Requerimientos

Software de estudio de mercado

Versión documento

1.4

Integrantes

Leonardo Ugalde Arenas

Fatima Vianey Montaño Flores

Leonardo Uriel Peralta de Jesús

Alan Jesús Mendoza Bermúdez

Ana Maria Vigueras Muñoz

Randy Javier Villanueva Montes

Índice

[Introducción 2](#_Toc939373846)

[Sistema a construir 3](#_Toc1606496599)

[Tipos de usuarios 4](#_Toc1750018816)

[Requerimientos Funcionales 5](#_Toc598057595)

[Requerimientos no funcionales 7](#_Toc156736206)

[Restricciones 8](#_Toc24701809)

[Plan de trabajo 8](#_Toc924812073)

[Diagramas de caso de uso. 12](#_Toc1319479251)

[Diagramas de actividades. 15](#_Toc1745011314)

# Introducción

El siguiente documento esclarece toda la información necesaria dentro del análisis de los requerimientos del proyecto “Software de estudio de mercado”, basado en un mapa de software libre con información estadística, el cual consistirá en agregar una ubicación especifica dentro del mapa y en base a ciertos parámetros, el sistema mostrara al usuario si es buena opción poner el negocio en dicho lugar, además de la generación de reportes automáticos en base a la actividad que haya por los usuarios dentro del sistema.

# Sistema a construir

Un sistema web, que al igual funcionará como una aplicación hibrida para dispositivos móviles.

* Un inicio de sesión y una pantalla de registro, en cuanto a iniciar sesión se hará una petición a un servicio específico el cual se validará si dichos campos son correctos, en el caso del registro de cliente, al igual se hará una petición a un servicio específico para el registro del cliente con los datos ingresados.
* Vista del mapa de la República Mexicana, el cual para mostrarse se consumirá un API en *back-end* para su visualización.
* Búsqueda por filtros, las cuales se mostrarán opciones al cliente para que sean seleccionadas y se le muestren los resultados requeridos, en este caso.
* En pantalla si visualizar información de la ubicación determinado si es buena opción de negocio, el cual el mensaje dependerá en cuanto a todos los filtros que pase en *back-end.*
* Botón de guardar ubicación, se hará una petición a un servicio específico en *back-end* y guardará el nuevo registro en base de datos.
* Lista de los lugares guardados por el cliente, el cual, para visualizarlos en la pantalla del cliente, se hará una petición a un servicio específico en *back-end* de todos los lugares guardados por ese cliente.
* Actualización de las ubicaciones ya guardadas, se hará una petición a un servicio específico en *back-end* para poder actualizar cualquier registro que ya se haya hecho.
* Eliminación de los lugares guardados, el cual para eliminar los registros se hará una petición a un servicio específico en *back-end*, para hacer un borrado lógico en base de datos.

# Tipos de usuarios



Cliente

Administrador

# Requerimientos Funcionales

* Registro de cliente

Para llevar a cabo el registro de un cliente, al administrador se le mostrara un formulario donde se solicitarán los siguientes datos: nombre, apellido paterno, apellido materno, empresa, cargo, teléfono y correo.

* Consultar cliente registrado.

En una interfaz el administrador y el usuario pie podrá ver todos los clientes que ha dado de alta, donde se mostraran los siguientes datos de este: nombre, apellido paterno, apellido materno, empresa, cargo, teléfono y correo.

* Actualizar cliente registrado.

En esta parte se mostrarán todos los datos del cliente para su modificación.

* Eliminar cliente registrado.

En esta parte se deberá seleccionar un cliente y oprimir la opción de eliminar.

* Registro de usuarios.

Para añadir un nuevo usuario se visualizará un formulario que solicite los siguientes datos: nombre, apellido paterno, apellido materno, empresa, contraseña, teléfono, estatus, rol y nombre de usuario.

* Actualizar usuarios.

En esta parte se visualizarán los siguientes datos del usuario los cuales son los únicos que se podrán modificar nombre, apellido paterno, apellido materno, contraseña, teléfono, estatus y rol.

* Inicio de sesión de usuario.

Para iniciar sesión mostrará un interfaz que solicite el nombre de usuario y contraseña, de ser estas correctas las credenciales se le permita acceder al sistema.

* Inicio de sesión de administrador.

Para iniciar sesión como administrador se mostrará una interfaz donde se soliciten nombre de usuario y contraseña, de ser correctas las credenciales se le permitirá el acceso al sistema teniendo como primera vista el panel de control.

* Panel de control para el administrador

En el panel de control del administrador se podrá visualizar a los clientes que ha registrado, también tendrá la opción de acceder a la sección de reportes así mismo podrá acceder a la interfaz de registrar cliente.

* Vista del mapa de la República Mexicana

El mapa de la República Mexicana se mostrará en la interfaz principal del sistema, el mapa puede ir cambiando según la aplicación de filtros, en el mapa se pueden ver estados y municipios.

* Vista de la ubicación seleccionada

El sistema mostrara la información de la zona seleccionada.

* Búsqueda de una zona mediante filtros.

El sistema permitirá realizar búsquedas mediante filtros como el área geográfica, actividad económica, etc. dentro del mapa de la República Mexicana.

* Análisis de estudio de mercado económico

El sistema determinara si la ubicación seleccionada es buena opción de negocio para el cliente.

* Generación de reportes

El sistema generara reportes donde se especifique lo siguiente nombre del reporte, longitud, latitud, razón social, clase actividad, ubicación, colonia, calle, código postal y los datos del cliente.

* Consulta de reportes generados por el administrador.

El sistema tendrá un apartado donde se permitirá visualizar los reportes generados.

* Vista para cada rol.

Cada usuario tendrá diferente vista al panel de control con base a su rol

# Requerimientos no funcionales

* Uso de un navegador.

El sistema deberá ser compatible al motor del navegador de Google o algún otro.

* Uso de recaptcha o programas automatizados.

Utilizará un método de validación para proteger al sistema de spam y abusos de robots automatizados.

* Autentificación al iniciar sesión.

El sistema permitirá el acceso a los usuarios que estén previamente registrados y autentifiquen el código enviado a su número de teléfono.

* UX y UI en las interfaces.

Las interfaces deberán ser intuitivas y amigables para la navegabilidad del usuario.

* La aplicación consumirá peticiones basada en programación multihilo

El sistema deberá realizar diversas tareas que se ejecuten a la vez para agilizar la ejecución del proceso.

* Cifrado de contraseñas

El sistema contara con métodos hash para el cifrado de contraseñas.

# Restricciones

1. Nuevas *features* no previstas.
2. Consumo de algunas *API* que en su caso sean de paga.

# Plan de trabajo

|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| # | ACTIVIDADES | CONTROL | Semanas | | | | | | | | | | | |
| 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 10 | 11 | 12 |
| 1 | Análisis | | | | | | | | | | | | | |
| 1.1 | Identificación de la problemática | PROG. |  |  |  |  |  |  |  |  |  |  |  |  |
| REAL |  |  |  |  |  |  |  |  |  |  |  |  |
| 1.2 | Definición de objetivos | PROG. |  |  |  |  |  |  |  |  |  |  |  |  |
| REAL |  |  |  |  |  |  |  |  |  |  |  |  |
| 1.3 | Definición de alcance y limitaciones | PROG. |  |  |  |  |  |  |  |  |  |  |  |  |
| REAL |  |  |  |  |  |  |  |  |  |  |  |  |
| 1.4 | Definición de requerimientos funcionales y no funcionales | PROG. |  |  |  |  |  |  |  |  |  |  |  |  |
| REAL |  |  |  |  |  |  |  |  |  |  |  |  |
| 1.5 | Definición de usuarios | PROG. |  |  |  |  |  |  |  |  |  |  |  |  |
| REAL |  |  |  |  |  |  |  |  |  |  |  |  |
| 1.6 | Diagramas de casos de uso | PROG. |  |  |  |  |  |  |  |  |  |  |  |  |
| REAL |  |  |  |  |  |  |  |  |  |  |  |  |
| 1.7 | Diseño de la arquitectura del software | PROG. |  |  |  |  |  |  |  |  |  |  |  |  |
| REAL |  |  |  |  |  |  |  |  |  |  |  |  |
| 1.8 | Documentación de la fase de análisis | PROG. |  |  |  |  |  |  |  |  |  |  |  |  |
| REAL |  |  |  |  |  |  |  |  |  |  |  |  |
| 2 | Planeación y diseño | | | | | | | | | | | | | |
| 2.1 | Diseño conceptual de la base de datos entidad-relación | PROG. |  |  |  |  |  |  |  |  |  |  |  |  |
| REAL |  |  |  |  |  |  |  |  |  |  |  |  |
| 2.2 | Diseño lógico de la base de datos modelo relacional | PROG. |  |  |  |  |  |  |  |  |  |  |  |  |
| REAL |  |  |  |  |  |  |  |  |  |  |  |  |
| 2.3 | Creación del diccionario de datos | PROG. |  |  |  |  |  |  |  |  |  |  |  |  |
| REAL |  |  |  |  |  |  |  |  |  |  |  |  |
| 2.4 | Determinación de las herramientas tecnológicas y ambientación | PROG. |  |  |  |  |  |  |  |  |  |  |  |  |
| REAL |  |  |  |  |  |  |  |  |  |  |  |  |
| 2.5 | Documentación fase de planeación y diseño | PROG. |  |  |  |  |  |  |  |  |  |  |  |  |
| REAL |  |  |  |  |  |  |  |  |  |  |  |  |
| 3 | Desarrollo *back-end* | | | | | | | | | | | | | |
| 3.1 | Creación de la base de datos | PROG. |  |  |  |  |  |  |  |  |  |  |  |  |
| REAL |  |  |  |  |  |  |  |  |  |  |  |  |
| 3.2 | Definición de funciones | PROG. |  |  |  |  |  |  |  |  |  |  |  |  |
| REAL |  |  |  |  |  |  |  |  |  |  |  |  |
| 3.3 | Crear repositorio remoto para proyecto *back-end* | PROG. |  |  |  |  |  |  |  |  |  |  |  |  |
| REAL |  |  |  |  |  |  |  |  |  |  |  |  |
| 3.4 | Subir proyecto al repositorio | PROG. |  |  |  |  |  |  |  |  |  |  |  |  |
| REAL |  |  |  |  |  |  |  |  |  |  |  |  |
| 3.5 | Crear ramas para el desarrollo | PROG. |  |  |  |  |  |  |  |  |  |  |  |  |
| REAL |  |  |  |  |  |  |  |  |  |  |  |  |
| 3.6 | Crear *pull request* para revisión | PROG. |  |  |  |  |  |  |  |  |  |  |  |  |
| REAL |  |  |  |  |  |  |  |  |  |  |  |  |
| 3.7 | Merge entre rama y rama principal | PROG. |  |  |  |  |  |  |  |  |  |  |  |  |
| REAL |  |  |  |  |  |  |  |  |  |  |  |  |
| 3.8 | Creación de servicios *REST Post, Get, Put, Delete.* | PROG. |  |  |  |  |  |  |  |  |  |  |  |  |
| REAL |  |  |  |  |  |  |  |  |  |  |  |  |
| 3.9 | Configuración *Gateway Gunicorn* | PROG. |  |  |  |  |  |  |  |  |  |  |  |  |
| REAL |  |  |  |  |  |  |  |  |  |  |  |  |
| 3.5 | Configuración de la máquina virtual en *AWS* | PROG. |  |  |  |  |  |  |  |  |  |  |  |  |
| REAL |  |  |  |  |  |  |  |  |  |  |  |  |
| 3.6 | Documentación desarrollo *back-end* | PROG |  |  |  |  |  |  |  |  |  |  |  |  |
| REAL |  |  |  |  |  |  |  |  |  |  |  |  |
| 4 | Desarrollo *front-end* | | | | | | | | | | | | | |
| 4.1 | Diseño y creación del mapa de navegabilidad | PROG. |  |  |  |  |  |  |  |  |  |  |  |  |
| REAL |  |  |  |  |  |  |  |  |  |  |  |  |
| 4.2 | Diseño de los *mockups* | PROG. |  |  |  |  |  |  |  |  |  |  |  |  |
| REAL |  |  |  |  |  |  |  |  |  |  |  |  |
| 4.3 | Maquetación de las interfaces | PROG. |  |  |  |  |  |  |  |  |  |  |  |  |
| REAL |  |  |  |  |  |  |  |  |  |  |  |  |
| 4.4 | Crear repositorio remoto para proyecto *front-end* | PROG. |  |  |  |  |  |  |  |  |  |  |  |  |
| REAL |  |  |  |  |  |  |  |  |  |  |  |  |
| 4.5 | Subir proyecto al repositorio | PROG. |  |  |  |  |  |  |  |  |  |  |  |  |
| REAL |  |  |  |  |  |  |  |  |  |  |  |  |
| 4.6 | Crear ramas para el desarrollo | PROG. |  |  |  |  |  |  |  |  |  |  |  |  |
| REAL |  |  |  |  |  |  |  |  |  |  |  |  |
| 4.7 | Crear *pull request* para revisión. | PROG. |  |  |  |  |  |  |  |  |  |  |  |  |
| REAL |  |  |  |  |  |  |  |  |  |  |  |  |
| 4.8 | *Merge* entre ramas y rama principal | PROG. |  |  |  |  |  |  |  |  |  |  |  |  |
| REAL |  |  |  |  |  |  |  |  |  |  |  |  |
| 4.9 | Desarrollo de interfaces basadas en *single page application.* | PROG. |  |  |  |  |  |  |  |  |  |  |  |  |
| REAL |  |  |  |  |  |  |  |  |  |  |  |  |
| 4. | Consumo de los servicios creados en *back-end.* | PROG. |  |  |  |  |  |  |  |  |  |  |  |  |
| REAL |  |  |  |  |  |  |  |  |  |  |  |  |
| 4.6 | Consumo *API* del INEGI | PROG. |  |  |  |  |  |  |  |  |  |  |  |  |
| REAL |  |  |  |  |  |  |  |  |  |  |  |  |
| 4.7 | Validaciones en formularios. | PROG. |  |  |  |  |  |  |  |  |  |  |  |  |
| REAL |  |  |  |  |  |  |  |  |  |  |  |  |
| 4.8 | Documentación *Front-end* | PROG. |  |  |  |  |  |  |  |  |  |  |  |  |
| REAL |  |  |  |  |  |  |  |  |  |  |  |  |
| 5 | Pruebas del sistema y entrega | | | | | | | | | | | | | |
| 5.1 | Creación de pruebas unitarias | PROG. |  |  |  |  |  |  |  |  |  |  |  |  |
| REAL |  |  |  |  |  |  |  |  |  |  |  |  |
| 5.2 | Puesta en producción y configuraciones del *web container* | PROG. |  |  |  |  |  |  |  |  |  |  |  |  |
| REAL |  |  |  |  |  |  |  |  |  |  |  |  |
| 5.3 | Documentación final | PROG |  |  |  |  |  |  |  |  |  |  |  |  |
| REAL |  |  |  |  |  |  |  |  |  |  |  |  |
| 5.4 | Entrega del producto mínimo viable | PROG. |  |  |  |  |  |  |  |  |  |  |  |  |
| REAL |  |  |  |  |  |  |  |  |  |  |  |  |

# Diagramas de caso de uso.

En el siguiente apartado se muestra el diagrama de casos de uso general de la *Landing Page* (Página de destino), en este se pueden observar todas las acciones que realizara el sistema.

Diagrama

Descripción generada automáticamente

Figura 1 Diagrama de caso de uso General.

Diagrama

Descripción generada automáticamente

Figura 2 Vista Principal.

Se muestra el diagrama de casos de uso, en el cual se observan las interacciones con las que se relaciona el usuario.

Se muestra el diagrama de caso de uso sobre la gestión de los clientes que realiza para la acción que se desea realizar.

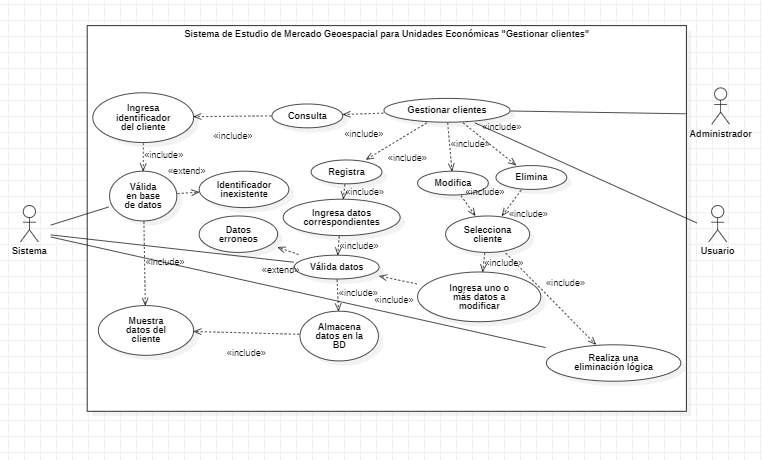


Figura 3 Vista Gestión de Clientes.

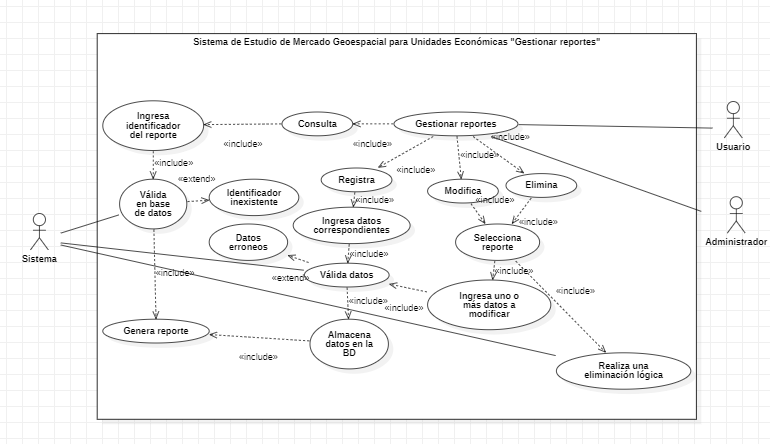


Figura 4 Vista Gestión de Reportes.

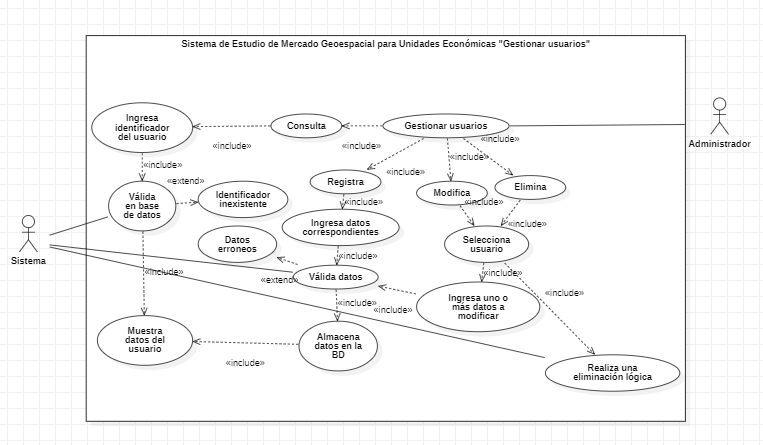


Figura 5 Vista Gestión de Usuarios.

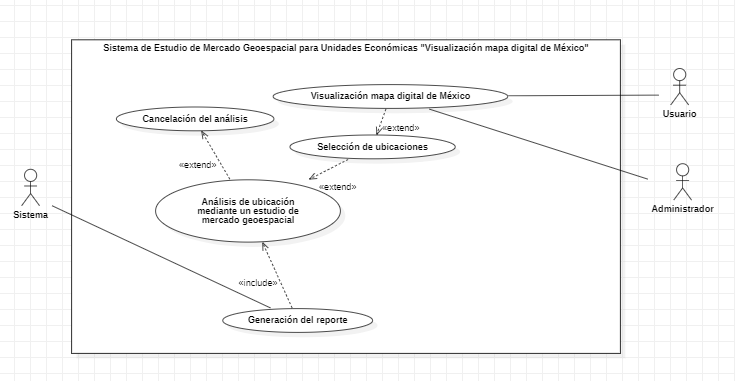


Figura 6 Vista del Menú.

# Diagramas de estados.

Se muestra el diagrama de actividades, el cual hace referencia a la vista principal de la *Landing Page* (Página de destino).



Figura 7 Diagrama de estado Ver reporte.

A continuación, en las siguientes figuras se mostrarán los diagramas de actividades sobre la gestión de clientes con los que cuenta dicha *Landing Page* (Página de destino).

Se muestra el diagrama de actividades para el registro de un nuevo usuario, en este se observan las acciones que debe seguir el administrador para realizar dicho registro en cada una de las acciones del diagrama.

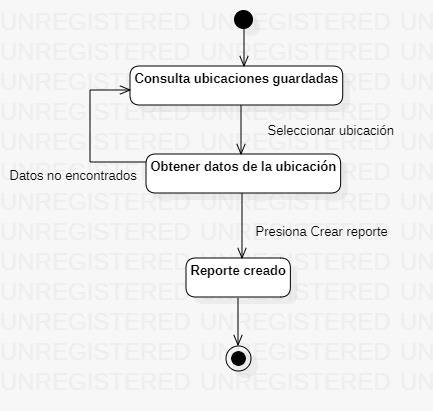


Figura 8 Diagrama de estado Crear reporte.

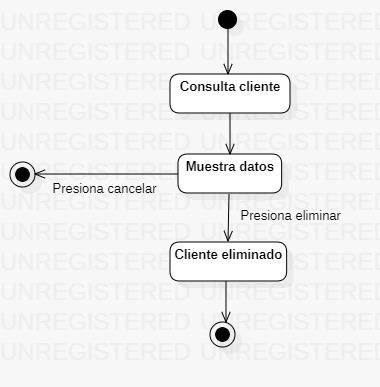


Figura 9 Diagrama de estado Borrar cliente

Diagrama

Descripción generada automáticamente

Figura 10 Diagrama de estado Borrar usuario.

Se muestra el diagrama de actividades el cual hace referencia a la consulta de los registros de dicha *Landing Page* (Página de destino).

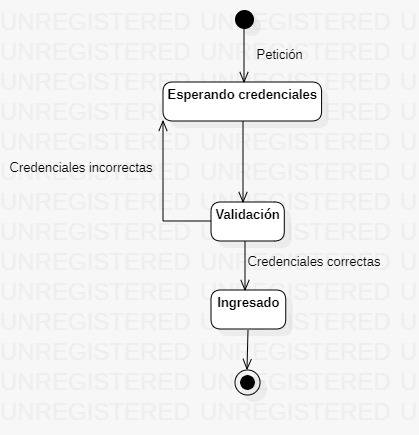


Figura 11 Diagrama de estado Iniciar sesión

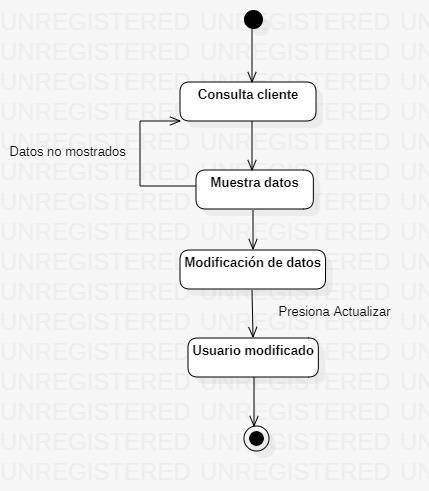


Figura 12 Diagrama de estado Modificar usuario.

Se muestra el diagrama de actividades, el cual hace referencia a la modificación de una ubicación.

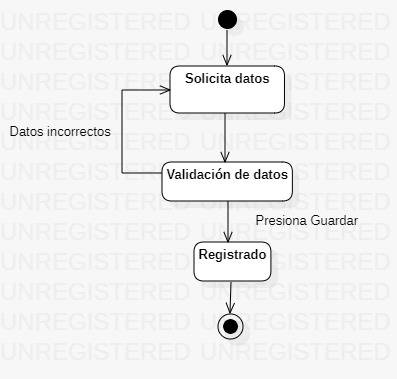


Figura 13 Diagrama de estado Registrar usuario o cliente.

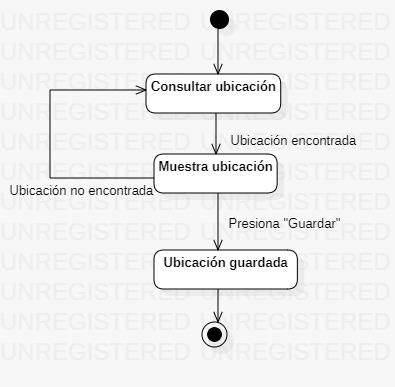


Figura 14 Diagrama de estado Guardar ubicación.