



# **BENEMERITA UNIVERSIDAD AUTONOMA DE PUEBLA**

Facultad de Ciencias de la Computación  
Ingeniería en Tecnologías de la Información  
Control de calidad de Software

## **Integrantes:**

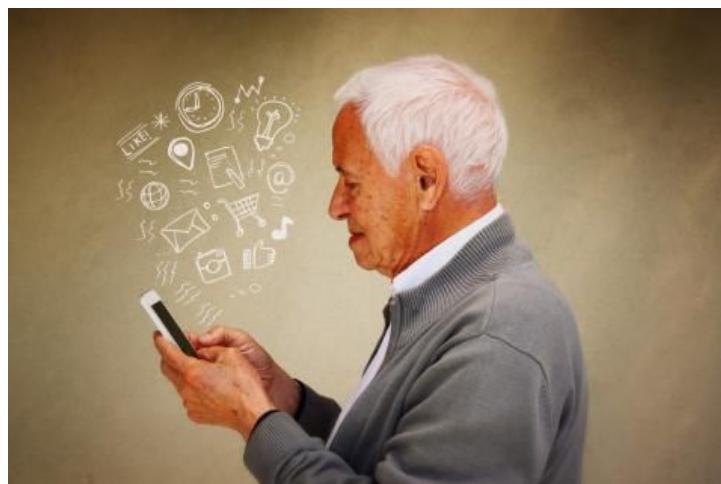
Citlali Castillo Morales  
Mitsu Biali Govea Saucedo

## **Docente:**

Dr. Juan Manuel González Calleros

## **Proyecto**

AS - AlzheimerSoftware



## Revisión histórica

Nombre	Fecha	Motivo	Versión
Mitsu Govea Citlali Morales	15 abril 2020	Estimación del tamaño del proyecto usando lógica difusa	0-1
Citlali Morales Mitsu Govea	15 mayo 2020	<ul style="list-style-type: none"><li>• Especificación de requerimientos</li><li>• Características del sistema</li><li>• Diagramas</li><li>• Diseño de interfaces</li></ul>	0-2

## Introducción

El Alzheimer es un problema cognitivo degenerativo que afecta principalmente a las personas de la tercera edad. Por lo tanto, para mejorar la calidad de vida de aquellos que padecen esta enfermedad, existen personas que se encargan de atender sus necesidades, como enfermeros, los propios familiares y especialistas que evalúan el estado mental y su evolución.

También existe una herramienta llamada telesalud, que ayuda al personal de salud para asistir a sus pacientes que se encuentran en zonas o situaciones que no permiten atención presencial.

## Problemática identificada

La atención por parte de los familiares, finalmente es lo más complicado, debido a que cada integrante tiene sus propias actividades, además de que el trato y la atención que dedican a su familiar no siempre es el óptimo.

Además, contar con la atención de una enfermera que atienda desde el hogar corresponde a un costo económico al que no todos tienen acceso.

## Solución propuesta

Se desarrollará una aplicación móvil dedicado a los familiares de la persona que padece de Alzheimer.

Este software ofrece exámenes cognitivos que miden el estado mental como lo son Addenbrooke-III (ACE-III), Addenbrooke-R (ACE-R), INECO Frontal Screening (IFS). Al mismo tiempo brinda retroalimentación y estrategias que logren un cerebro activo, de forma que disminuya los problemas de memoria principalmente.

## **Marco de desarrollo de software aplicada**

Scrum es un marco que reduce la complejidad de desarrollo, permite resolver problemas imprevisibles gracias a la autoorganización del equipo y satisface las necesidades del cliente, asegurando un producto de calidad.

### **Roles de usuario**

#### *Familiar*

- Conocimiento en tecnologías básico.
- Conocimiento del padecimiento básico.
- El paciente puede interactuar con el sistema.
- Puede visualizar los resultados del examen cognitivo del paciente.

### **Identificación de requisitos**

SCRUM marca una forma para identificar requisitos. A continuación se presentan las historias de usuario.

### **Product Backlog**

## **ESPECIFICACIÓN DE REQUERIMIENTOS**

El presente documento describe la especificación de requerimientos de la aplicación para el proyecto titulado "ALZHEIMER SOFTWARE.

### **1.1 Propósito del Documento**

Especificar de forma única y completa los requerimientos con que deberá cumplir el proyecto, con la finalidad de ser una referencia para los grupos de relación involucrados en el mismo, como son: encargado del proyecto,

desarrolladores, usuarios finales; tomando importancia dicho documento para la toma de decisiones relevantes dentro del seguimiento del proyecto.

## **1.2 Alcance del Documento**

El presente documento considera los elementos necesarios para la operación del proyecto el cual integra una parte general:

Se desea crear una aplicación móvil que permita a través de los exámenes Addenbrooke-III (ACE-III), Addenbrooke-R (ACE-R) e INECO Frontal Screening (IFS) conocer el nivel de deterioro cognitivo de una persona que padece Alzheimer y a la vez presentar una serie de ejercicios como completar palabras, memorizar palabras, nombrar imágenes, que minimicen el impacto.

## **2. Características del Usuario**

Los usuarios finales son personas de 3ra edad que presentan la enfermedad de alzheimer

Para la utilización de la aplicación, es necesario contar con un familiar que pueda apoyar al paciente al momento de usar un dispositivo móvil

## **3. Requisitos de la Interfaz Externa**

### **3.1 Interfaces del usuario**

A continuación, se muestran las interfaces de usuario requeridas para el desarrollo y uso de la aplicación alzheimer software.

### **3.2 Interfaces del hardware**

Uno de los requerimientos para el uso de la aplicación. es agilizar la memoria del paciente permitiendo realizar ejercicios que le permita recordar y mejorar su deterioro cognitivo.

El esquema de comunicación de AS(Alzheimer Software )para tener acceso y uso a la aplicación AS funciona bajo el modelo cliente-android , para lo cual es necesario contar con un dispositivo móvil que permita alojar al sistema operativo móvil.

### **3.3 Interfaces del software**

las interfaces del software se muestran en la siguiente tabla donde se especifican todos los productos requeridos para el desarrollo del

producto como el manejo de android studio, dispositivo móvil y entre aquellas aplicaciones que interactúan en el funcionamiento del producto.

Nombre	Nemónico	Versión	Fuente
android studio		3.6.3	developers
MockFlow			
Bizagi modeler		2.0	BPMN
Balsamiq Project		3.0	Balsamiq
StarUML			StarUMLink

**Tabla 1. Productos de software requeridos**

## **4. Características de la aplicación**

### **4.1 Descripción general**

La presente aplicación busca poder mantener las funciones cognitivas el mayor tiempo posible y así mejorar la calidad de vida de los mayores a través de uso de un dispositivo móvil.

Para poder ingresar el usuario (Familiar) deberá de contar con un registro de usuario y previamente entra con ella misma, al estar dentro de la aplicación el usuario podrá elegir la sección con la que el paciente pueda trabajar y api mismo la aplicación permitirá realizar una pruebas para tener conocimiento y del avance del paciente.

### **4.2 Requisitos funcionales**

Esta sección incluye los requerimientos que especifican todas las acciones fundamentales del sistema. Se empieza especificando la clase de usuario al cual va dirigida la primera sección de requerimientos seguido de una lista de los mismos.

- La aplicación debe de almacenar la información en la base datos
- La aplicación debe de permitir el registro, visualización del contenido, selección de sección.

- El sistema debe de permitir el ingreso a los usuarios.
- El aplicación debe de almacenar el avance del paciente posterior a sus resultados.
- El usuario (Familiar )controlará el acceso.

## **4.3 Requisitos no funcionales**

### **4.3.1 Requerimientos del producto**

Alzheimer Software estará disponible en celulares android.

### **4.3.2 Requerimientos de usabilidad**

Se manejarán manuales y guías dentro de la aplicación.

### **4.3.3 Requerimientos de confiabilidad**

Toda la información será respaldada diariamente con la finalidad de no perder la continuidad del sistema en caso de ser necesario realizar modificaciones o reparaciones a un proceso.

### **4.3.4 Requerimientos de seguridad**

Para mantener la seguridad de los datos se restringen ciertas operaciones a los usuarios cómo eliminar información o modificar el tipo de tratamiento.

### **4.3.5 Requerimientos de la organización**

Los usuarios de Alzheimer Software podrán acceder al sistema a través de una cuenta creada por ellos junto con una contraseña

## **5. Historias de usuario**

Las historias de usuario son pequeñas descripciones de los requerimientos de un cliente.

Especificaciones funcionales que invitan a la conversación para que el detalle de los requisitos del sistema sea más eficaz.

Redactadas bajo el formato “Yo como (rol) necesito (funcionalidad) para (beneficio)”.

- Yo como usuario quiero saber el contenido de cada sección para evaluar las capacidades cognitivas del paciente

- Yo como usuario deseo saber las instrucciones de la aplicación
- Yo como usuario necesito entrar al sistema para activar el banco de recuerdos y entrenar al paciente para que mejore en los resultados
- Yo como usuario deseo ver los ejercicios mentales que debo utilizar con el paciente
- Yo como usuario deseo conocer los resultados del examen cognitivo

## 6. Estimaciones

Calcular la tabla de estimación del equipo.

- a) Construir la tabla de forma individual. Recuerden que los valores deben ser del programa más pequeño y más grande que han hecho en el lenguaje de programación que van a usar para su proyecto

**Tabla 1**

El programa más <b>pequeño</b> tiene <b>47 LOC</b>	El programa más <b>pequeño</b> tiene <b>188 LOC</b>
El programa más <b>grande</b> tiene <b>522 LOC.</b>	El programa más <b>grande</b> tiene <b>743 LOC.</b>

**Tabla 2**

Divide este rango entre conjuntos iguales en una escala logarítmica

<ul style="list-style-type: none"> <li>• Tomar logaritmo <b>base 10</b></li> <li>• El más pequeño 47 da <b>1.672</b></li> <li>• El más grande 522, da <b>2.718</b></li> </ul>	<ul style="list-style-type: none"> <li>• Tomar logaritmo <b>base 10</b></li> <li>• El más pequeño 188 da <b>2.274</b></li> <li>• El más grande 743, da <b>2.8709</b></li> </ul>
---	---

Para conseguir 5 categorías, dividir la diferencia entre 4

$2.718 - 1.672 = 1.046 / 4 = \mathbf{0.2615}$	$2.8709 - 2.274 = .5969 / 4 = \mathbf{0.1492}$
---	--

Este valor es el incremento al logaritmo para cada categoría

**Tabla 1**

**Tabla 2**

1.672	1.9335	2.195	2.4565	2.718	2.274	2.4232	2.5724	2.7216	2.8709
47	86	157	286	522	188	265	374	527	743

Resultado de las tablas individuales

**Tabla 1**

**Tabla 2**

	Log10(LOC)	LOC
Very Small	1.672	47
Small	1.9335	86
Medium	2.195	157
Large	2.4565	286
Very Large	2.718	522

	Log10(LOC)	LOC
Very Small	2.274	188
Small	2.4232	265
Medium	2.5724	374
Large	2.7216	527
Very Large	2.8709	743

Resultado de la tabla promedio con nuevos valores

	Log10(LOC)	LOC
Very Small	2.071	118

Divide este rango entre conjuntos iguales en una escala logarítmica

- Tomar logaritmo **base 10**
- El más pequeño **118** da **2.071**
- El más grande **633** da **2.801**



Small	2.2535	176
Medium	2.436	266
Large	2.6185	407
Very Large	2.801	633

Para conseguir 5 categorías, dividir la diferencia entre 4				
$2.801 - 2.071 = .73 / 4 = 0.1825$				
Este valor es el incremento al logaritmo para cada categoría				
2.071 118	2.2535 176	2.436 266	2.6185 407	2.801 633

Estimar cuánto tiempo les va a requerir acabar con su proyecto. Calcule el promedio de programación LOC. Por ejemplo, un programador experimentado produce 1000 LOC/Mes con un promedio de horas de trabajo es de 170 horas al mes hace 6 LOC/Hora. Recuerda que esto incluye Documentación, análisis, diseño y pruebas.

Size	Use Cases that fit this category	Hours Required
Very Small	Entrar al sistema	8
Small	Investigación sobre la enfermedad	8
	Documentación	8
	Diseño de cronograma	8
	Conocer el contenido	8
Medium	Investigación en el desarrollo del entorno planteado	16
	Diseño de base de datos	16
	Diseño de interfaces	16
	Conocer las instrucciones de la aplicación	16
Large	Ver los ejercicios mentales	32
	Interactuar las secciones	32
Very Large	Conocer los resultados cognitivos	56
	Pruebas	56
	Total	280 Horas

**Planning Poker**  
**Método de Estimación**  
**Ponderación (Valor Fibonacci 1-13)**

Tareas	Estimación Planning Poker	Tiempo
Investigación sobre la enfermedad	3	1 día
Investigación en el desarrollo del entorno planteado	5	2 días
Documentación	3	1 día
Diseño de cronograma	3	1 día
Diseño de interfaces	5	2 días
Diseño de base de datos	5	2 días
Entrar al sistema	1	1 día
Conocer las instrucciones de la aplicación	5	2 días
Conocer el contenido	3	1 día
Ver los ejercicios mentales	8	4 días

<b>Interactuar las secciones</b>	<b>8</b>	<b>4 días</b>
<b>Conocer los resultados cognitivos</b>	<b>13</b>	<b>7 días</b>
<b>Pruebas</b>	<b>13</b>	<b>7 días</b>
<b>Total</b>	<b>75</b>	<b>35 días</b>

### **Justificación:**

Consideramos la técnica de planning poker ya que nos permite calcular el tamaño relativo basado en las tareas para el desarrollo del software, entonces tomamos en cuenta el uso del fibonacci para cada tarea con un grado de relevancia, es de decir, 1, 3,5,8,13 con un promedio de 8 horas de trabajo diarias contando fines de semana x un mes = 224 horas. Esta estimación corresponde a la documentación, análisis y diseño.

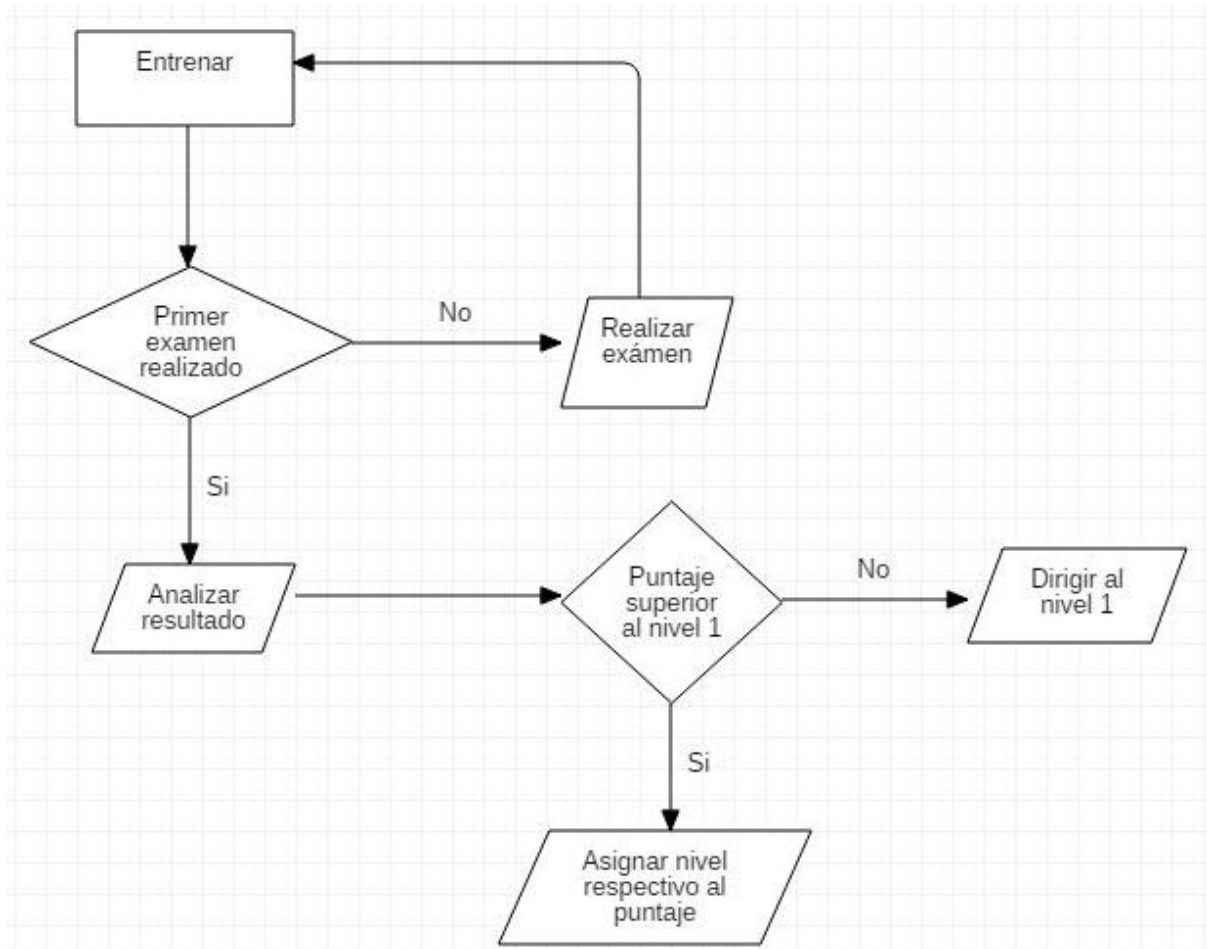
Para contemplar las pruebas con usuarios, se calcula una semana extra a la estimación anterior (56 horas), contemplando en caso de existir retrasos en alguna etapa o correcciones finales. Por lo tanto se estima un tiempo total de 280 horas para la realización del proyecto.

## 7. Diseño de diagramas

### Arquitectura antes del diseño



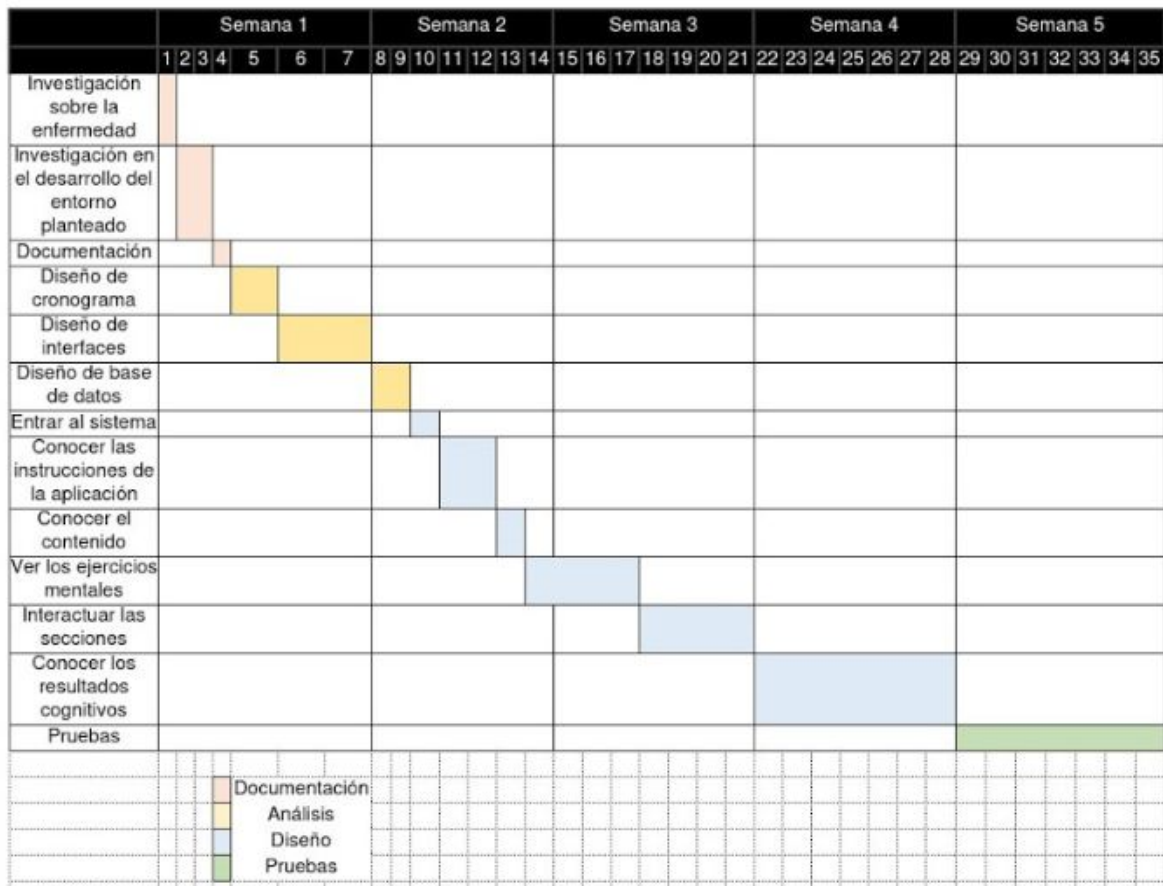
## Diagrama de flujo



## Diagrama de distribución



## Diagrama de Gantt



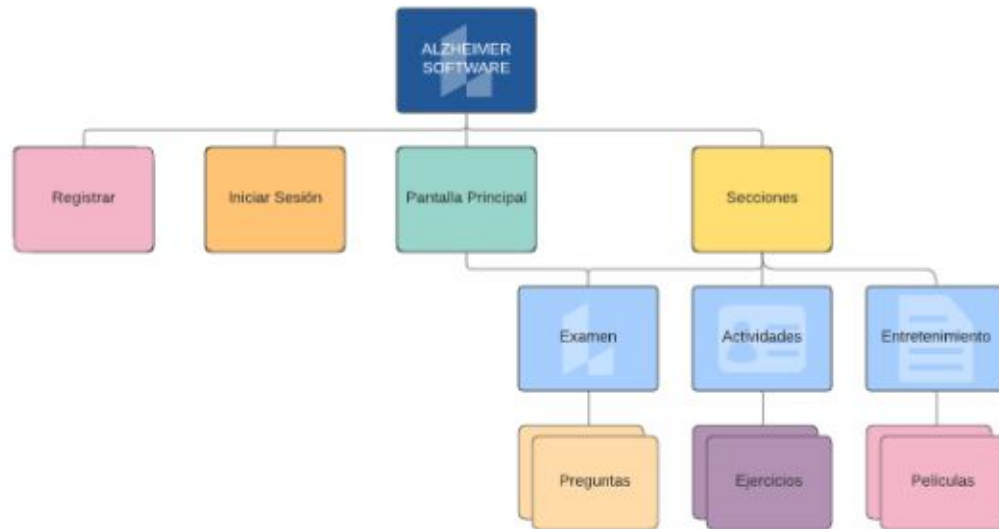
## Diagrama Venn



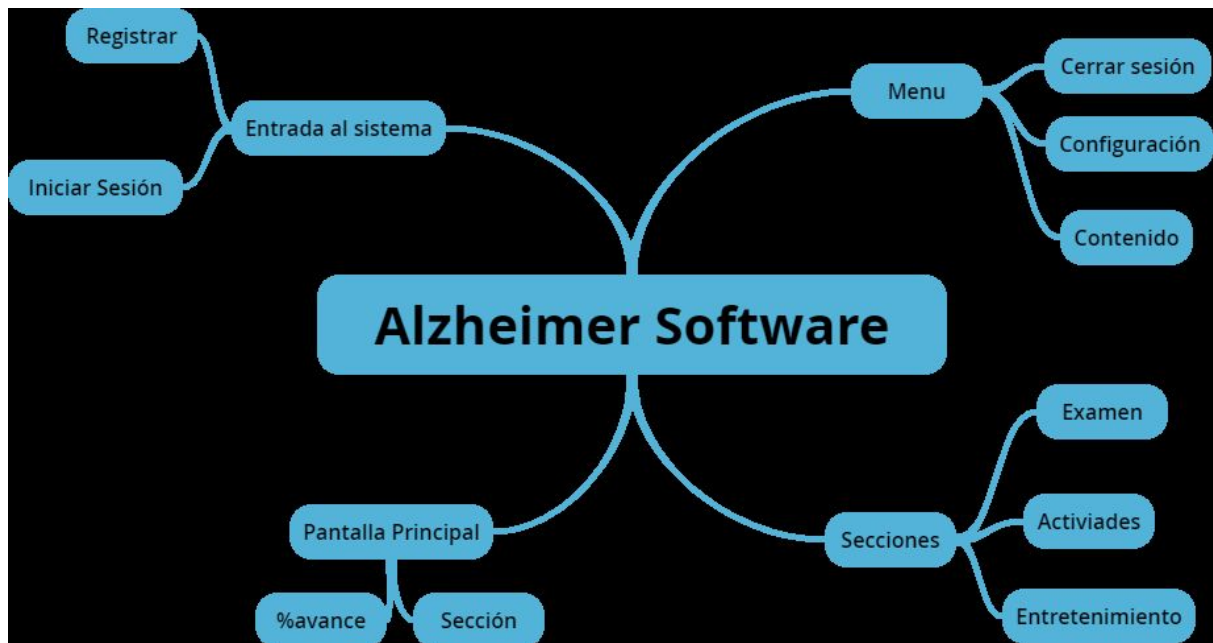
## Diagrama de Jerarquías

## ALZHEIMER SOFTWARE

cdlali morales | April 29, 2020

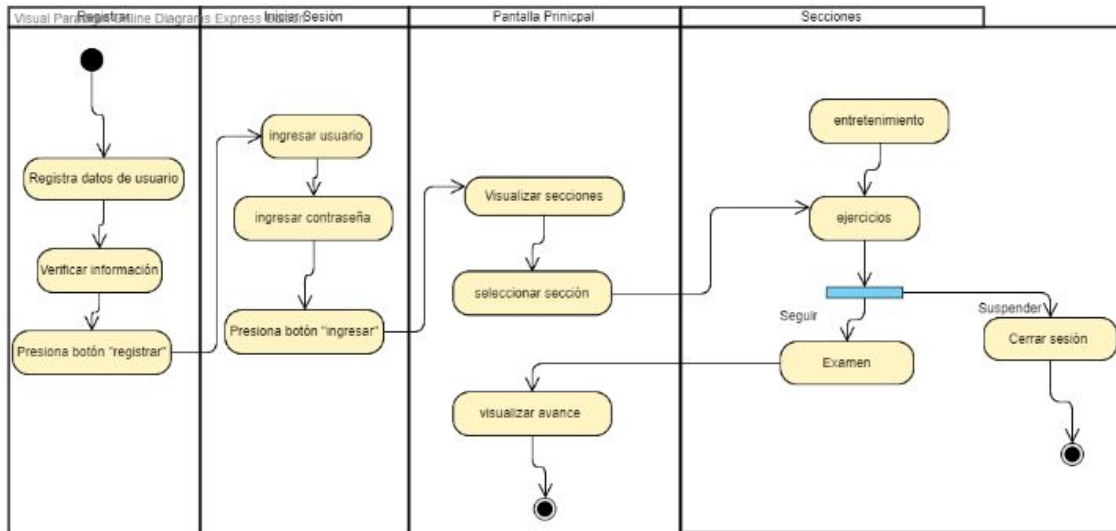


## Mapa Conceptual

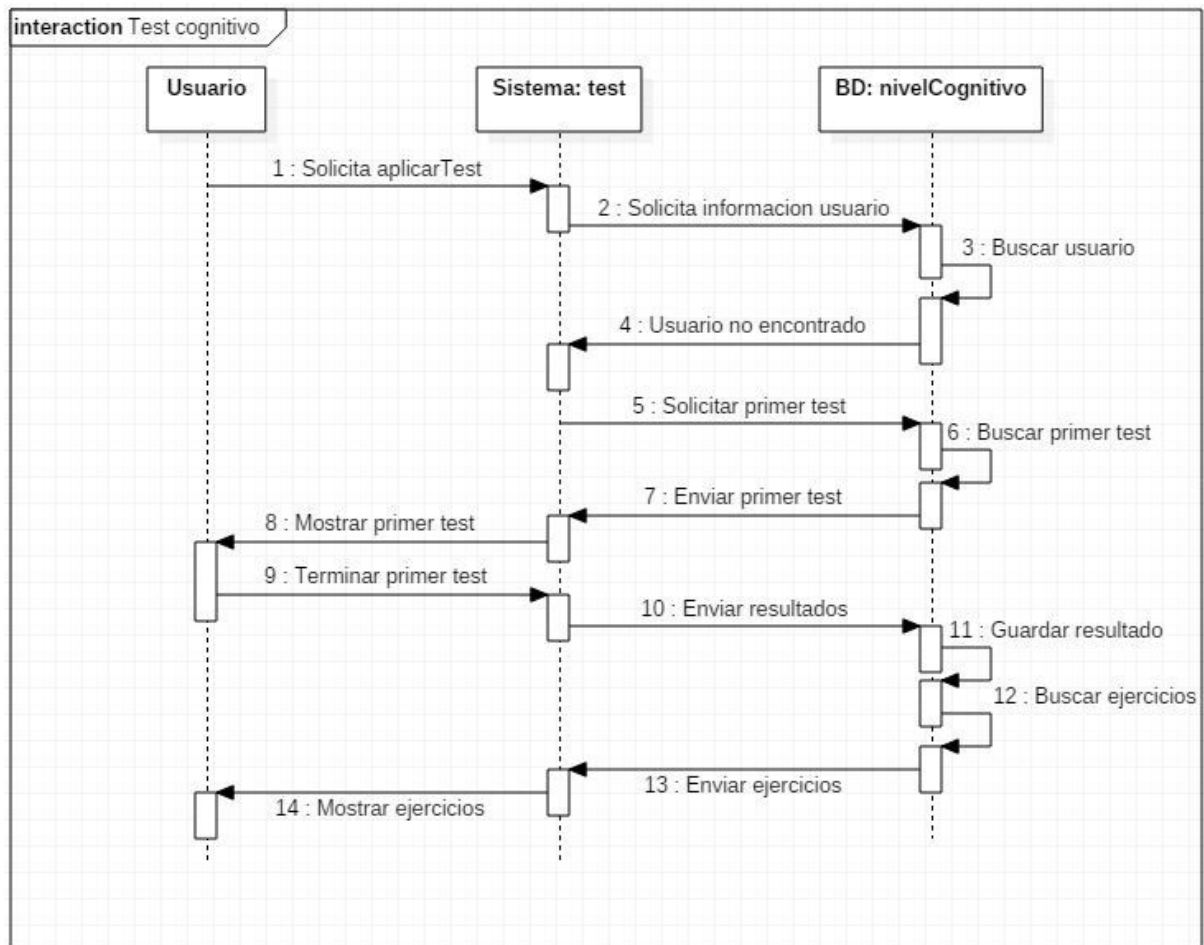


## Diagrama de actividades

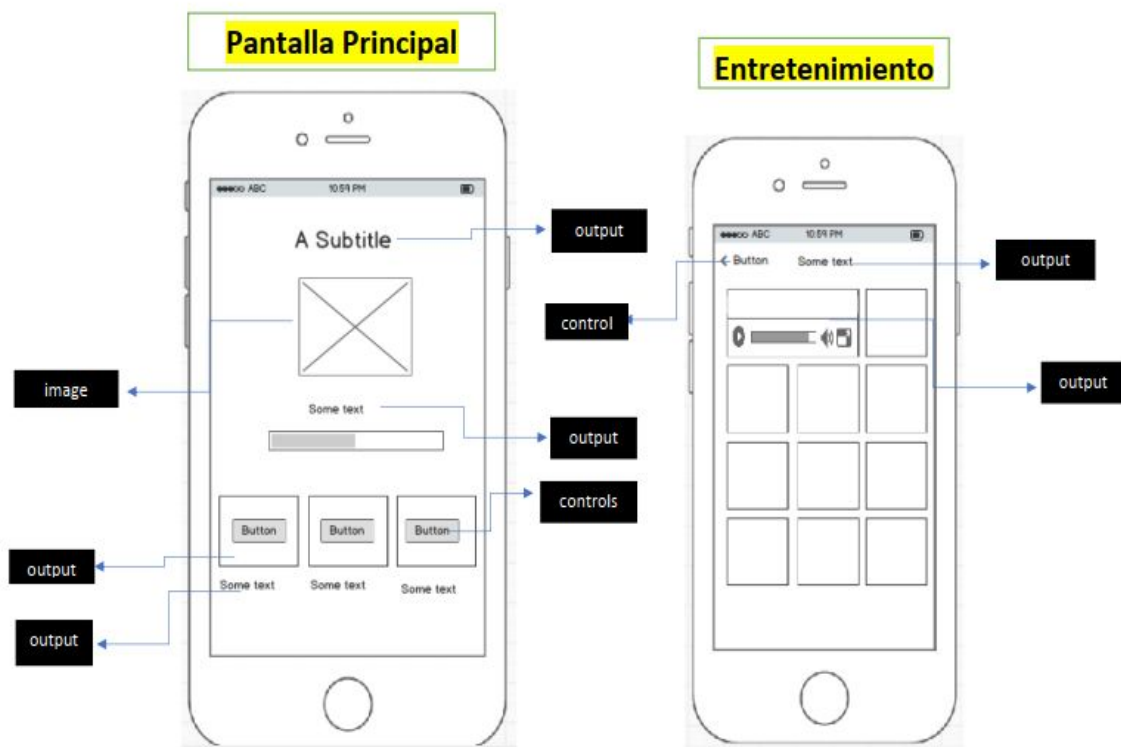
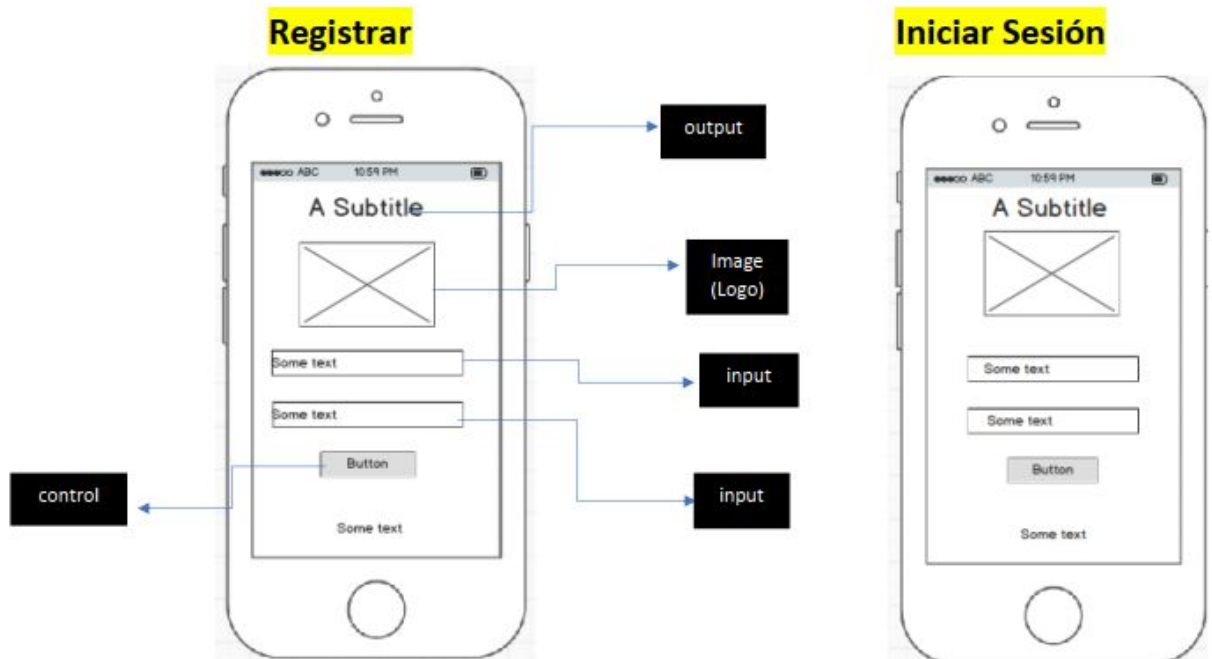




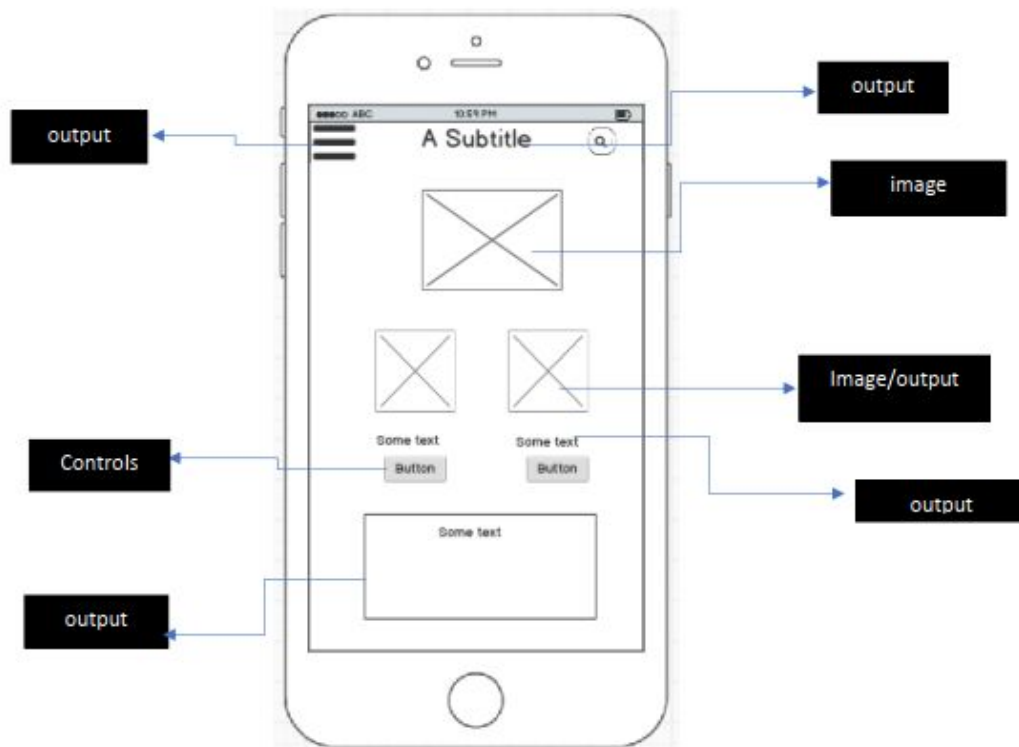
## Diagrama de secuencia



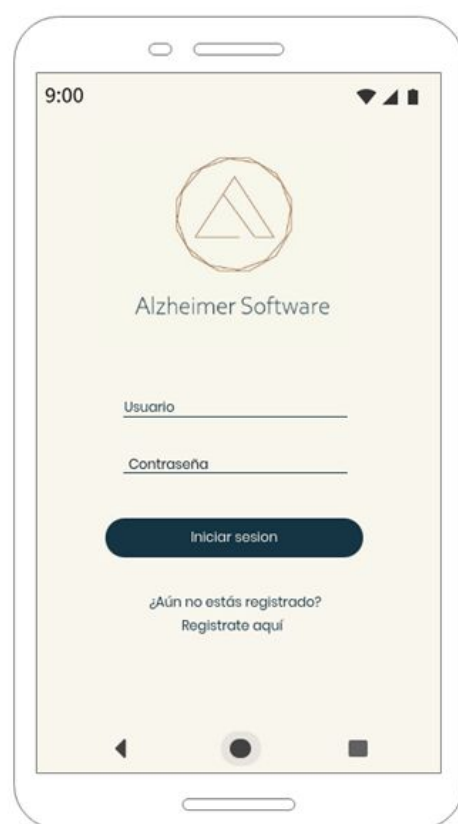
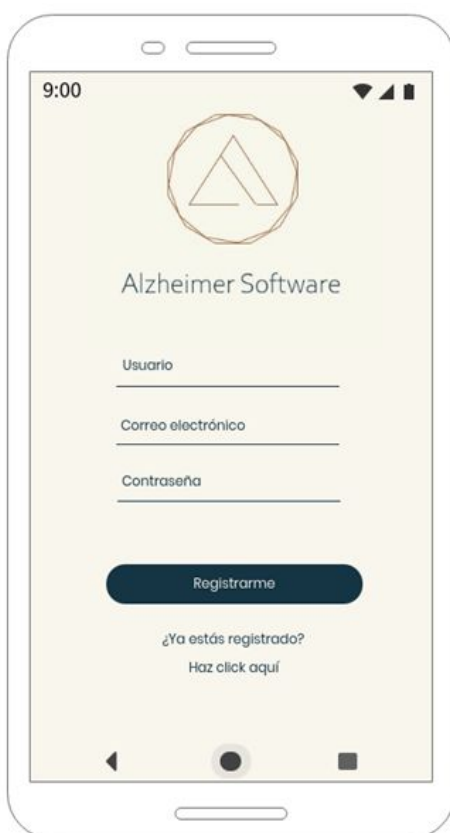
## Interfaces abstractas



## Actividades



## Interfaces concretas



## 8. Conclusiones

El uso de las tecnologías podrá facilitar el tratamiento de las personas con este padecimiento, debido a que las personas que tienen la responsabilidad de ver por ellos en muchas ocasiones se vuelve una actividad pesada y se termina perdiendo el rumbo de lo que se quiere lograr: ayudar a una persona con trastorno del desarrollo intelectual (TDI).

## 9. Referencias

Instituto Gerontológico, 2018,

Nueve enfermedades neurológicas comunes en los adultos mayores

<http://www.liderdoctor.es/2018/01/29/nueve-enfermedades-neurologicas-comunes-en-los-adultos-mayores/>

alzheimer's association, 2019,

Hechos y cifras sobre la enfermedad de Alzheimer de 2019,  
<https://www.alz.org/alzheimer-demencia/hechos-y-cifras>

Instituto nacional de Neurología y Neurocirugía

Enfermedad de Alzheimer

<http://www.innn.salud.gob.mx/interna/medica/padecimientos/alzheimer.html>