Instituto Universitario Aeronáutico



TRABAJO FINAL “Analista de Sistemas”

Empresa: EasyLocation s.r.l

Tutor: Ing. Brenda Meloni

Alumno: Alejandro Manuel Sosa Adorati

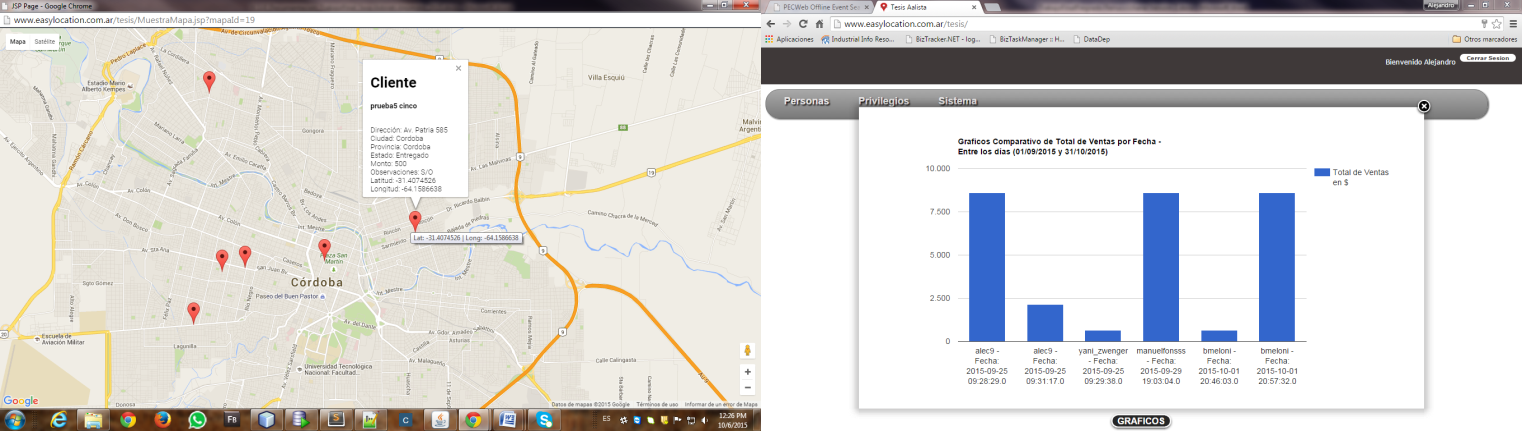


**INDICE**

**EL SISTEMA:**

****

**EASYLOCATION - SISTEMA ORIENTADO A LA REPRESENTACIÓN GEOGRAFICA DE CLIENTES Y ESTADISTICAS DE VENTAS Y REPARTOS.**



A. Introducción

Con la documentación que aquí precede, se dejara en claro todo el análisis correspondiente al sistema anteriormente mencionado y tiene como propósito llevar a cabo el desarrollo de un sistema de software con una metodología específica, realizando todas las etapas del proceso obteniendo como resultado su funcionamiento, probado y con valor agregado.

A.1 Motivación.

La idea original del presente trabajo corresponde exclusivamente a la observación del proceder de los ejecutivos en distintas empresas sobre la manipulación de los datos en sus respectivas bases de datos. En su mayoría los ejecutivos tienden a editar planillas de EXCEL para luego importar esos datos a la base local, de esta manera ahorran tiempo en cargar de a uno las altas, bajas o modificaciones de Datos, entonces simplemente se tomó esa idea de edición de archivos .xls para cargar en nuestro sistema, en donde, un listado de clientes exportados de una base existente puede convertirse en un mapa con tantos "GOOGLEMAP MARKERS" como clientes existan en el archivo .xls.

A.2 Objetivos Principales (Generales).

Este trabajo tiene como su primer objetivo la posibilidad de referenciar geográficamente, un listado especifico de clientes exportados de una base de datos existente en formato .xls. De esta manera se puede mapear un listado de direcciones en un único mapa, para distintas comparaciones y análisis detallados.

También como objetivo principal, es darle al usuario del sistema la posibilidad de generar gráficos por fechas partiendo de la información cargada a través de las direcciones en los archivos .xls, y toda esta información también estará disponible para ser exportada, siendo estas 3 funciones las importantes del sistema.

**A.3 Objetivos específicos**.

Dentro de los objetivos específicos encontraremos como principal, brindarle al cliente una sensación de seguridad respecto a los datos que se están guardando en la base de datos del sistema, esto el sistema debe lograrlo garantizando un acceso mediante "Usuario" y "Contraseña" y por supuesto con permisos otorgados previamente por el administrador del sistema. Además, al cliente se le debe brindar la "LATITUD" y "LONGITUD" resuelta de cada cliente en el listado originalmente exportado, esto es, que en el momento próximo después de haber cargado en la base de datos las direcciones y los datos pertinentes de cada cliente, se debe resolver los datos de posicionamiento global mediante la "API - GOOGLE MAPS" y guardarlos con su dirección correcta en la base del sistema. También, un objetivo particular, es la satisfacción total del cliente en la experiencia del uso del sistema, relacionado con la interfaz grafica, las funciones y los resultados esperados, entonces solo se podrá llegar a tan especifico objetivo si solo se cumplen los pasos pre impuestos del desarrollo y análisis del sistema.

A.4 Alcance.

El alcance de este sistema, pienso personalmente no tiene límites, en primera instancia puede ser usado por cualquier empresa o organización que cuente con un listado de clientes existente en su base de datos, obviamente siempre que cada cliente cuente con su dirección guardada correspondientemente. Pero, obviamente, el alcance empieza por estos clientes, pero se puede extender para referenciar cualquier listado de dirección, en la industria o en la administración pública, la posibilidad de importar un listado de direcciones y mostrarlas en un mismo mapa se extiende en un espectro muy grande y solo la imaginación nos limita en este caso.

**1.5 Marco de Desarrollo de Software (Metodología).**

El Proceso Unificado de Desarrollo Software o simplemente Proceso Unificado es un marco de desarrollo de software que se caracteriza por estar dirigido por casos de uso, centrado en la arquitectura y por ser iterativo e incremental.

**Iterativo e Incremental**: El Proceso Unificado es un marco de desarrollo iterativo e incremental compuesto de cuatro fases denominadas Inicio, Elaboración, Construcción y Transición. Cada una de estas fases es a su vez dividida en una serie de iteraciones (la de inicio sólo consta de varias iteraciones en proyectos grandes). Estas iteraciones ofrecen como resultado un incremento del producto desarrollado que añade o mejora las funcionalidades del sistema en desarrollo. Cada una de estas iteraciones se divide a su vez en una serie de disciplinas que recuerdan a las definidas en el ciclo de vida clásico o en cascada: Análisis de requisitos, Diseño, Implementación y Prueba. Aunque todas las iteraciones suelen incluir trabajo en casi todas las disciplinas, el grado de esfuerzo dentro de cada una de ellas varía a lo largo del proyecto .

**Dirigido por los casos de uso** En el Proceso Unificado los casos de uso se utilizan para capturar los requisitos funcionales y para definir los contenidos de las iteraciones. La idea es que cada iteración tome un conjunto de casos de uso o escenarios y desarrolle todo el camino a través de las distintas disciplinas: diseño, implementación, prueba, etc.

**Centrado en la arquitectura** El Proceso Unificado asume que no existe un modelo único que cubra todos los aspectos del sistema. Por dicho motivo existen múltiples modelos y vistas que definen la arquitectura de software de un sistema. La analogía con la construcción es clara, cuando construyes un edificio existen diversos planos que incluyen los distintos servicios del mismo: electricidad, fontanería, etc.

**Enfocado en los riesgos** El Proceso Unificado requiere que el equipo del proyecto se centre en identificar los riesgos críticos en una etapa temprana del ciclo de vida. Los resultados de cada iteración, en especial los de la fase de Elaboración, deben ser seleccionados en un orden que asegure que los riesgos principales son considerados primero.

**Lenguaje Unificado de Modelado (UML)**

Lenguaje Unificado de Modelado (UML, por sus siglas en inglés, UnifiedModelingLanguage) es el lenguaje de modelado de sistemas de software más conocido y utilizado en la actualidad; está respaldado por el OMG (Object Management Group).

Es un lenguaje gráfico para visualizar, especificar, construir y documentar un sistema. UML ofrece un estándar para describir un "plano" del sistema (modelo), incluyendo aspectos conceptuales tales como procesos de negocio, funciones del sistema, y aspectos concretos como expresiones de lenguajes de programación, esquemas de bases de datos y compuestos reciclados.

Es importante remarcar que UML es un "lenguaje de modelado" para especificar o para describir métodos o procesos. Se utiliza para definir un sistema, para detallar los artefactos en el sistema y para documentar y construir. En otras palabras, es el lenguaje en el que está descrito el modelo.

**Fases**

El Proceso Unificado de desarrollo puede ser dividido en cuatro fases para su mejor desarrollo. Estas fases ayudando tanto a la elaboración como a la resolución de problemas.

**Inicio**

En la fase de inicio se define el negocio: facilidad de realizar el proyecto, se presenta un modelo, visión, metas, deseos del usuario, plazos, costos y viabilidad.

**Elaboración**

En esta fase se obtiene la visión refinada del proyecto a realizar, la implementación iterativa del núcleo del de la aplicación, la resolución de riesgos altos, nuevos requisitos y se ajustan las estimaciones.

**Construcción**

Esta abarca la evolución hasta convertirse en producto listo incluyendo requisitos mínimos. Aquí se afinan los detalles menores como los diferentes tipos de casos o los riesgos menores.

**Transición**

En esta fase final, el programa debe estar listo para ser probado, instalado y utilizado por el cliente sin ningún problema. Una vez finalizada esta fase, se debe comenzar a pensar en futuras novedades para la misma.

Desde el punto de vista Técnico: el proyecto está formado por los flujos de trabajo fundamentales: captura de requerimientos, análisis, diseño, implementación y pruebas.

Tantos el punto de vista Gerencial como el Técnico concuerdan en: La iteración.

**A.6 Distribución de Contenido del Documento.**

**Flujos de Trabajo:**

**Modelo de Negocio.**

Dentro del modelo de negocio vemos cada proceso especificando sus datos, actividades y tares, el manejo de roles y todas las reglas que corresponden a este negocio en particular. Capturando los procesos de negocio se da inicio al modelado del Negocio, una vez terminado este modelo recien estaremos en condiciones de observar todos los requerimientos del sistema, pero como lo indica el Proceso Unificado de Desarrollo de Software el modelo de negocio debe hacerse basado en todas las funcionalidades del sistema en primer paso.

**Requerimientos del Sistema.**

En los requerimientos del sistema se expone la vista externa del usuario, en donde la interfaz es el lenguaje que utiliza el cliente para comunicarse con el sistema, en esta sección están todas las funciones fundamentales que satisfacen las necesidades del cliente. Esta interfaz debe estar orientada al cliente para su rápida comprensión.

**Análisis del Sistema**

El Análisis transforma los requerimientos antes descriptos en especificaciones de como se implementara el sistema. El análisis fundamentalmente consiste en obtener una visión que se preocupa de ver que hace el sistema de software a desarrollar, por tal motivo este se interesa en los requerimientos funcionales.

**Diseño de sistema**

El diseño comprende la segunda fase de la ingeniería de software. Es un refinamiento que toma en cuenta los requerimientos no funcionales, por lo cual se centra en cómo el sistema cumple sus objetivos. Es decir, en el análisis se definió qué debe hacer el sistema y cómo lo debe hacer. En el Diseño se define a través de que tecnología teniendo en cuenta el proceso que se definió en la primera fase. Es la definición de la arquitectura del sistema y del entorno tecnológico que le va a dar soporte, junto con la especificación detallada de los componentes del sistema de información.

**Implementación**

En este flujo de trabajo, una vez que se dispone de un diseño para la solución del problema se comienza a plasmar ese diseño en el código que permita realizar o implementar en ficheros fuente y ejecutables. Adicionalmente es necesario comprobar la ejecución correcta del módulo, por tanto interesan las pruebas que se hacen a nivel de módulo, también llamadas pruebas unitarias. El resultado final de este flujo de trabajo es un sistema ejecutable.

**Pruebas**

En este flujo se evaluará la calidad del producto que se está desarrollando. Consistirá en planificar qué es lo que hay que probar, diseñar cómo se va a hacer, implementar lo necesario para llevarlos a cabo, ejecutarlos en los niveles necesarios y obtener los resultados, de forma que la información obtenida nos sirva para ir refinando el producto a desarrollar.

B. Modelo de negocio

B.1 La Empresa

## DATOS DEL NEGOCIO

La empresa “EasyLocation” está ubicada en la ciudad de córdoba y se dedica exclusivamente a servicios de fácil Geo - localización de un listado de clientes en un Área especifica. Este servicio se exclusivamente a través de un sistema implementado en una plataforma web en donde los clientes pueden acceder mediante un login y cargar los datos de sus clientes y verlos inmediatamente posicionados en el mapa que provee la API V.3.0 de “googleMaps”.

Datos de la Empresa

Teléfono: (0351) 4843255.

Web.- [www.easylocation.com.ar](http://www.easylocation.com.ar)(dominio registrado a nombre de la empresa)

## Descripción de la empresa

La empresa se desarrolla en un ambiente familiar en donde el esfuerzo en conjunto a puesto en marcha la empresa, que tiene como clientes a cualquier empresa que tenga como tarea los despachos o las entregas a distintos clientes. Podemos decir como ejemplo de cliente, la Lomiteria “Betos” ubicada en la Av. Fuerza Aérea 2450, en donde simplemente con cargar los datos de sus clientes exportados de su sistema podrá ver inmediatamente ubicados en un mapa la exacta localización de todos los clientes de la lista. La lista se trata de un archivo .xls (excel). La mayoría de los cuentan con sistemas informáticos y tiene también la posibilidad de exportar su información, como todos usan archivos xls, de ahí nació la idea de poder posicionar esos datos en un mapa.

## Rubro

El rubro de esta empresa es enteramente de servicios de internet, una empresa dedica al 100% de soluciones de IT. Debemos destacar que la empresa es muy nueva y brinda soluciones puntuales, con el tiempo se irán añadiendo más soluciones al sistema y por consiguiente la empresa se expenderá con cada nueva funcionalidad.

**Domicilio de la Empresa**

La empresa se encuentra ubicada en la calle “Pedro Domingo Isnardi N° 4250” del B° Las Palmas de la ciudad Córdoba, Argentina.

**Objetivo General**

El objetivo siempre y desde un principio ha estado muy claro para los integrantes de la empresa, la completa satisfacción de los clientes en cuanto a todas las soluciones y funcionalidades que brinda el sistema.

**Políticas y Calidad**

Política:

* Satisfacción del cliente
* Actualización constante de las tecnologías existentes

Calidad.

Sistema sin “ISSUES” que entorpezcan la experiencia del cliente.

## Factores critico de éxitos

Mantener el nivel de calidad en su punto máximo sin superar el presupuesto fijado.

## Funcionalidad de la Empresa

En este punto nos detendremos para explicar más detalladamente la función final de la empresa, o sea de que se trata el servicio completo que se le ofrece al cliente.

El cliente, con su cuenta ya creada, tiene la posibilidad de subir a nuestro sistema un archivo xls, con todos los datos pertinentes de sus clientes, una vez terminado el proceso el cliente vera el resultado en un mapa de googlemap donde le situara cada cliente en su respectiva posición, de esta manera el cliente puede ver representativamente la expansión de sus ventas y también compararlos con subidas anteriores, y aquí mencionamos una funcionalidad inédita, ya que no hay en el mercado la posibilidad de ver historiales de repartos. De esta manera el cliente compara con mapas anteriores que tanto se expandió o no desde la última vez.

**B. 2 Usuarios y Privilegios**

**B.2.1 Usuarios.**

El sistema está diseñado para soportar multi-usuarios, esto significa que la cantidad de usuarios es ilimitada y todos los usuarios fueron dados de alta en el sistema solo por usuarios que cuenten con el privilegio para dar de alta. El sistema inicia con un Administrador General creador de la aplicación que tiene las capacidades y los privilegios para dar de alta a cada usuario con sus respectivos datos. Cada usuario por defecto en el sistema solo puede modificar sus datos y tiene las funciones principales activadas, cada privilegio es un producto y todo producto es comercializable, entonces cada función está sujeta a comercialización que es en si el objetivo principal de cualquier empresa.

**B.2.2 Privilegios.**

La jerarquización de los privilegios esta seteada desde el modelo de negocio, aquí se fijaron las pautas para las funciona que originalmente tendrán disponibles los usuarios y las funciones que se tendrán que pagar para poder accederlas, un usuario con privilegios puede crear un "Privilegio" pero si para que este, esté disponible deberá comprar la funcionalidad al administrador general del sistema, es obvio que nadie creara privilegios que no ha comprado pero puede crearlos en la base de datos a la espera del alta en el sistema principal. Los privilegios originales del sistema está altamente ligado con los casos de uso de este, quiere decir que casi el sistema no tiene funcionalidades que no se puedan comprar.

**B.2.3 Tabla de Privilegios actuales en el Sistema**.

|  |  |
| --- | --- |
| priviId | privilegio |
| 1 | altaPersona |
| 4 | altaPrivilegio |
| 6 | asignaPrivilegio |
| 2 | bajaPersona |
| 5 | bajaPrivilegio |
| 8 | exportaInfo |
| 3 | modificaPersona |
| 7 | SuperAdmin |
| 9 | visualizaGraficos |

Estos Privilegios están presentes actualmente en la base de datos del sistema y son esenciales para el funcionamiento completo del mismo, es importante destacar que por lo menos un usuario deberá tener todos los privilegios, de esta manera se transformara en el administrador y será la persona que originalmente le dé inicio al sistema en sí.

El usuario que cuente con el privilegio de "SuperAdmin" cargara con una responsabilidad importante ya que la función principal de este privilegio es ver y exportar los mapas generados por todos los usuarios del sitio y esto en si es sensible y riesgoso, pero como todo sistema debe tener un administrador, se agregó esta función como complementando los demás privilegios que son de ABM (Alta - Baja - Modificación) generales.

## C. Requerimientos y Datos Técnicos:

* Conectividad WAN (World Area Network).
* Motor de Base de Datos.
* Disco rígido de 500MB o superior.
* Servidor, compatible con Tomcat v7.

## C.1 Software de desarrollo

Todo el proyecto estará basado en la arquitectura Web de Java J2EE con el JDK 1.7.

* Enterprise JavaBeans (EJB).
* Java Servlet
* Java Server Page (JSP)
* Java Server Pages Standard Tag Library (JSTL).
* Java Hibernate.
* Java Spring.
* Java Server Faces (JSF)
* Java MessageService (JMS).
* Java Transaction API (JTA).
* JavaMail API y JavaBeans Activation Framework (JAF).
* Tecnologías XML (JAXP, JAX-RPC, JAX-WS, JAXB, SAAJ, JAXR)
* JPA, JDBC API
* Java Naming and Directory Interface (JNDI)
* Java Authentication and Authorization Service (JAAS)
* La capa del cliente (Client-tier) que será la capa destinada a mostrar la interfaz gráfica de usuario.   
  El sistema bajo plataforma Java EE, será una aplicación Web renderizada en un navegador web. Esta capa se ejecutará en la computadora cliente.
* La capa de la lógica de negocio (Business-tier) y la capa de la lógica de presentación (Web-tier). Estas capas se ejecutarán en el servidor de aplicaciones (Servidor tomcat).
* La capa de los datos (Data-tier) que será la capa destinada a la gestión de los datos y se ejecutará en un servidor de base de datos (Servidor MySQL).
* Así mismo se han utilizado dos librerías de componentes:
* JQuery: permite simplificar la manera de interactuar con los documentos HTML, manipular el árbol DOM, manejar eventos, desarrollar animaciones y agregar interacción con la técnica AJAX a páginas web.

D. Plan de proyecto.

D.1 Comenzamos con el Plan de Proyecto

Mediante este Plan de Proyecto vamos a guiar la total ejecución del proyecto, explicitar los resultados esperados, documentar las restricciones y supuestos asumidos durante la planificación, facilitar la comunicación entre las entidades involucradas y proporcionar las bases para la medición del progreso y el control del proyecto.

Aquí buscamos delimitar el proyecto indicando sus alcances, justificar la importancia y necesidad del mismo, explicitar supuestos y restricciones a tener en cuenta, presentar una agenda tentativa de trabajo y determinar los resultados esperados del proyecto. Así mismo, se hará referencias a proyectos asociados al sistema software, tales como los que permitieron conformar la plataforma de hardware sobre la que será implementado.

Como generalidad, podemos decir que el proyecto conseguirá generar una serie de artefactos, tales como: software, arquitectura de hardware, documentación específica, planes de capacitación en el uso del sistema y manual del usuario.

En particular, el proyecto tiene como objetivo principal brindar referenciar geográficamente direcciones en general resueltas solamente con la dirección física y no con los datos de latitud y longitud.

Es importante destacar que toda la información que es subida al sistema o sea guardada en la base de datos instalada en el servidor está sujeta a exportación y análisis de gráficos, de esta manera se da un servicio también de “BackUP” de los clientes o direcciones que se quieran guardar.

Se trabajará bajo una forma de implementación paulatina del sistema en la firma, en virtud del Método de Conversión Por Fases y del Prototipo Evolutivo, que permitirá una inclusión y una familiarización gradual con el mismo, al tiempo que permitirá contemplar y corregir errores sobre la marcha.

El proyecto se basa en las siguientes premisas:

* El sistema a crear debe ser realista, útil a la organización, y generar un cambio genuino en la misma. La idea es que el sistema software-hardware a implementar, sea soporte de la reestructuración de la empresa, y no que se genere en torno a un accionar rutinario y quizás no del todo racionalizado.
* Como consecuencia de lo anterior, se comenzará con un análisis de la empresa, un diagnóstico de la misma y el estudio de todos los cambios posibles. Los cambios deben gozar de compromiso, y el sistema debe ser un reflejo de ello.
* El sistema en general debe ser amigable y transparente en su uso, alejando a los usuarios de los detalles de implementación.

Concluyendo decimos que se estudiará el sistema-empresa, se diagnosticará y se propondrán cambios. Tras ello, se hará efectivo el rediseño. Como apoyo y reflejo del nuevo sistema-empresa, más eficiente, se generará un sistema software-hardware que coordine y organice las actividades de las áreas bajo estudio, del modo más eficaz y eficiente posible.

D.2 Visión Global del Proyecto

### D.2.1 Justificación del Proyecto.

Justificar el desarrollo, implementación y venta de este proyecto me parece como visión personal una de la tereas más importantes de este documento ya que sin esta justificación la idea en si puede ser una gran pérdida de tiempo para todos los que hemos participado en su creación.

La idea de “Easy Location” es una idea como cualquier otra, parte desde una necesidad y a través del conocimiento va tomando forma para volverse una solución efectiva dando un importante servicio a sus clientes, creo firmemente que representar geográficamente un listado de clientes tiene infinitas posibilidades de aplicación, no solo para un listado de clientes sino también para un listado de industrias o negocios o para lo que se quiera comparar en su localización con respecto al tiempo, entonces de esta manera exponemos las dos variables más relevantes del sistema, la localización y el tiempo y es en lo que si el sistema es, un gran localizador en función del tiempo, por eso es tan importante que cada vez que se guarde un mapa generado, en la base se guarde también la fecha y la hora en que se generó. Para terminar la justificación de que por que se debe realizar este proyecto, es porque veo un sistema de relativamente bajo costo de desarrollo y puede ser utilizado para cualquier negocio que lo necesite, de esta manera se puede tener una empresa dependiendo solamente del sistema.

### D.2.2 Supuestos y restricciones

Supuestos:

* El usuario con privilegios de “SuperAdmin” tendrá protagonismo activo del sistema en gran medida, existiendo una excelente interacción usuario-máquina.
* Información clara, que se generará en tiempo y forma para los usuarios.
* Los cálculos realizados por el software serán rigurosos, brindando información segura.

Restricciones:

* No contar con los datos necesarios en tiempo y forma.
* Necesitar más tiempo del planeado en el desarrollo del proyecto.
* Mala comunicación con los usuarios y apostadores del proyecto.
* Poco grado de involucramiento de todas o algunas de las partes en el desarrollo.
* Falta de especificación de requerimientos o modificación de estos.

### D.2.3 Resultados del proyecto.

**Modelo de negocio**: describe cómo desarrollar una visión de la nueva organización, y en base a esta visión define los roles y las responsabilidades mediante:

* Modelo de casos de uso de negocio
* Modelo de objetos de negocio

Además de estos modelos se apoyará en los siguientes resultados:

* Visión de negocio
* Actores de negocio
* Entidades de negocio
* Realización de los Casos de Uso del Negocio.
* Manual del usuario

### D.3 Plan de Fases.

El total del desarrollo de las 4 fases es de aproximadamente 3 meses, con la siguiente distribución:

Inicio = 1 mes, Elaboración y transición = 2 meses, transición = 1 mes.

**Inicio**: Primera fase del ciclo de vida del software, en la que la idea inicial definida para el desarrollo es refinada hasta el punto de quedar lo suficientemente bien establecida como para garantizar la entrada en la fase siguiente. Se define el negocio y sus objetivos, y la factibilidad del proyecto, se delimita el ámbito, se describe la arquitectura candidata y se identifican los riesgos críticos.

*Hito principal*: objetivos del ciclo de vida. Este Hito será aceptado bajo los siguientes criterios:

Objetivos:

* Claros
* Fácilmente medibles
* De importancia fundamental para el proyecto y la organización
* Conocidos por todos los participantes del proyecto.

**Elaboración**: se capturan los requerimientos restantes, se establece la arquitectura base para la construcción y se monitorean los riesgos.

*Hito principal*: arquitectura del ciclo de vida. Criterios de aceptación:

La arquitectura deberá:

* Comprender el sistema
* Organizar el desarrollo
* Fomentar la reutilización
* Hacer evolucionar el sistema

**Construcción:** en esta fase se realiza el sistema según una arquitectura estable, se genera una versión beta del mismo y se crea material del usuario.

*Hito principal*: funcionalidad operativa inicial. Criterios de aceptación:

Funcionalmente deberá ser:

* Amigable al usuario
* Fácil de interpretar y utilizar
* Portable en varios sistemas
* Funcional en todos los sentidos.

**Transición**: se actualiza el entorno en que va a funcionar el sistema, se crean manuales del sistema, se ajusta el software al entorno de trabajo, se corrigen errores y deficiencias y se genera una versión formal del sistema.

*Hito principal*: versión del producto. Criterio de aceptación: garantía de que se cuenta con un producto preparado para entregar a la comunidad de usuarios.

### D.4 Supervisión del proyecto.

#### D.4.1 Control del alcance

Como es de esperar ante la búsqueda de un desarrollo flexible a las necesidades del usuario, el sistema puede requerir cambios en el alcance planificado. Estos cambios deberán ir controlándose durante el proceso. Para llevar a cabo el control se realizarán reuniones con el / los usuario/s mediante las cuales se intentará detectar los desvíos y realizar las correcciones correspondientes.

#### D.4.2 Control de agenda

Para llevar adelante el control de los tiempos planeados, se podrán realizar controles de los entregables, e ir viendo las desviaciones producidas. Además, como se trata de un grupo de diseño, se podrán realizar reuniones para evaluar en equipo los tiempos. En base a las observaciones realizadas, en caso de encontrar demoras o tiempos fuera de lo planificado, se podrán re planificar las mejoras que correspondan.

#### D.4.3 Control de Presupuesto

Debe hallarse un presupuesto detallado en el comienzo del proyecto así como también un presupuesto para gastos circunstanciales que surjan sobre el progreso.

#### D.4.4 Control de calidad

El control de los resultados del proyecto, también será posible desde las distintas reuniones que se van a ir realizando por las diversas causas, analizando si los requerimientos de los clientes usuarios van siendo satisfechos, y en caso de que no lo sean podrán ir reformándose los distintos puntos del proceso de desarrollo, para que sean satisfechos y se mantenga la calidad deseada.

#### D.4.5 Agenda de Reportes

Se darán reportes informales los cuales serán entregados y tratados por diversos medios de comunicación, usualmente vía correo electrónico, telefónica o personalmente.

Los diferentes artefactos generados llevarán consigo las aclaraciones y explicaciones de funcionamiento necesarias.

#### D.4.6 Aceptación del producto

Para que se produzca la aceptación del producto, el mismo deberá ser probado por los usuarios quedando aprobado de parte de estos. Para esto se pueden utilizar interfaces de prueba para que el usuario experimente y califique.

Para que el producto sea aceptado de esta manera desde un comienzo, se deberán cumplir los ítems concebidos en el alcance del producto.

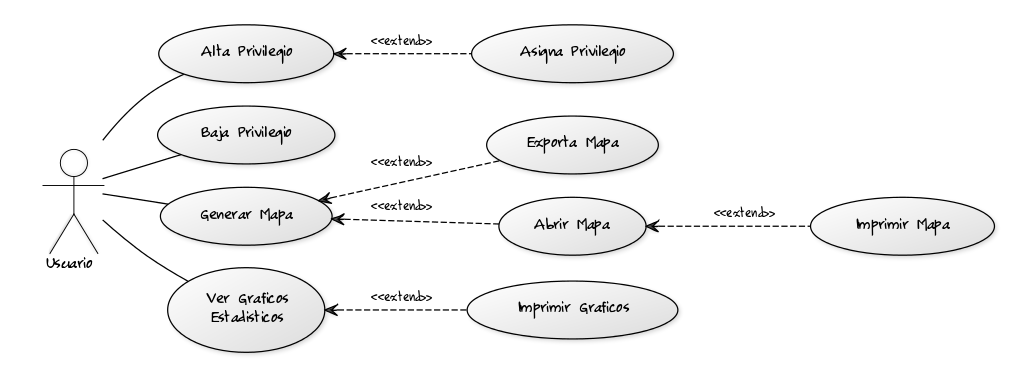
E. Modelo de UC (Use Case) de Negocio

E.1 Listado de Casos de Uso de Negocios

* Alta Privilegio
* Baja Privilegio
* Asigna Privilegio
* Generar Mapa
* Ver Gráficos Estadísticos
* Imprimir Gráficos Estadísticos
* Abrir Mapa
* Imprimir Mapa
* Exportar Información de Mapa.

Definimos a todos los casos de uso que tienen que ver con los privilegios como parte del negocio del sistema ya que están comprendidos en las ventas de productos finales, los privilegios son comercializables y pertenecen al negocio y por ende deben estar en el modelado final del negocio.

E.2 Modelado de UC (Use Case) de Negocio.



E.3 Descripción de los casos de usos de negocio

**E.3.1 Use Case - Alta Privilegio**

|  |  |  |
| --- | --- | --- |
| **Especificación de casos de uso** | | |
| **Nombre:** Alta Privilegio | **Autor:** Alejandro Manuel Sosa Adorati **Versión:** 1.0 | |
| **Nivel del CU**: X Negocio Sistema de Información | | |
| **Objetivo:** Crear una restricción para una funcionalidad del sistema con la capacidad de volverse un producto vendible. | | |
| **Prioridad: A** (**A**lta, **M**edia, **B**aja) | **Actor Principal: Usuario** | |
| **Complejidad: A** (**A**lta, **M**edia, **B**aja) | **Actor Secundario: ----** | |
| **Tipo**: **C** (**C**oncreto, **A**bstracto) | **Extiende en:**   * Asigna Privilegio   **Usa:** | |
| **Precondiciones:** Que el usuario tenga el privilegio de “AltaPrivilegio” y que la función exista en el código, solo el desarrollador del sistema puede crear la función. | | |
| **Post Condiciones:** Checkear que el privilegio haya sido de alta correctamente. | | |
| **Curso Normal** | | **Curso Alternativo** |
| El usuario se loguea en el sitio mediante usuario y contraseña. | | El usuario no está dado de alta, no puede ingresar al sitio. Fin del use case. |
| Logueo exitoso en el sistema, se cargan en una variable de sesión todos los privilegios del usuario | |  |
| El usuario se dirige a la sección de “Alta Privilegio”. | | El usuario no tiene el privilegio de “altaPrivilegio”. Fin del use case. |
| El usuario abre la sección de “Alta Privilegio” | |  |
| El Usuario, ingresa el nombre del Privilegio exactamente con las especificaciones que le dio el administrador del sistema | | EL usuario ingresa el un nombre incorrecto. Fin del use case. |
| El usuario, da de alta el nuevo “Privilegio” | |  |
| El usuario, comprueba que la nueva funcionalidad, este disponible. | | La funcionalidad no se encuentra disponible. Fin del use case. |
| Fin del UC. | |  |

**E.3.2 Use Case - Baja Privilegio**

|  |  |  |
| --- | --- | --- |
| **Especificación de casos de uso** | | |
| **Nombre:** Baja Privilegio | **Autor:** Alejandro Manuel Sosa Adorati **Versión:** 1.0 | |
| **Nivel del CU**: X Negocio Sistema de Información | | |
| **Objetivo** Quitar una funcionalidad del sistema. | | |
| **Prioridad: A** (**A**lta, **M**edia, **B**aja) | **Actor Principal: Usuario** | |
| **Complejidad: A** (**A**lta, **M**edia, **B**aja) | **Actor Secundario: ----** | |
| **Tipo**: **C** (**C**oncreto, **A**bstracto) | **Extiende en:**  **Usa:**   * Alta Privilegio | |
| **Precondiciones:** El privilegio debe haber sido dado de alta y el usuario debe tener el privilegio de dar de bajas a los privilegios. | | |
| **Post Condiciones:** Checkear que el privilegio haya sido dado de baja correctamente. | | |
| **Curso Normal** | | **Curso Alternativo** |
| El usuario se loguea en el sitio mediante usuario y contraseña. | | El usuario no está dado de alta, no puede ingresar al sitio. Fin del use case. |
| Logueo exitoso en el sistema, se cargan en una variable de sesión todos los privilegios del usuario | |  |
| El usuario se dirige a la sección de “Baja Privilegio”. | | El usuario no tiene el privilegio de “bajaPrivilegio”. Fin del use case. |
| El usuario selección el privilegio | |  |
| El usuario Elimina el privilegio | |  |
| Todos los usuario pierden esa funcionalidad en el sistema | |  |
| Fin del UC. | |  |

**E.3.3 Use Case - Asigna Privilegio**

|  |  |  |
| --- | --- | --- |
| **Especificación de casos de uso** | | |
| **Nombre:** Asigna Privilegio. | **Autor:** Alejandro Manuel Sosa Adorati **Versión:** 1.0 | |
| **Nivel del CU**: X Negocio Sistema de Información | | |
| **Objetivo:** Funcionalidad principal de asignación de funciones disponibles para cada usuario. | | |
| **Prioridad: A** (**A**lta, **M**edia, **B**aja) | **Actor Principal: Usuario** | |
| **Complejidad: A** (**A**lta, **M**edia, **B**aja) | **Actor Secundario: ----** | |
| **Tipo**: **C** (**C**oncreto, **A**bstracto) | **Extiende en:**  **Usa:**   * Alta Privilegio | |
| **Precondiciones:** Que el privilegio exista y que el usuario tenga el privilegio “asignaPrivilegio”. | | |
| **Post Condiciones:** Chequear la asignación de privilegios correctamente. | | |
| **Curso Normal** | | **Curso Alternativo** |
| El usuario se loguea en el sitio mediante usuario y contraseña. | | El usuario no está dado de alta, no puede ingresar al sitio. Fin del use case. |
| Logueo exitoso en el sistema, se cargan en una variable de sesión todos los privilegios del usuario. | |  |
| El usuario se dirige a la sección de “Asigna Privilegio”. | | El usuario no tiene el privilegio de “Asigna Privilegio”. Fin del use case. |
| El usuario abre la sección de “Asigna Privilegio”. | |  |
| El Usuario, asigna los privilegios existentes al usuario seleccionado. | |  |
| El usuario acepta la asignación. | |  |
| El usuario, comprueba que la asignación fue exitosa. | |  |
| Fin del UC. | |  |

**E.3.4 Use Case – Generar Mapa**

|  |  |  |
| --- | --- | --- |
| **Especificación de casos de uso** | | |
| **Nombre:** Generar Mapa. | **Autor:** Alejandro Manuel Sosa Adorati **Versión:** 1.0 | |
| **Nivel del CU**: X Negocio Sistema de Información | | |
| **Objetivo:** Funcionalidad principal para generar un mapa partiendo de un archivo Excel. | | |
| **Prioridad: A** (**A**lta, **M**edia, **B**aja) | **Actor Principal: Usuario** | |
| **Complejidad: A** (**A**lta, **M**edia, **B**aja) | **Actor Secundario: ----** | |
| **Tipo**: **C** (**C**oncreto, **A**bstracto) | **Extiende en:**   * Abrir Mapa * Exporta Mapa   **Usa:** | |
| **Precondiciones:** Contar con un archivo .xls que tenga el formato prefijado con las direcciones de los clientes cargadas. | | |
| **Post Condiciones:** | | |
| **Curso Normal** | | **Curso Alternativo** |
| El usuario se loguea en el sitio mediante usuario y contraseña. | | El usuario no está dado de alta, no puede ingresar al sitio. Fin del use case. |
| Logueo exitoso en el sistema, se cargan en una variable de sesión todos los privilegios del usuario. | |  |
| El usuario abre la sección Generar Mapa. | |  |
| El usuario carga el archivo .xls que contiene todas las dirección de los clientes. | | El usuario no cuenta con un archivo .xls. Fin del use case. |
| El mapa generado es abierto para corroborar que todo funciona con se previó. | |  |
| Fin del UC. | |  |

**E.3.5 Use Case – Ver Gráficos Estadísticos**

|  |  |  |
| --- | --- | --- |
| **Especificación de casos de uso** | | |
| **Nombre:** Ver Gráficos Estadísticos. | **Autor:** Alejandro Manuel Sosa Adorati **Versión:** 1.0 | |
| **Nivel del CU**: X Negocio Sistema de Información | | |
| **Objetivo:** Partiendo de los mapas generados se puede evaluar toda la información subida al sistema. | | |
| **Prioridad: A** (**A**lta, **M**edia, **B**aja) | **Actor Principal: Usuario** | |
| **Complejidad: A** (**A**lta, **M**edia, **B**aja) | **Actor Secundario: ----** | |
| **Tipo**: **C** (**C**oncreto, **A**bstracto) | **Extiende en:**   * Imprimir Grafico   **Usa:** | |
| **Precondiciones:** Se necesita por lo menos un mapa generado para poder generar los gráficos estadísticos y contar con el privilegio de ver los gráficos estadísticos. | | |
| **Post Condiciones:** | | |
| **Curso Normal** | | **Curso Alternativo** |
| El usuario se loguea en el sitio mediante usuario y contraseña. | | El usuario no está dado de alta, no puede ingresar al sitio. Fin del use case. |
| Logueo exitoso en el sistema, se cargan en una variable de sesión todos los privilegios del usuario. | |  |
| El usuario se dirige a la sección de Ver Gráficos Estadísticos | | El usuario no tiene el privilegio de ver los gráficos estadísticos. Fin del Use Case. |
| El usuario abre la Sección de Ver gráficos Estadísticos | |  |
| El usuario selecciona la fechas para comparar los mapas generados | |  |
| El gráficos muestras los mapas generados en formato de grafico de barras | | No se encuentra ningún mapa generado entre las fechas seleccionadas. Fin del use case |
| Fin del UC. | |  |

**E.3.6 Use Case – Imprimir Gráficos Estadísticos**

|  |  |  |
| --- | --- | --- |
| **Especificación de casos de uso** | | |
| **Nombre:** Imprimir Gráficos Estadísticos. | **Autor:** Alejandro Manuel Sosa Adorati **Versión:** 1.0 | |
| **Nivel del CU**: X Negocio Sistema de Información | | |
| **Objetivo:** Imprimir en un archivo .pdf un gráfico generado. | | |
| **Prioridad: M** (**A**lta, **M**edia, **B**aja) | **Actor Principal: Usuario** | |
| **Complejidad: M** (**A**lta, **M**edia, **B**aja) | **Actor Secundario: ----** | |
| **Tipo**: **C** (**C**oncreto, **A**bstracto) | **Extiende en:**  **Usa:**   * Ver Gráficos estadísticos. | |
| **Precondiciones:** Se necesita un gráfico de barras ya generado. | | |
| **Post Condiciones:** | | |
| **Curso Normal** | | **Curso Alternativo** |
| El caso de uso inicia cuando se generó un gráfico de barras estadístico. | |  |
| El usuario selecciona imprimir (crtl+p) | |  |
| El usuario selecciona la opción de guardar como pdf. | | El usuario selecciona otra opción. Fin del use case. |
| El archivo es generado exitosamente | |  |
| El usuario abre el archivo con el grafico generado | |  |
| Fin del UC. | |  |

**E.3.7 Use Case – Abrir Mapa**

|  |  |  |
| --- | --- | --- |
| **Especificación de casos de uso** | | |
| **Nombre:** Abrir Mapa. | **Autor:** Alejandro Manuel Sosa Adorati **Versión:** 1.0 | |
| **Nivel del CU**: X Negocio Sistema de Información | | |
| **Objetivo:** Abrir un mapa generado. | | |
| **Prioridad: A** (**A**lta, **M**edia, **B**aja) | **Actor Principal: Usuario** | |
| **Complejidad: A** (**A**lta, **M**edia, **B**aja) | **Actor Secundario: ----** | |
| **Tipo**: **C** (**C**oncreto, **A**bstracto) | **Extiende en:**  **Usa:**   * Generar Mapa. | |
| **Precondiciones:** Se necesita haber creado un mapa con anterioridad. | | |
| **Post Condiciones:** | | |
| **Curso Normal** | | **Curso Alternativo** |
| El usuario se loguea en el sitio mediante usuario y contraseña. | | El usuario no está dado de alta, no puede ingresar al sitio. Fin del use case. |
| En el listado de mapas generados el usuario selecciona el mapa a abrir | | No existe un mapa generado. Fin del Use case |
| El usuario abre el mapa seleccionado | |  |
| Fin del UC. | |  |

**E.3.8 Use Case – Imprimir Mapa**

|  |  |  |
| --- | --- | --- |
| **Especificación de casos de uso** | | |
| **Nombre:** Imprimir Mapa. | **Autor:** Alejandro Manuel Sosa Adorati **Versión:** 1.0 | |
| **Nivel del CU**: X Negocio Sistema de Información | | |
| **Objetivo:** Imprimir en un archivo .pdf un mapa generado. | | |
| **Prioridad: M** (**A**lta, **M**edia, **B**aja) | **Actor Principal: Usuario** | |
| **Complejidad: M** (**A**lta, **M**edia, **B**aja) | **Actor Secundario: ----** | |
| **Tipo**: **C** (**C**oncreto, **A**bstracto) | **Extiende en:**  **Usa:**   * Ver Gráficos estadísticos. | |
| **Precondiciones:** Se necesita un mapa ya generado. | | |
| **Post Condiciones:** | | |
| **Curso Normal** | | **Curso Alternativo** |
| El caso de uso inicia cuando se abre un mapa generado. | |  |
| El usuario selecciona imprimir (crtl+p) | |  |
| El usuario selecciona la opción de guardar como pdf. | | El usuario selecciona otra opción. Fin del use case. |
| El archivo es generado exitosamente | |  |
| El usuario abre el archivo con el mapa generado | |  |
| Fin del UC. | |  |

**E.3.9 Use Case – Exporta Información de Mapas.**

|  |  |  |
| --- | --- | --- |
| **Especificación de casos de uso** | | |
| **Nombre:** Exporta Información de Mapas. | **Autor:** Alejandro Manuel Sosa Adorati **Versión:** 1.0 | |
| **Nivel del CU**: X Negocio Sistema de Información | | |
| **Objetivo:** Exportar en un archivo .cvs la información completa del mapa generado. | | |
| **Prioridad: A** (**A**lta, **M**edia, **B**aja) | **Actor Principal: Usuario** | |
| **Complejidad: A** (**A**lta, **M**edia, **B**aja) | **Actor Secundario: ----** | |
| **Tipo**: **C** (**C**oncreto, **A**bstracto) | **Extiende en:**  **Usa:**   * Genera Mapa. | |
| **Precondiciones:** Se necesita un mapa ya generado y el privilegio de exportar información. | | |
| **Post Condiciones:** | | |
| **Curso Normal** | | **Curso Alternativo** |
| El usuario se loguea en el sitio mediante usuario y contraseña. | | El usuario no está dado de alta, no puede ingresar al sitio. Fin del use case. |
| En el listado de mapas generados el usuario selecciona el mapa a exportar. | | No existe un mapa generado. Fin del Use case |
| El usuario exporta la información. | | El usuario selecciona otra opción. Fin del use case. |
| El archivo es generado exitosamente | |  |
| El usuario abre el archivo con los datos exportados | |  |
| Fin del UC. | |  |

E. Captura de Requerimientos

E.1 Introducción

E.1.1Propósito

Sistematizar los resultados generados durante el flujo de trabajo de captura de requerimientos.

E.1.2 Alcances

Recoger todos los artefactos generados durante la fase de trabajo de requerimientos del sistema a implementarse en la empresa Easy Location.

E.2 Listado de Casos de Uso del Sistema:

Cuando hablamos del sistema a diferencia del negocio vamos a listar todos los casos de uso que el sistema cuenta, y procederemos a documenta con granularidad gruesa cada uno de ellos. Es importante diferenciar entre el negocio y el sistema para documentar debidamente el procedimiento de caso de uso, si no nos veríamos documentando siempre la misma información y cometeríamos el pecado de la redundancia.

Los casos de uso evitan típicamente el lenguaje técnico, prefiriendo la lengua del usuario final o del experto del campo del saber al que se va a aplicar. Los casos del uso son a menudo elaborados en colaboración por los analistas de requerimientos y los clientes.

Cada caso de uso se centra en describir cómo alcanzar una única meta o tarea. Desde una perspectiva tradicional de la ingeniería de software, un caso de uso describe una característica del sistema. Para la mayoría de proyectos de software, esto significa que quizás a veces es necesario especificar decenas o centenares de casos de uso para definir completamente el nuevo sistema. El grado de la formalidad de un proyecto particular del software y de la etapa del proyecto influenciará el nivel del detalle requerido en cada caso de uso.

Los casos de uso pretenden ser herramientas simples para describir el comportamiento del software o de los sistemas. Un caso de uso contiene una descripción textual de todas las maneras que los actores previstos podrían trabajar con el software o el sistema. Los casos de uso no describen ninguna funcionalidad interna (oculta al exterior) del sistema, ni explican cómo se implementará. Simplemente muestran los pasos que el actor sigue para realizar una operación.

E.2.1 Listado de Casos de Uso imprescindibles en el Sistema.

1- Casos de Uso del Usuario.

* Alta Usuario ( Se crea un nuevo Usuario)
* Baja Usuario (Se elimina un usuario existente)
* Modifica Usuario (Se modifican los datos del usuario existente)

2- Casos de Uso de Privilegios

* Alta Privilegio (Se crea un nuevo privilegio)
* Baja Privilegio (Se da de baja un privilegio existente)
* Asigna Privilegios (Se relacionan los privilegios con los usuarios)

3- Casos de Uso de Mapas

* Generar Mapa(Se cargan los datos de direcciones para generar un único mapa)
* Abrir Mapa (Se accede a un mapa ya generado)
* Exportar Mapa (Se exporta toda la información del mapa en un archivo .cvs)
* Imprimir Mapa (Se exporta el mapa gráficamente a un archivo .pdf)

4- Casos de Uso de Gráficos.

* Abrir Gráficos (Genera gráficos estadísticos entre fechas)
* Imprimir Gráficos (Imprime en .pdf el grafico generado)

E.3 Modelo de Casos de Uso de Sistema.