delphi).

Este lenguaje será utilizado para desarrollar la aplicación móvil, junto con un grupo de otros lenguajes que contiene el Compact Framework de .Net.

**Html**  
Siglas de HyperText Markup Language (Lenguaje de Marcado de Hipertexto), es el lenguaje de marcado predominante para la elaboración de páginas web. Es usado para describir la estructura y el contenido en forma de texto, así como para complementar el texto con objetos tales como imágenes. HTML se escribe en forma de "etiquetas", rodeadas por corchetes angulares (<,>). HTML también puede describir, hasta un cierto punto, la apariencia de un documento, y puede incluir un script (por ejemplo Javascript), el cual puede afectar el comportamiento de navegadores web y otros procesadores de HTML.

**JavaScript**  
Es un lenguaje de scripting basado en objetos no tipeado y liviano, utilizado para acceder a objetos en aplicaciones. Principalmente, se utiliza integrado en un navegador web permitiendo el desarrollo de interfaces de usuario mejoradas y páginas web dinámicas.  
JavaScript es un dialecto de ECMAScript y se caracteriza por ser un lenguaje basado en prototipos, con entrada dinámica y con funciones de primera clase. JavaScript ha tenido influencia de múltiples lenguajes y se diseñó con una sintaxis similar al lenguaje de programación Java, aunque más fácil de utilizar para personas que no programan.

**Css (Hojas de estilo en cascada)**  
CSS es un lenguaje usado para definir la presentación de un documento estructurado escrito en HTML o XML (y por extensión en XHTML). El W3C (World Wide Web Consortium) es el encargado de formular la especificación de las hojas de estilo que servirán de estándar para los agentes de usuario o navegadores.

**Motor de base de Datos SQL Server 2012 o superior.**  
Todos sabemos que hay muchos datos en el mundo y crecen de manera exponencial, por eso las bases de datos de SQL suelen ser la "columna vertebral" de la arquitectura TI de la organización.  
Muchas organizaciones confían en SQL Server para aprovechar sus datos al máximo sin tener en cuenta dónde están o cómo desean usarlos, y la plataforma proporciona un máximo valor en el sector por una pequeña parte del precio de sus competidores.

Características y ventajas.   
SQL Server 2016 proporciona espectaculares funciones para tareas críticas con flexibilidad, rendimiento y disponibilidad para su OLTP más importante y las cargas de trabajo de almacenamiento de datos.

Ampliable hasta 12TB de memoria y 640 procesadores lógicos con Windows Server 2016

Consigue transacciones hasta 30 veces y consultas hasta 100 veces más rápidas con rendimiento mejorado en la memoria.

Realiza Análisis operacionales en tiempo real sobre datos de transacciones

Equilibra las cargas a través de auxiliares legibles en grupos Siempre disponibles.

Seguridad y rendimiento.  
El acercamiento multicapa a la seguridad tiene un historial comprobado de proporcionar las bases de datos menos vulnerables, a pesar de ser la base de datos del mundo más utilizada.

Confíe en la base de datos menos vulnerable entre las plataformas importantes durante seis años seguidos

Proteja los datos almacenados y en movimiento con TDE y el nuevo AlwaysEncrypted

Enmascare los datos sensibles con un impacto de aplicación mínimo utilizando Enmascaramiento de datos dinámico

Garantice el acceso basado en las características del usuario con Seguridad de nivel de fila Inteligencia Empresarial completa SQL Server 2016 proporciona una plataforma BI completa, in situ y preparada para la empresa que le ayuda a transformar datos complejos en conocimientos prácticos.

Cree informes modernos y visualice datos densos con tipos de gráficos adicionales Acceda a los KPI y a informes móviles y paginados utilizando el portal web de Servicios de informes de SQL Server

Consiga un rendimiento más rápido de los Servicios de Análisis de SQL Server con procesamiento paralelo.

Utilice modelos multidimensionales mejorados en los Servicios de análisis de SQL Server.

Configure fácilmente la vista previa de las Herramientas de datos de SQL Server en Visual Studio 2015

Análisis avanzado en la base de datos.  
Los análisis avanzados integrados proporcionan escalabilidad y rendimiento para construir y ejecutar los algoritmos de análisis avanzados directamente en el núcleo de la base de datos transaccional de SQL Server.

Transforme datos complejos procedentes de diversas fuentes en modelos de datos de confianza utilizando el lenguaje de modelado estadístico más popular

Procese análisis in situ, reduciendo las latencias y los costes operacionales

Cree modelos una vez e implementarlos en cualquier lugar: en bases de datos, para la nube o para Linux, Hadoop y Teradata.

Acceda a miles de Scripts R y Modelos en CRAN (Common R Archival Network)

Experiencia consistente tanto In situ como en la nube SQL Server 2016 proporciona una experiencia consistente in situ y en la nube. Obtendrá una experiencia excepcional si los datos están en su centro de datos, en una nube privada o en Azure.

Ceda dinámicamente datos templados y fríos a Azure con Stretch Database

Afronte sus cargas de trabajo de tareas críticas con tamaños de máquinas virtuales de Azure más grandes

Confíe en nuestras características cloud-first, probadas por millones de bases de datos de Azure

Utilice donde quiera los conocimientos que ya tiene, con las herramientas de gestión y desarrollo comunes y con T-SQL.

**Microsoft Visual Studio**   
Es un entorno de desarrollo integrado para sistemas operativos Windows.

Soporta varios lenguajes de programación tales como Visual C++, Visual C#, Visual J#, ASP.NET y Visual Basic .NET, aunque actualmente se han desarrollado las extensiones necesarias para muchos otros.

Visual Studio permite a los desarrolladores crear aplicaciones de escritorio, sitios y sistemas Web, así como servicios web en cualquier entorno que soporte la plataforma .NET.

Así se pueden crear aplicaciones que se intercomunican entre estaciones de trabajo, páginas Web y dispositivos móviles. Para el proyecto, específicamente en la etapa de desarrollo de la aplicación móvil, se ocupará el .NET Compact Framework que está diseñado para ofrecer un rendimiento óptimo bajo las restricciones de los limitados recursos de los dispositivos móviles.

Capítulo 4: Resultados

# Resultados

En este capítulo se hablará sobre los resultados que hasta la fecha se han obtenido del proyecto, el margen de error de las estimaciones, la arquitectura y el diseño planificado inicial, versus el obtenido, la planificación real, versus la planeada entre otros aspectos.

En los anexos se presentan los resultados de los hitos y avances del proyecto.

# Arquitectura

Como arquitectura inicial se diseñó un diagrama de despliegue, véase la figura 2.11 En donde podemos ver de qué forma interactúan y se comunican los distintos componentes de la aplicación.

Los cambios realizados en el diagrama corresponden a incorporar la seguridad ligado a los controles de acceso con sus respectivos perfiles como un componente más dentro de las aplicaciones y la incorporación de servidor FDM para el almacenaje de las imágenes las cuales se comunica a través de un webservice. Por otro lado se elimina la integración con torpede TCE y con SAP. Para lo cual se ha modificado el diagrama quedando de la siguiente forma.

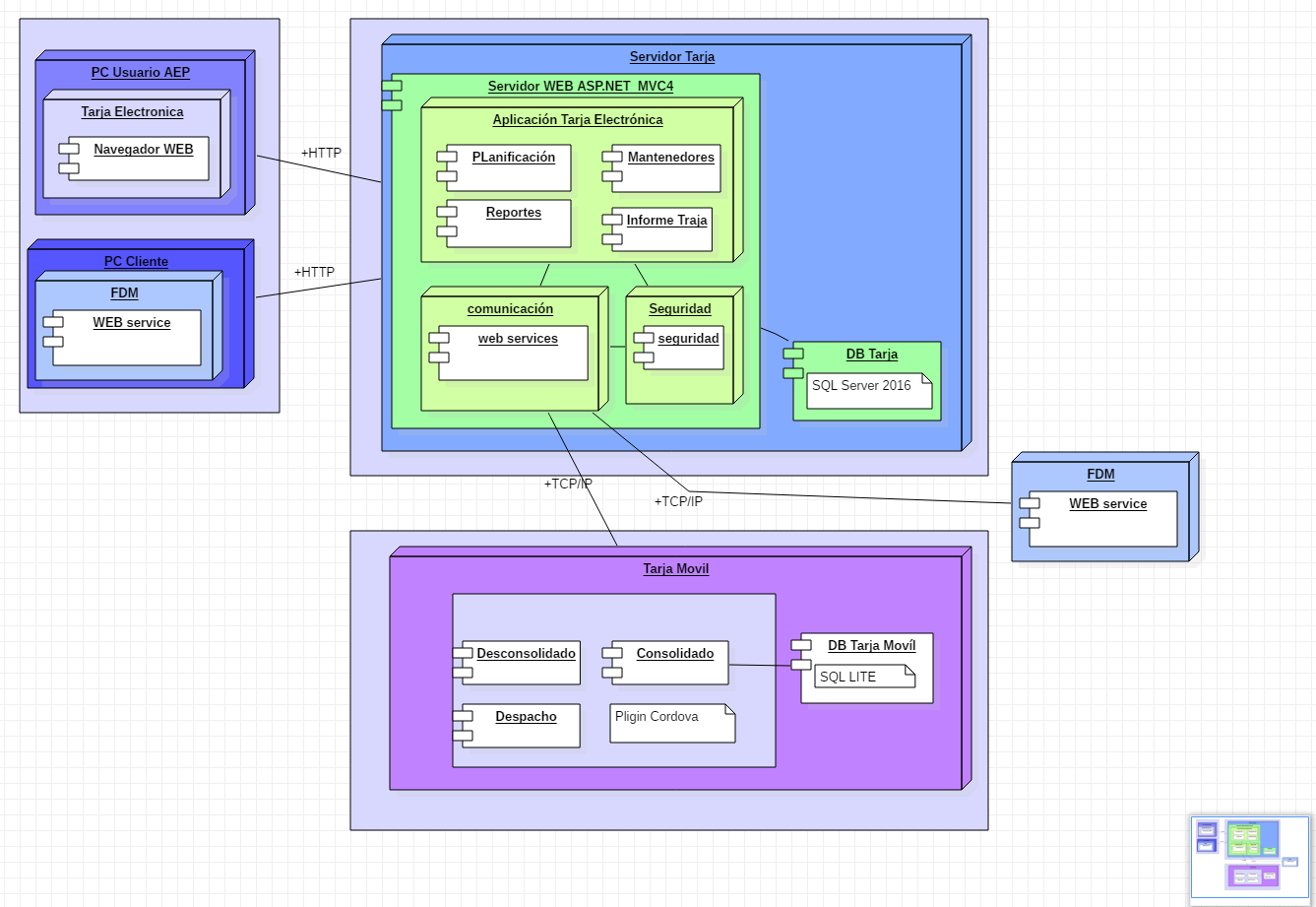


Figura 4.1 Diagrama de despliegue

# Diseño caso de uso

El diagrama de caso de uso nos permite modelar las vistas de casos de uso un sistema. Nos permite visualizar la forma en que los actores del sistema interactúan con la aplicación.

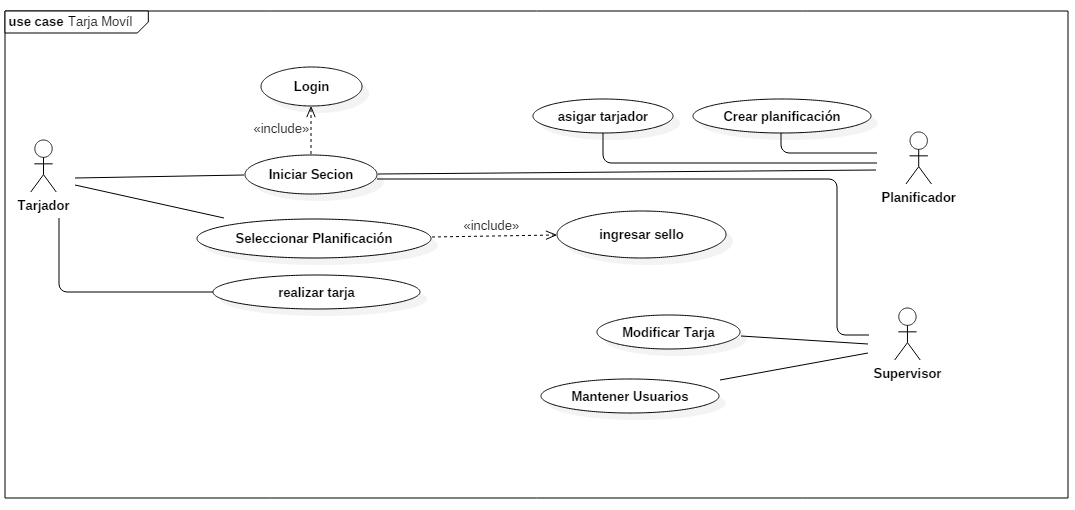


Figura 4.1 Diagrama de despliegue

# Diagrama de proceso.

Un diagrama de proceso es una representación gráfica de los pasos que se siguen en una secuencia de actividades. Para ello se ha desarrollado un diagrama para la actividad desconsolidado y consolidado.

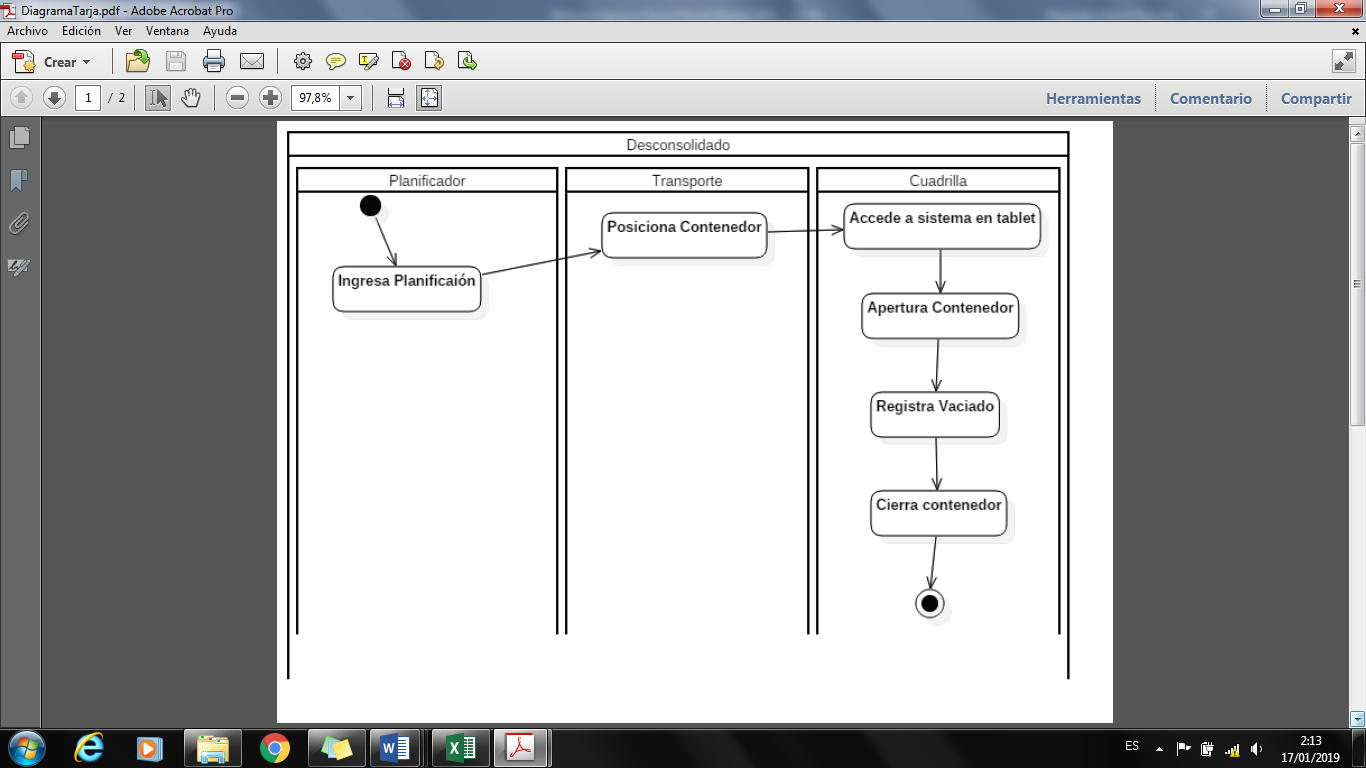


Figura 4.3.1 Diagrama de proceso desconsolidado

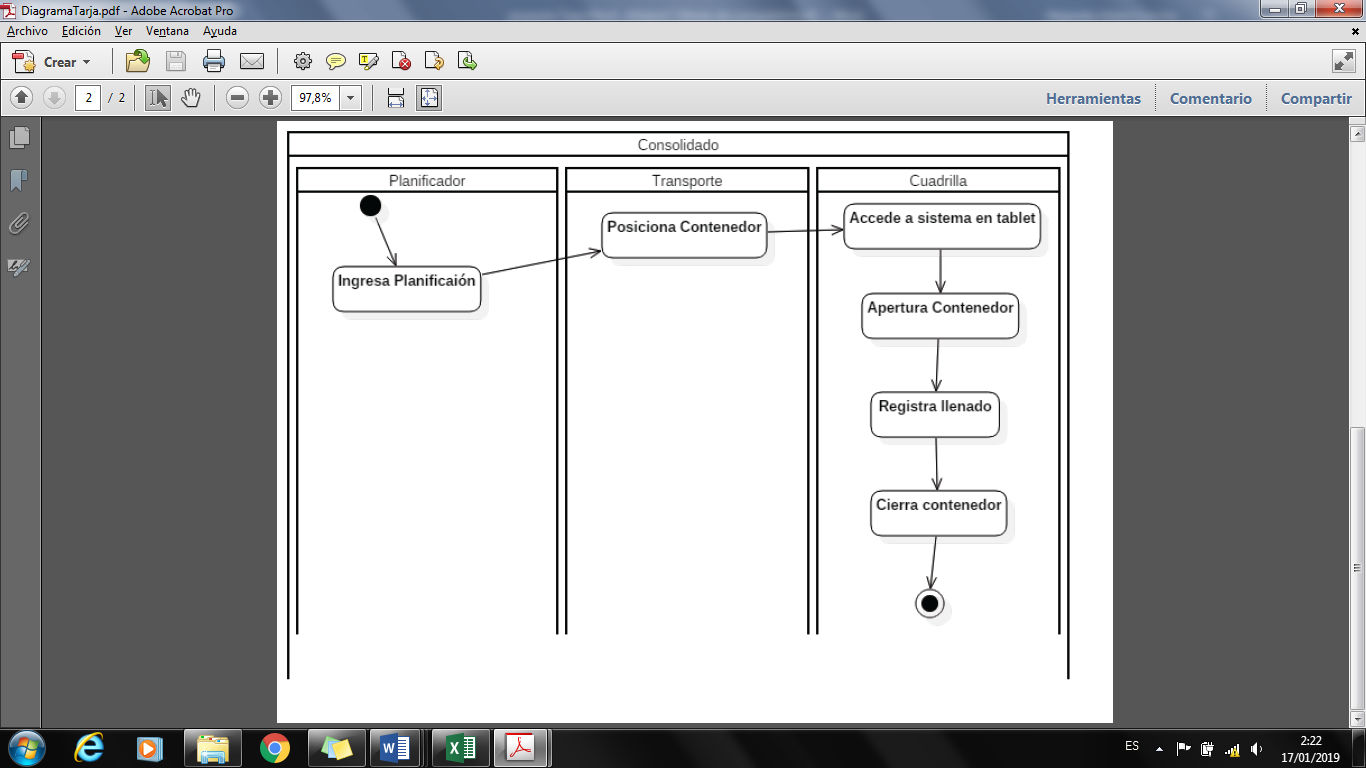


Figura 4.3.2 Diagrama de proceso Consolidado

# Diagrama de Clases

En ingeniería de software, un diagrama de clases en Lenguaje Unificado de Modelado (UML) es un tipo de diagrama de estructura estática que describe la estructura de un sistema mostrando las clases del sistema, sus atributos, operaciones (o métodos), y las relaciones entre los objetos.Imagen que contiene texto, mapa

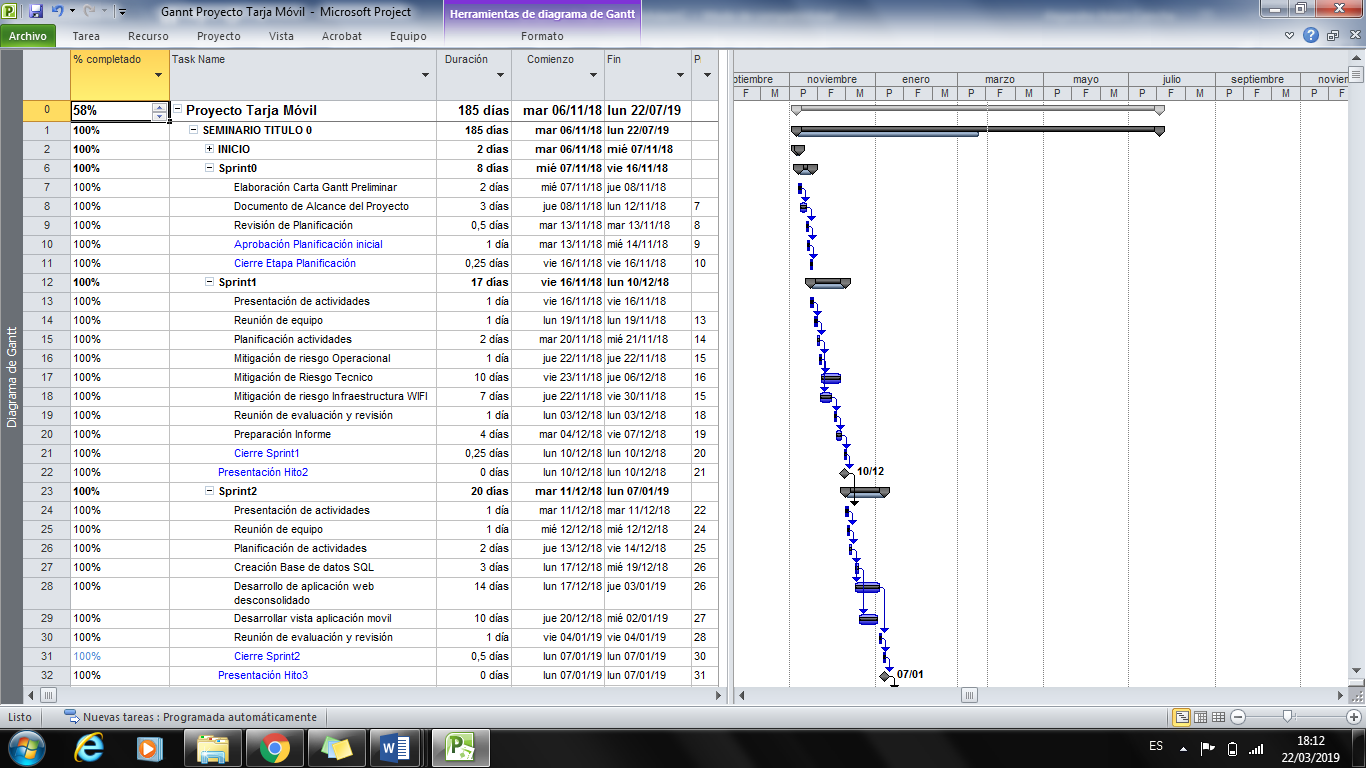
Descripción generada con confianza muy alta

# Avance del proyecto

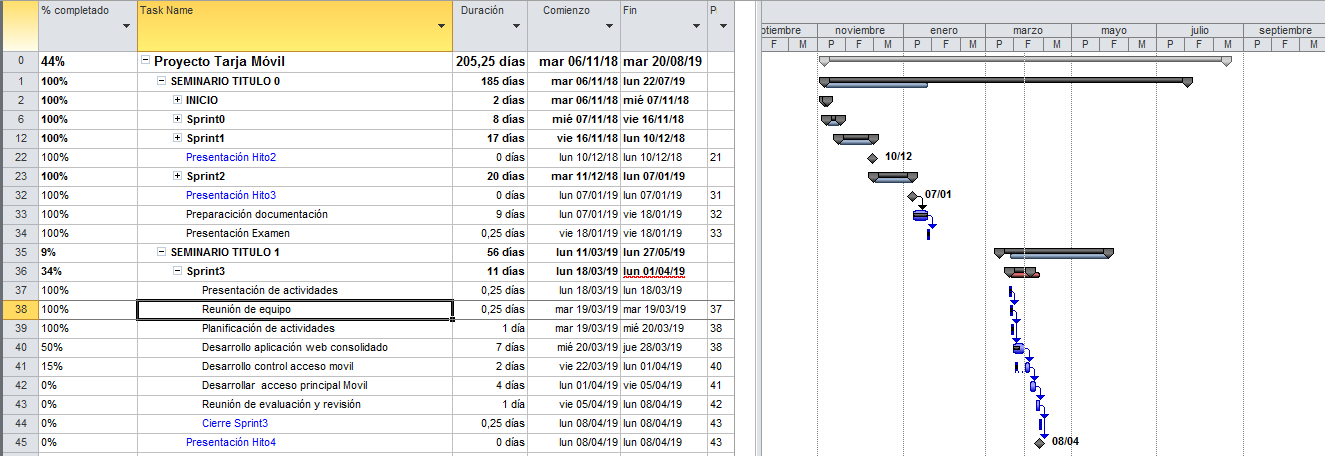
El proyecto se ha dividido en 3 etapas con entregas parciales por hitos.

I Etapa: proyecto titulo 0

A continuación, se presentas la Gantt desarrollada para esta etapa que contempla 3 hitos de entrega.



II Etapa: proyecto titulo 1



Capítulo 5: Conclusiones

# Conclusiones.

# Posmortem

**Problemas detectados**

1. Los problemas de infraestructuras detectados en un comienzo como es la falta de cobertura WIFI en los terminales, Hecho que es de vital si se está considerando contar con una aplicación móvil. Para un proceso que es crítico como es la tarja.
2. El factor tiempo y de experiencia en desarrollo en aplicaciones móviles, hace que sea un factor de riesgo dentro del proyecto.
3. La falta de compromiso por parte de los JOP hace que cueste avanzar con lo planificado.
4. La falta de experiencia dentro de la organización en el uso y trabajo bajo metodologías ágiles hace que se tenga que replantear.
5. Los cambios realizados en la seguridad informática en las últimas semanas, hace que se tenga que replantear actividades y mecanismos establecidos en el proyecto tales como, control de cambio, pruebas, control de versiones, etc.

**Acciones tomadas**

Dado los inconvenientes presentados, se debió tomar acciones con el fin de generar el menor impacto en resultado del proyecto.

1.- La falta de cobertura wifi en algunas zonas de los terminales, que se logró detectar mediante un estudio de cobertura realizado en forma oportuna, permite definir la forma en que debe operar el equipo móvil, siendo esta asincrónica.

2.- La falta de experiencia en proyectos con metodologías ágiles y el poco compromiso hace que se deba replantear. para ello, se establece cambios en la que inicialmente era SCRUM y se cambia por incremental. Para mejorar el factor compromiso se estable las reuniones en forma alternada en los terminales de esta manera se refuerza el compromiso.

3.- Los cambios generados en la seguridad informática, nos obliga a cambiar la forma de trabajar. Para ello, el control de versiones, cambios, fuentes, etc. se llevarán en paralelo usando el Drive de Google de esta manera se tendrá acceso desde fuera, de igual forma de monta un ambiente similar al de producción en un PC local de manera de poder realizar las pruebas y el avance del proyecto.

**Lecciones aprendidas**

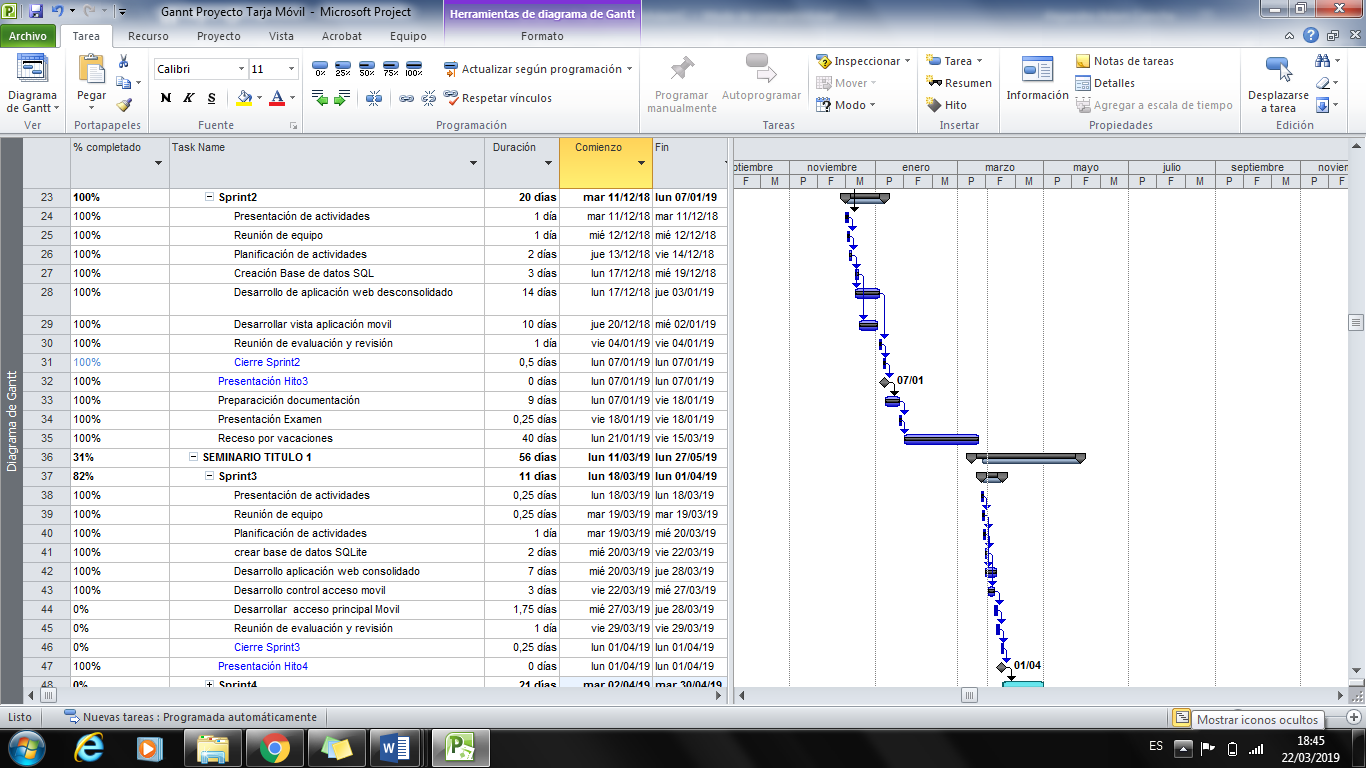
Se aprendió que es necesario tener presente todos los factores técnicos, culturales y de recursos existente en la organización y como estos pueden incidir en el proyecto. Este ejercicio, debe ser permanente durante el ciclo de vida del proyecto, incluso, antes del inicio de este.

# Problemas abiertos.

Dentro de los problemas que están en estado abierto es el cierre del hito 3 por parte de los Jefes de Operaciones dado que se espera que este terminada la solución web por completo. Es decir, revisada, probada y entregada se formalizará la recepción y cierre de etapa.

# Trabajo futuro.

Terminado el proyecto, se continuará con las mantenciones del sistema. Las modificaciones solicitadas por el negocio serán evaluadas por el jefe de proyecto asignado, donde se especificará el requerimiento generando un backlog donde se cuantificará el impacto se estimarán los tiempo, recursos y dificultad. Las de baja dificulta, bajo impacto, mínimo recurso se considerar en el servicio de mantención. Todas las demás se presentarán como nuevo desarrollo (proyecto)



Referencias

1. <http://www.asesoriaintegral.cl/web/temas/29/funcion-de-trabajador-portuario---procedimiento-de-tarja.html>
2. ttps://es.wikipedia.org/wiki/Arquitectura\_de\_software

<http://www.solucioneslogisticas3pl.com/consolidacion-y-desconsolidacion/>

<http://www.emb.cl/negociosglobales/articulo.mvc?xid=1511>

1. Ingeniería del Software: Un enfoque Práctico, Pressman Roger S. 2005.
2. OBS Business School Metodologías Agiles.