



Benemérita Universidad Autónoma de Puebla

Facultad de Ciencias de la Computación

Sistema Web de Ejercicios Visuales

Tesis que para obtener el título de Ingeniero en Ciencias de la Computación

Presenta

Víctor Castillo Tomas

Asesor

Dr. Pedro Bello López

Marzo, 2023

Agradecimientos a Mi Padre y Madre por enseñarme lo que hace el esfuerzo!!

Contenido

Capítulo 1 . Introducción	5
1.1 Propósito.....	5
1.2 Alcance.....	6
1.3 Objetivos Generales y Específicos del Proyecto.....	7
1.4 Estado del Arte	7
1.4. Organización de la tesina	12
Capítulo 2. Marco Teórico	13
2.1 Ingeniería de software.....	13
2.2 Bases de datos.....	15
2.3 Herramientas de desarrollo web.....	17
Capítulo 3. Análisis y diseño del sistema.....	18
3.1. Requerimientos funcionales y no funcionales	18
Reglas de negocio:.....	18
Requerimientos funcionales:	20
Requerimientos no funcionales:	20
3.2 Casos de uso.....	21
3.3 Diagrama de clases	35
3.4. Diagrama entidad relación del sistema	38
3.5. Diagrama Relacional del Sistema	40
Capítulo 4. Implementación y pruebas del sistema.....	47
4.1 Implementación de la base de datos.....	47
4.2 Implementación de las funcionalidades del sistema.....	51
4.3 Maquetado y pruebas de funcionalidad del sistema	53
Conclusiones y trabajo futuro.....	66
Bibliografía.....	67

Lista de Imágenes

IMG 1 LA APP DE ESTIMULACIÓN COGNITIVA (HTTPS://WWW.IMENTIA.COM/)	10
IMG 2 MEMORY BOX (HTTPS://WWW.DEMENTIAFORUM.ORG/)	10
IMG 3 MEMORY BOX (HTTPS://WWW.DEMENTIAFORUM)	11
IMG 4 CASOS DE USO DEL SISTEMA V1 PROPIO	22
IMG 5 CASOS DE USO DEL SISTEMA V2 PROPIO	23
IMG 6 DIAGRAMA DE CLASES V1 PROPIO	36
IMG 7 DIAGRAMA DE CLASES V2 PROPIO	37
IMG 8 DIAGRAMA ENTIDAD RELACIÓN PROPIO	39
IMG 9 DIAGRAMA RELACIONAL V1 PROPIO	42
IMG 10 DIAGRAMA RELACIONAL V2 PROPIO	43
IMG 11 TABLA USUARIO PROPIO	44
IMG 12 CATALOGO TIPOUSUARIO PROPIO	44
IMG 13 DIAGRAMA RELACIONAL V3 PROPIO	45
IMG 14 CREACIÓN DE LAS TABLAS DE LA BASE DE DATOS DEL SISTEMA WEB PROPIO	47
IMG 15 DESCRIPCIÓN DE LAS TABLAS EJERCICIOS E IMÁGENES PROPUESTAS PARA LA BASE DE DATOS DEL SISTEMA WEB. PROPIO	49

Capítulo 1 . Introducción

Se explica de manera general las características y motivos principales de la creación de un sistema web que contenga ejercicios visuales para personas con problemas de memoria débil, primeras etapas de Alzheimer. El sistema web podrá registrar usuarios con 3 tipos de roles: Tutor, Paciente y Administrador. Donde el Tutor podrá registrar a su Paciente para llevar un seguimiento del posible avance de su enfermedad, agregar, modificar y eliminar fotos/ imágenes al sistema y saber su estatus actual. El administrador dará de alta, baja o modificación de tutores y pacientes. El Paciente podrá realizar ejercicios de memorización y reconocimiento de las imágenes cargadas por su tutor. El motivo inicial de este sistema es poder ayudar a tener un seguimiento y evaluación del estatus general de un paciente, respecto a su capacidad para reconocer o no imágenes familiares, y turnar esta información a algún profesional de la salud para su apoyo durante el tratamiento/diagnóstico de su paciente.

1.1 Propósito

Apoyar a la comunidad es parte importante de la convivencia en sociedad, además de que los problemas sociales nos conciernen a todos de manera directa o indirecta en mayor o en menor cantidad.

En México más de 350,000 personas están afectadas por la enfermedad de Alzheimer y mueren por ella anualmente 2,030 pacientes. En Estados Unidos hay 4 millones y mueren anualmente más de 100,000, convirtiendo a la enfermedad de Alzheimer en la cuarta causa de muerte entre adultos. Se estima que uno de cada tres de nosotros enfrentará esta enfermedad en algún ser querido o en un familiar [1].

Al pasar de los años nuestros cuerpos pueden llegar perder funciones vitales de manera rápida o paulatina, dependiendo de diversos factores, como calidad de vida, enfermedades crónicas o hereditarias etc.

Nuestras funciones mentales son de las más apreciadas por todos, en el caso de comenzar a perderlas genera una gran inestabilidad tanto para la persona que las pierde como las personas que lo rodean.

La demencia es considerada el termino más común utilizado para describir un Deterioro en la habilidad mental, lo suficientemente fuerte para interferir en la vida

diaria. Un ejemplo de ello es la pérdida de memoria. El tipo de demencia más común es el Alzheimer [2].

El Alzheimer es una enfermedad neuro degenerativa que afecta a nuestra sociedad, en más cantidad a los adultos mayores.

El Alzheimer presenta de manera general 3 etapas: leve, moderada y severa.

Etapas leve:

En esta etapa, el paciente presenta déficits para la memoria reciente, desorientación temporal y dificultades para realizar tareas complejas. También puede presentar síntomas emocionales, como ansiedad o depresión y negación de la enfermedad. El paciente puede mantener suficiente autonomía como para vivir solo sin ayuda o realizar un trabajo relativamente simple.

Etapas moderada:

En esta etapa el paciente presenta alteraciones graves en la memoria y en la orientación. Aumenta los problemas de comunicación y las reacciones emocionales se acentúan. Aumenta la dependencia en las actividades de la vida diaria, tales como vestirse y asearse.

Etapas severa:

En esta etapa el paciente presenta desorientación total, con severos problemas de comunicación. Muchos de los pacientes presentan gran agitación, trastornos del sueño, comportamientos repetitivos e incontinencia esfinteriana. El paciente es dependiente para todas actividades diarias [3].

1.2 Alcance

En el proyecto aquí presentado se propone generar un sistema web el cual se centrará en generar ejercicios visuales para el reforzamiento de la memoria en pacientes con Alzheimer en etapa leve.

Apoyando al reporte de los resultados generados por el paciente cuando contesta una serie de estos ejercicios.

Los ejercicios visuales serian una serie de preguntas asociadas a imágenes. Las imágenes serán objetos, lugares y fotos familiares que el tutor del paciente cargara al sistema. Las preguntas serán de opción múltiple asociadas a la imagen que se le muestre en ese momento al paciente.

1.3 Objetivos Generales y Específicos del Proyecto.

Objetivo General:

Diseñar e implementar un sistema web que permita mostrar, reportar y operar ejercicios visuales para personas con problemas de memoria débil.

Objetivos Específicos:

- Realizar la especificación de requisitos de acuerdo con la metodología especificada [4].
- Diseñar la base de datos y el diseño del sistema web [5].
- Crear una interfaz amigable para la ejecución y consulta de ejercicios visuales [6].
- Realizar pruebas del funcionamiento general del sistema web.

1.4 Estado del Arte

En la actualidad existen algunas técnicas que nos ayudan a socavar en cierta medida esta enfermedad como las citas a continuación:

Técnicas para mejorar la memoria:

- El trabajo se orienta a ejercitar y mejorar tanto la memoria reciente, como favorecer el mantenimiento de recuerdos más lejanos.
- Repetición de números o palabras para ejercitar la memoria inmediata.
- Juegos de memoria con tablillas de imágenes distintas.
- Juegos de memorización de palabras o frases.
- Ejercicios para recordar acontecimientos y noticias.
- Recordar hechos cotidianos como la comida o lo que se hizo el día anterior.
- Apoyarse en material visual y verbal, como una canción o frase hecha, para evocar recuerdos remotos.

Técnicas para estimular la orientación:

- Tanto para mejorar el conocimiento que el enfermo de Alzheimer tiene sobre el ámbito espacio temporal en el que se desenvuelve, como los datos sobre la propia persona.
- La técnica más sencilla es la de realizar al paciente preguntas básicas para mejorar la orientación en tres sentidos: temporal, espacial y personal.
- Orientación temporal: preguntas sobre el mes, año, día o estación en la que nos encontramos o por festivos determinados.
- Orientación espacial: cuestiones sobre la ciudad, el barrio o incluso el domicilio en el que se encuentra.
- Orientación personal: sobre el nombre del enfermo, su fecha de nacimiento o incluso si está casado.
- Esta técnica se refuerza mediante el uso de ayudas gráficas como calendarios, relojes o mapas, para ayudar así también a mejorar el nivel de atención.

Técnicas para estimular el lenguaje:

- Para optimizar la capacidad verbal del paciente debe trabajarse tanto en el lenguaje oral como escrito. Los ejercicios encaminados a estimular esta área son:
- Descripción de imágenes, hechos, láminas o situaciones.
- Ejercicios de lectura y escritura. Mediante dictados, copias o descripciones. Centrándose en frases, palabras o sílabas.
- Asociación de palabras y complementar familias de palabras.
- Ejercicios de repetición de palabras, letras, frases o números. La repetición es más importante cuanto mayor es el deterioro del paciente.
- Denominación de los objetos cotidianos, incluso de la comida que hay en la mesa o el nombre de familiares.

Técnicas para mejorar las gnosias:

- Se debe ejercitar la percepción y reconocimiento tanto de formas como de características físicas de personas y objetos, de forma visual, táctil y auditiva. Algunos ejercicios a tener en cuenta son:
- Reconocimiento de letras y números en relieve.
- Tocar objetos cotidianos para reconocerlos.

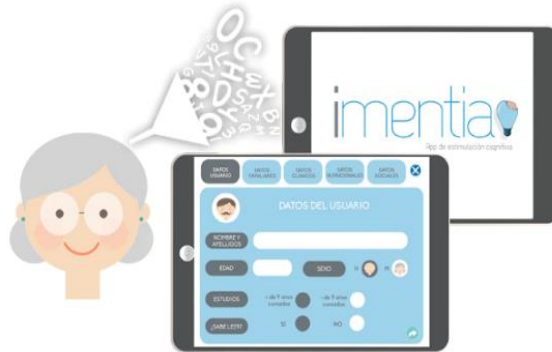
- Reconocer e imitar sonidos. Realizar sonidos de objetos cotidianos o animales, por ejemplo.
- Reconocimiento facial de familiares, conocidos y de personajes famosos.
- Reconocimiento de colores. Nombre de los colores, asociar los objetos con su color tradicional.
- Reconocer imágenes gráficas, mediante asociación de imágenes y palabras, fotografías, letras, números. Apoyarse en láminas para el reconocimiento de objetos cotidianos.

Técnicas para mejorar las praxias:

- Para trabajar las capacidades praxias del enfermo de Alzheimer se debe aprovechar sus gustos personales y aficiones y buscar tareas aplicables a su vida diaria.
- Unir puntos numerados para completar un dibujo.
- Utilizar objetos cotidianos, dárselos al paciente para que los nombre, diga para qué sirven y demuestre su uso.
- Pedir que dibuje elementos cotidianos como un teléfono, un árbol o una casa. [7]

En su mayoría estas técnicas llegan a requerir un especialista de la salud que las aplique o que las evalúe. Que es donde interviene nuestra participación como afines al desarrollo de Software, ya que actualmente contamos con ya algunos acercamientos de desarrollo de software que están encaminadas a dar apoyo para controlar o monitorear esta enfermedad.

En el mercado existen algunas ofertas como son:

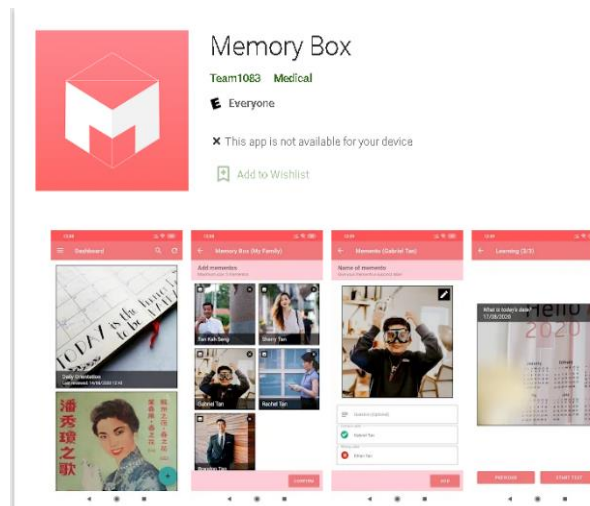


Img 1 La app de estimulación cognitiva (<https://www.imentia.com/>)

Imentia (Img 1) es una app de estimulación cognitiva basada en una serie de ejercicios interactivos que entrenan las distintas áreas cognitivas.

Las sesiones de estimulación cognitiva trabajan todas las áreas neurocognitivas: memoria, orientación, lenguaje, atención, razonamiento, gnosias, praxias, funciones ejecutivas y comprensión. Cada área de trabajo tiene hasta seis niveles de complejidad.

La cual tiene un costo de 24 euros, 9 euros y gratuita, de donde cada una de estas versiones difieren por ser mono o multiusuario, mono o multi expediente y con sesiones muy generales o personalizadas. Hay versiones en español e inglés.



Img 2 Memory box (<https://www.dementiaforum.org/>)

Memory box (Img 2): Es un software, una aplicación móvil, que permite guardar imágenes, recuerdos y genera conversaciones con el usuario referentes a sus imágenes generada por la fundación Dementia forum

Está disponible para Android y iPhone, solo que se encuentra con soporte solo en inglés y recientemente Chino, el cual tiene su versión gratuita.

Senda: es otro rubro de herramientas ya que este software está más dirigido a la localización de la persona con Alzheimer y no al refuerzo de la memoria como las antes indicadas.



Img 3 Memory box (<https://www.dementiaforum>)

Esta herramienta de software (Img 3) también incorpora un botón de pánico S.O.S en caso de ser necesario. La cual tiene soporte en español, pero solo es para sistemas iOS. Ya que para sistemas Android no está disponible o esta desactualizada.

Existen varios sistemas en el mercado, de los cuales la mayoría son de pago y en inglés, además de tratar la enfermedad de la memoria en etapas más avanzadas, la idea general de nuestra propuesta de sistema web para ejercitar la memoria, es un seguimiento en etapas tempranas de personas con problemas de memoria débil. Este sistema será gratuito y estará disponible a través de internet.

1.4. Organización de la tesina

Este proyecto de tesina está organizado considerando una introducción general donde se describe el proyecto, los alcances, los objetivos y el estado del arte. En el capítulo 2 se presenta el marco teórico metodológico necesario para el desarrollo del trabajo. En el capítulo 3 se presenta el análisis y el diseño del sistema en base a la metodología descrita. En el capítulo 4 se presenta la implementación del prototipo del sistema, así como las principales pruebas funcionales y de integración del sistema web. Al final del documento se presenta las conclusiones del trabajo dando énfasis en el trabajo que a futuro se puede desarrollar tomando como referencia el trabajo aquí presentado.

Capítulo 2. Marco Teórico

2.1 Ingeniería de software

Sobre la Ingeniería de software, muchas o varias necesidades propician a la generación de soluciones o soluciones, modificaciones en coste de hardware y o software, mantenimiento del software, avances en hardware y software más complejo son algunas de las áreas de oportunidad que dan pie a la Ingeniería de Software.

El nombre de "Ingeniería de Software" acunado de inicio en 1968 en una conferencia de la OTAN [8], pero en inicio podríamos hablar de Software, el cual entre tantas definiciones podríamos expresarlo como:

Software is: (1) instructions (computer programs) that when executed provide desired features, function, and performance; (2) data structures that enable the programs to adequately manipulate information, and (3) descriptive information in both hard copy and virtual forms that describes the operation and use of the programs (El software es: (1) instrucciones (programas de computadora) que cuando se ejecutan proporcionan las características, la función y el rendimiento deseados; (2) estructuras de datos que permiten a los programas manipular adecuadamente la información, y (3) información descriptiva tanto en forma impresa como virtual que describe el funcionamiento y el uso de los programas) [9]

La palabra ingeniería definirla de la siguiente manera:

"Engineering is 'the art or science of making practical applications of the knowledge of pure sciences.' In other words, although engineers are not scientists, they study the sciences and use them to solve problems of practical interest, most typically .by the process that we call creative design" (La ingeniería es "el arte o la ciencia de hacer aplicaciones prácticas del conocimiento de las ciencias puras". En otras palabras, aunque los ingenieros no son científicos, estudian las ciencias y las utilizan para resolver problemas de interés práctico, más típicamente por el proceso que llamamos diseño creativo.) [10].

"Engineering is often defined as the use of scientific knowledge and principles for practical purposes usually by designing useful products"(La ingeniería a menudo se define como el uso del conocimiento científico y los principios para la práctica. Propósitos.... generalmente diseñando productos útiles) [11]

Ahora como definiciones de la ingeniera de software:

Software engineering is an engineering discipline that is concerned with all aspects of software production from the early stages of system specification through to maintaining the system after it has gone into use. (La ingeniería de software es una disciplina de ingeniería que se ocupa de todos los aspectos de la producción de software desde las primeras etapas de la especificación del sistema hasta el mantenimiento del sistema después de que ha entrado en uso.) [12]

En la creación de Software se debe llevar un control sobre una serie de actividades, las cuales corresponden a una metodología, actividades que nos llevan de la general a lo particular; para así satisfacer la necesidad del usuario con un producto de calidad el cual se adapta los requerimientos solicitados.

Hay varias metodologías de Software, utilizamos las etapas principales del ciclo de vida clásico del desarrollo de software donde se consideran las siguientes etapas:

1. Especificación del software. Donde se definen la funcionalidad del software y sus restricciones.
2. Desarrollo del software. Se produce el software que cumple con las especificaciones.
3. Validación del software. Se debe asegurar que el software cumple con lo que el cliente espera.
4. Evolución del software. El software debe evolucionar para cumplir con las necesidades cambiantes del cliente [13].

Para el caso de nuestra implementación usaremos específicamente la metodología RUP.

La implementación de la metodología RUP se recomienda cuando el proyecto tiene una alta complejidad técnica y una alta complejidad Gerencial.

En nuestro caso no tiene una alta complejidad gerencial, pero si un nivel medio en la complejidad técnica. Además de la experiencia propia sobre la implementación de esta metodología.

Que es RUP:

Es un proceso de desarrollo de software que por medio de los requisitos de usuario ejecutaremos un conjunto de actividades para obtener un sistema software.

“RUP está dirigido por casos de uso, centralizado en la arquitectura y es iterativo e incremental” [14]

Plan general del proyecto, esto lo llevaremos a cabo conducido por casos de uso de manera iterativa incremental:

- Con el modelo de casos de uso obtendremos los requisitos funcionales.
- Con nuestros casos de uso obtendremos nuestro modelo del análisis.
- Con nuestro modelo de análisis obtendremos nuestro modelo de diseño.
- Con nuestro modelo de diseño obtendremos nuestro modelo de implementación

Para así continuar con las pruebas sobre nuestros casos de uso Todo esto de manera iterativa e incremental durante el ciclo de vida del sistema, donde cada ciclo tendrá como fases:

- Inicio o factibilidad, elaboración, Construcción, Transición.
- Inicio: Alcancel del proyecto
- Elaboración: Planificación del proyecto
- Construcción: Construcción del sistema de SW.
- Transición: Transición a los usuarios.

2.2 Bases de datos

Sobre las BD podemos hablar de las generaciones de BD.

La primera podríamos hablar de ellas por el año 1960 donde se implementar las primeros BD de redes. Surgen estructuras de datos como las listas y árboles. En 1976 se crea Data Base Task Group(DBTG) lo que dio origen a CODASYL teniendo como objetivo empezar con la regulación de un lenguaje de desarrollo, de donde nació COBOL.

La segunda generación 1970, traída por Edgar Frank Codd ingeniero de IBM, esto con su artículo, “Un modelo relacional de datos para grandes bancos de datos compartido”. Aquí fue donde definió el modelo de base de datos relacional y sus 12 reglas de Codd para sistemas de datos relacionales. Surgiendo los primeros Sistemas de Gestión de Bases de Datos Relacionales (DBMS relacionales) como: System R, de IBM y Oracle Corporation.

Tercera generación 1980 y 1990

En esta generación SEQUEL (Structure English Query Language) evolución a SQL (Structure Query Language) el cual fue usado por Oracle y Sybase volviéndose un estándar y en los 90 las BD evolucionaron al paradigma de lo Orientado a Objetos, la literatura sugiere como ejemplo de herramientas que están en este paradigma a Microsoft Excel y Microsoft Acces. Y posteriormente evolucionando hacia las BD objeto relacionales que tienen características de la programación orientada a objetos.

Un breve paréntesis y de actualidad al margen del contexto: Hoy día contamos con servicio de Cloud, donde almacenamos grandes cantidades de información, donde los servicios más populares tenemos AWS, Azure Cloud y Google Cloud. Además de que estamos ante la revolución de la WE 5.0 con el BIG Data, Business Intelligence, Minería de datos, Data Mart, etc.

Y la cereza del pastel, hoy día tenemos las IAs creciendo e implementándose apoyados de inmensos bancos de información, Blockchain con sus cadenas de bloques usando bases de datos descentralizadas como su principal fuerte y el metaverso que está a la vuelta de la esquina, que obviamente está usando algún nuevo paradigma en BD sin lugar a duda.

En este proyecto usaremos las BD relacionales como base para nuestro desarrollo, debido a que dentro de la industria es el modelo de BD más usado además de que tenemos amplia documentación de apoyo, así como experiencia para trabajos futuros.

Para la implementación de las BD relacionales usaremos SQL Structured Query Language lenguaje estandarizado para su uso en los principales sistemas de gestión de datos relacionales.

Cabe mencionar que el software (SGBD: Sistema de Gestión de Bases de Datos) que usaremos para esta gestión es MySQL.

Que por mencionar alguno otros como referencia tenemos: Oracle, Microsoft SQL Server, DB2 etc.

Tabla resumida ref [15]

Desventajas BD Relacional	Ventajas BD Relacional
No permiten una cómoda utilización de los bloques de datos como tipos de datos.	Favorecen la normalización.
Presentan deficiencias en el tratamiento de los datos gráficos. Datos gráficos, multi-media, CAD y sistemas de información geográfica.	Garantiza la integridad referencial. Exacta y completa.
	Evita la duplicidad de registros.

2.3 Herramientas de desarrollo web

El servidor web, el lenguaje de programación y el gestor de bases de datos

Las herramientas de para el desarrollo de esta aplicación son las siguientes:

- Es sistema estará bajo el modelo de MVC (Model View Controller).
- Visual Paradigm 16 para el modelado de Casos de uso, Análisis de Texto, modelado del diagrama ER y de clases.
- WAPM 3.2, sobre Windows 8 que fue donde se corre la instalación actual del sistema.
- La parte visual del sistema estará realizado con HTML, Css, PHP 7 y JS.
- Dreamweaver 8 para la maquetación de proyecto.2
- Para el BackEnd Usaremos PHP 7, JS y Java 1.6 y SQL para mySQL
- Estará montado sobre un servidor Apache usando como DBMS MySql.
- WorkBench de MySql como IDE para la creación y modelado de la BD.
- Eclipse como IDE para el desarrollo en Java 1.6 y PHP 7 y apoyo como editor de código Note++.

Capítulo 3. Análisis y diseño del sistema

Hasta este punto tenemos realizado el análisis de los casos de uso para así poder comenzar con la implementación, esto tras haber detectado y clasificado los requerimientos funcionales y no funcionales que mencionaremos a continuación.

Cabe mencionar que los casos de uso y diagramas ER han sufrido cambios, debido a que tras obtener el conocimiento del diplomado se ha realizado mejoras y ajustes a estos.

3.1. Requerimientos funcionales y no funcionales

Reglas de negocio:

- El sistema cuando nace tiene un administrador por default, el cual se debe de modificar con los datos del Administrador que se encargara del sistema.
- El sistema carga parámetros iniciales para cada tutor
- Cada tutor nace con posibilidad de dar de alta hasta 3 pacientes.
- Cada tutor nace con posibilidad de dar de alta hasta 3 cuestionarios por paciente.
- Cada tutor nace con posibilidad de dar de alta hasta 5 preguntas por cuestionario.
- El sistema cuando nace crea un administrador por default.

Actores

- Administrador.
- Tutor.
- Paciente.

El Administrador debe darse de alta por primera vez y actualizar el perfil de Administrador con sus datos generales: Nombre completo, Fecha Nacimiento, Teléfono, Correo, Usuario Alias, Password.

El Administrador accede al sistema escribiendo su Usuario Alias y su Password en la vista de login.

El Administrador tendrá las siguientes facultades en su tablero:

El Administrador, activa tutores del sistema.

El Administrador, bloquea tutores del sistema.

El Administrador, puede modificar los siguientes parámetros de los tutores:

- Número máximo de pacientes.
- Número máximo de cuestionarios.
- Número máximo de las preguntas por cuestionario.

El Administrador podrá visualizar en su tablero tutores del sistema, estatus, número máximo de pacientes que puede dar de alta, número máximo de cuestionarios que puede tener por paciente, número máximo de reactivos que puede tener por cuestionario, total de espacio utilizado.

El tutor debe darse de alta en el sistema llenando los siguientes datos generales: Nombre completo, Fecha Nacimiento, Teléfono, Correo, Usuario Alias, Password.

El tutor accede al sistema escribiendo su Usuario Alias y Password en la vista de login.

El tutor debe dar de alta al menos 1 paciente con los siguientes datos generales: Nombre completo, Fecha Nacimiento, Usuario Alias.

El tutor debe dar de alta al menos 1 cuestionario con los siguientes datos de cuestionario: Nombre del Cuestionario y descripción, Fecha creación.

El tutor debe agregar al menos 5 reactivos por cuestionario.

El tutor debe agregar a cada reactivo la siguiente información (datos reactivo): Pregunta a realizar al Paciente, Imagen asociada a la pregunta, Opcional: Un sonido asociado a la pregunta.

El tutor tendrá las siguientes facultades en su tablero:

- El tutor puede dar de alta pacientes
- El tutor puede modificar pacientes
- El tutor puede eliminar pacientes
- El tutor puede crear cuestionarios
- El tutor puede modificar cuestionarios
- El tutor puede eliminar cuestionarios
- El tutor puede crear reactivos
- El tutor puede modificar reactivos

- El tutor puede eliminar reactivos

El tutor visualizara en su tablero los datos de pacientes que tiene: Nombre paciente, Fecha Alta

El tutor visualizara los datos de cuestionarios que tiene cada paciente: Nombre del cuestionario, descripción del cuestionario, Fecha Alta

El tutor visualizara los reactivos de cada cuestionario

El tutor podrá generar un resumen de pacientes

El resumen de pacientes tendrá los siguientes datos id Paciente, Nombre Paciente, Alias Paciente, Fecha Alta Paciente

El tutor podrá generar un resumen de los resultados por paciente. *El resumen tendrá Nombre del Cuestionario, Fecha Aplicación, Calificación, Aciertos y Errores por pregunta.

El paciente resolverá los cuestionarios que le ponga el tutor.

Requerimientos funcionales:

- El sistema mostrara a los pacientes una serie de preguntas, donde cada pregunta mostrara 3 imágenes entre ellas la correcta. El paciente podrá elegir la respuesta correcta hasta en 3 intentos.
- El sistema permitirá generar cuestionarios con preguntas claras y cortas y se le permitirá cargar imágenes asociadas para resolverlo.
- El sistema permitirá generar reportes de los pacientes que tiene cada tutor, así como los resultados de cada cuestionario.
- El sistema permitirá al administrados bloquear o activar a los tutores, así como los parámetros de cada tutor para su operación con el sistema.

Requerimientos no funcionales:

- Se requiere de un S.O que soporte la instalación de WAPM 3.2, por ejemplo, Windows 8 que fue donde se corre la instalación actual del sistema.
- Se requiere que el equipo donde se instale debe tener el siguiente espacio
- Espacio por reactivo: 10 MB MAX por reactivo.
- Espacio por cuestionario, Max 5 Reactivos: 50MB
- Espacio por 3 cuestionarios: 150MB
- Espacio por 3 pacientes :450MB
- Espacio por 1 tutor:450MB
- El espacio que se requiere reservar para el sistema se catalogó en 2:

Espacio de instalación:

WAMP: ~350MB (x86) / ~500MB (x64)

Sistema: 500MB

- Espacio de almacenamiento de recursos (Imágenes, Sonidos, Crecimiento de la BD):

Imágenes y sonidos (ANEXO12), dependerán del número de tutores que se permitan en el sistema, se recomienda 5 tutores.

Lo cual nos da un espacio requerido para multimedia de 2,250MB

En caso de instalación local, se requiere que el equipo cuente con al menos 4GB de RAM un procesador de al menos 2.2 Ghz.

3.2 Casos de uso

Los artefactos UML que se presentan corresponderá a los casos de uso y el diagrama de clases del sistema Web de la primer versión de análisis, posteriormente se mostrara la versión actualizada.

En este caso nuestros casos de uso se describen en (Img 4)

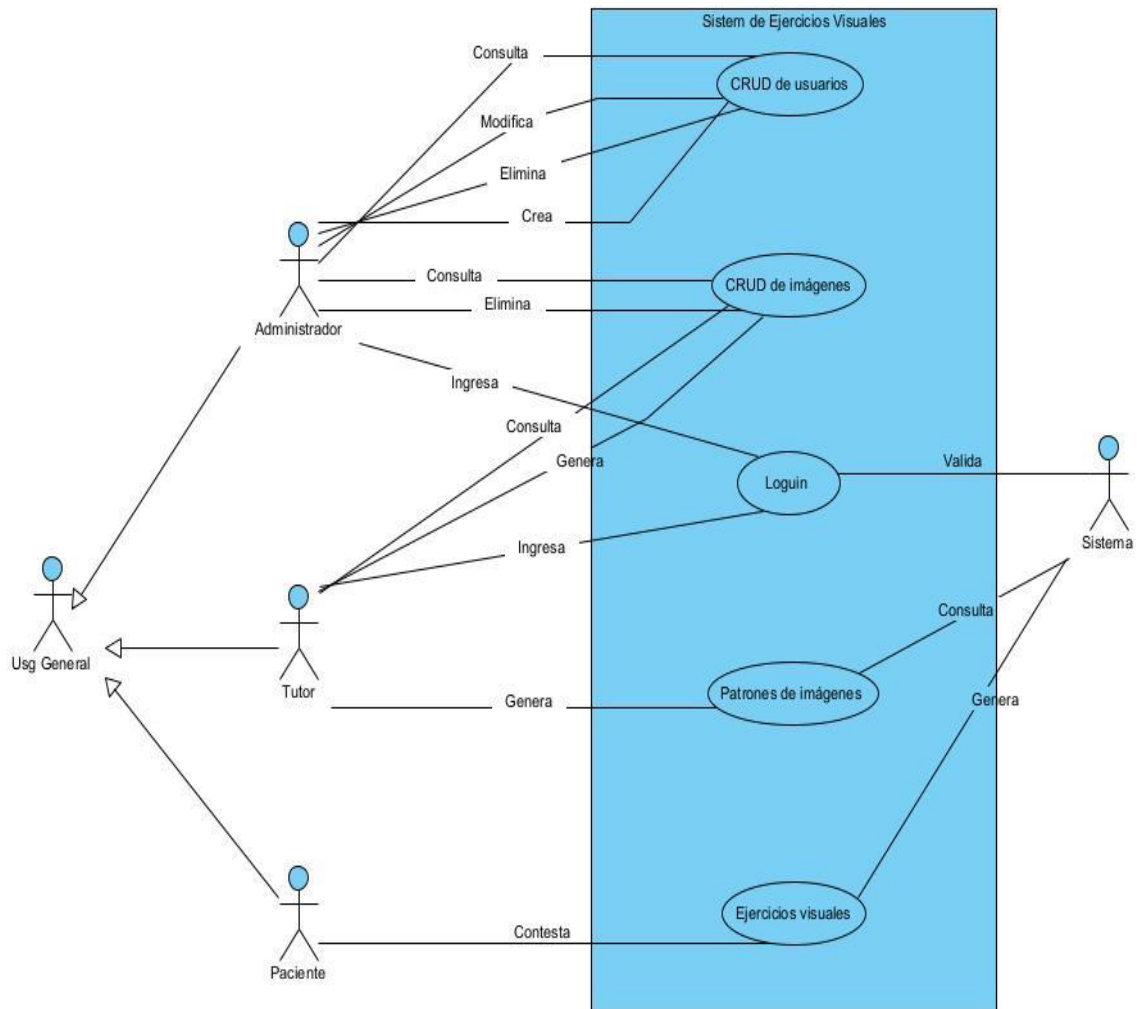
El sistema cuenta con 3 usuarios generales y 4 casos de uso generales.

Usuarios: Paciente, Tutor, Administrador

Casos de uso: Ejercicios visuales, Patrones de imágenes, CRUD (Create, Read, Update, Delete) de imágenes, CRUD de usuarios.

Usuarios:

- Paciente: Aquel usuario que realizará los ejercicios visuales en la plataforma web.
- Tutor: Aquel usuario que genera el patrón de imágenes a mostrar a su paciente, alta, baja o modificación de imágenes que se mostrarán en los ejercicios visuales y consultará el resultado de los ejercicios visuales.
- Administrador: Aquel usuario que ingresa, modifica o elimina usuarios del sistema web, así también como la asignación de roles al usuario registrado. CRUD de imágenes.



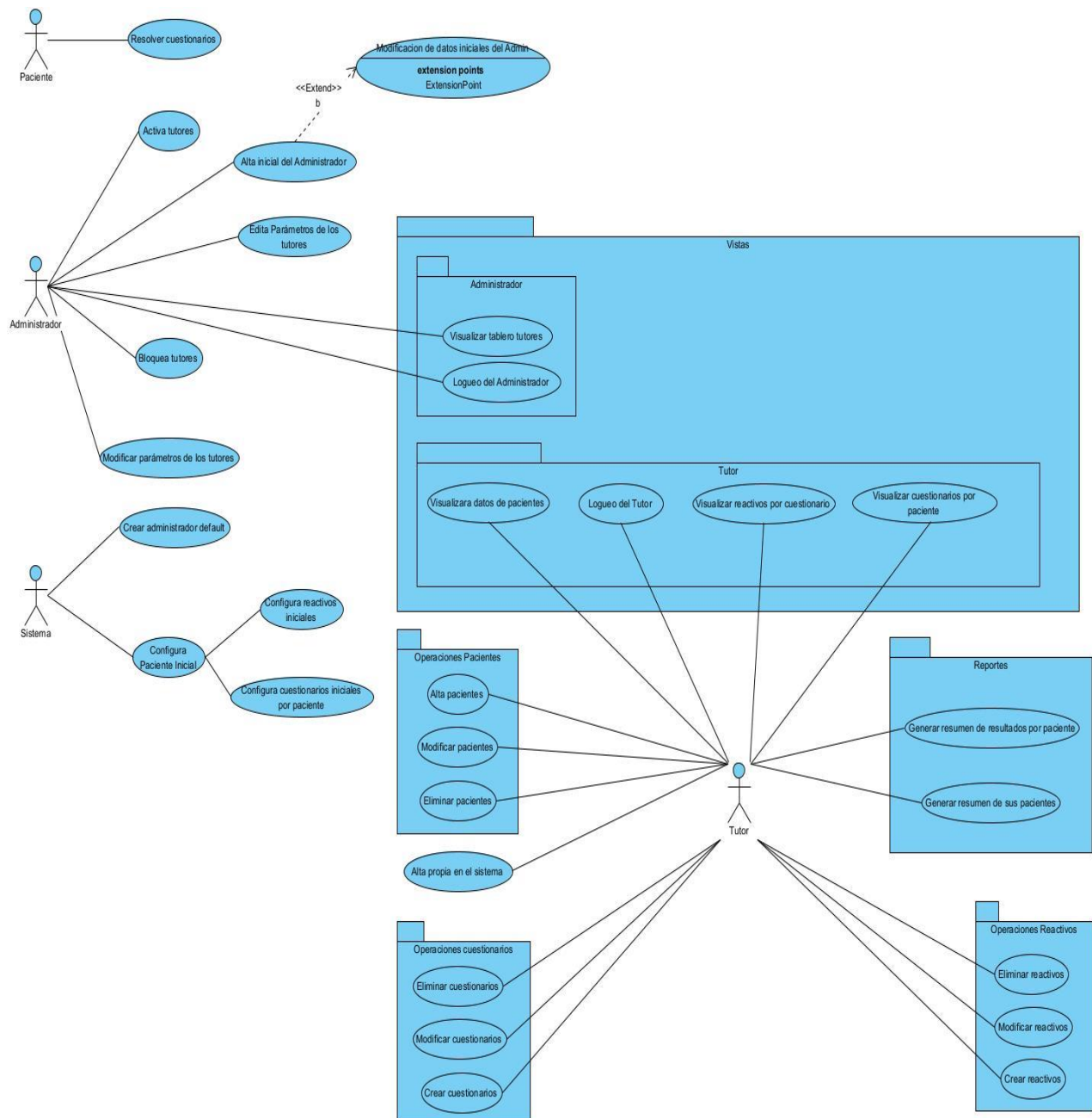
Img 4 Casos de uso del sistema v1 Propio

Casos de uso:

- Ejercicios visuales, el sistema mostrará una serie de imágenes al paciente para que el paciente pueda responder las preguntas asociadas a la imagen, por ejemplo, ¿Esta imagen corresponde a tu hijo?" y guardará su respuesta seleccionada. El sistema mostrará al tutor las evaluaciones de los ejercicios realizados por su paciente.
- Patrones de imágenes, El sistema permitirá crear al tutor la relación de la imagen almacenada contra la pregunta que se le mostrará al paciente, por ejemplo: Para la imagen X la pregunta será: ¿Esta persona es tu hijo?
- CRUD de imágenes, el sistema permitirá subir, visualizar, modificar y eliminar imágenes al repositorio del sistema.

- CRUD de usuarios, el sistema permitirá al administrador la Creación, Consulta, Modificación y Eliminación de usuario al Sistema, así como la asignación de un único rol: Paciente, Tutor o Administrador. 1

A continuación, mostraremos la versión actualizada (*Img5*) y sus especificaciones de CU:



Img 5 Diagrama general de casos de uso

Fichas V3 de casos de Uso

CU1 RESOLVER CUESTIONARIOS: El paciente resuelve los cuestionarios que se le presentan.

Caso de Uso	Resolver Cuestionarios CU-01
Actor	Paciente
Requerimientos relacionados	-
Precondiciones	El tutor del paciente debe estar logueado en el sistema.
Objetivo en contexto-Descripción	Resolver el cuestionario mostrado por el sistema.
Flujo básico	El paciente ingresa en la sección de cuestionarios. El sistema le muestra una vista con imágenes y preguntas. El paciente selecciona la imagen que considere sea correcta a la pregunta mostrada. Lo repite N veces hasta finalizar el cuestionario.
Flujos Alternativos	NA

CU2 Alta inicial del administrador: Cuando nace el sistema tiene un usuario administrador por default, se debe de actualizar los datos del admin default con los datos del administrador en turno.

Caso de Uso	Alta inicial del administrador CU-02
Actor	Administrador
Requerimientos relacionados	-
Precondiciones	El administrador debe estar logueado en el sistema.
Objetivo en contexto-Descripción	Actualizar en el sistema, los datos definitivos del usuario que será el administrador del sistema.

Flujo básico	El nuevo usuario administrador entrara a la vista de administración del sistema y actualizara sus datos generales: Nombre completo, Fecha Nacimiento, Teléfono, Correo, Usuario Alias, Password y guardara sus cambios.
Flujos Alternativos	El caso de no actualizarse estos datos, se podrá acceder al sistema con el rol de Admin por medio de las credenciales USR:Admin123@ Password:Password123@

CU3 Activa/des Activa tutores: El tutor activa/des activa tutores para el acceso al sistema.

Caso de Uso	Activa/des Activa tutores CU-03
Actor	Administrador
Requerimientos relacionados	-
Precondiciones	El administrador debe estar logueado en el sistema.
Objetivo en contexto-Descripción	Activación/Des activación de tutores.
Flujo básico	El administrador entra al tablero de administración tutores. El administrador selecciona al tutor a ACTIVAR/DESACTIVAR, aplica el cambio dando clic al botón aplicar cambio.
Flujos Alternativos	El flujo aplica para activar o desactivar un usuario.

CU4 Edición de parámetros de los tutores: El administrador puede editar los parámetros de los tutores.

Caso de Uso	Edición de parámetros de los tutores CU-04
-------------	--

Actor	Administrador
Requerimientos relacionados	-
Precondiciones	El administrador debe estar logueado en el sistema.
Objetivo en contexto-Descripción	Editar los parámetros de los tutores de acuerdo con las necesidades del tutor a editar. El administrador ingresa a la vista de edición de parámetros de tutores. Selecciona el tutor a editar. El administrador puede editar: Número máximo de pacientes. Número máximo de cuestionarios. Número máximo de las preguntas por cuestionario. Guarda cambios
Flujo básico	
Flujos Alternativos	--

CU5 Logueo del Administrador: El administrador accede al sistema.

Caso de Uso	Edición de parámetros de los tutores CU-04
Actor	Administrador
Requerimientos relacionados	
Precondiciones	CU-02, El administrador debe haber actualizado sus datos
Objetivo en contexto-Descripción	Acceder al sistema
	El administrador ingresa a la vista de Loguin, ingresa su usuario y password, envía la información, el administrador accede al sistema. El sistema muestra el tablero de administración.
Flujo básico	
Flujos Alternativos	--

CU6 Crear administrador del sistema: El sistema al nacer crea un usuario admin por default.

Caso de Uso	Crear administrador del sistema CU-06
Actor	Sistema
Requerimientos relacionados	
Precondiciones	
Objetivo en contexto-Descripción	Crear usuario default del sistema Tras la instalación del sistema, se corre un script que genera a nivel de BD el usuario default. Se ingresa al sistema con las credenciales default para validar el usuario creado.
Flujo básico	
Flujos Alternativos	--

CU7 Config. de paciente inicial: El sistema configura con sus valores iniciales un paciente.

Caso de Uso	Config. de paciente inicial CU-07
Actor	Sistema
Requerimientos relacionados	Un tutor debe estar dado de alta.
Precondiciones	
Objetivo en contexto-Descripción	Configurar los parámetros iniciales de un paciente. Configuración de numero de reactivos iniciales. Configuración de número de cuestionarios iniciales.
Flujo básico	Tras la instalación del sistema, se corre un script que genera a nivel de BD los parámetros iniciales por la configuración de numero de reactivos y cuestionarios iniciales por paciente.
Flujos Alternativos	--

CU8 Alta de tutor: El tutor genera su registro en el sistema

Caso de Uso	Alta tutor CU-08
Actor	Tutor
Requerimientos relacionados	--
Precondiciones	
Objetivo en contexto-Descripción	Creación del registro de un tutor en el sistema. Un usuario ingresa a la sección de Alta del sistema. Captura su Nombre completo, Fecha Nacimiento, Teléfono, Correo, Usuario Alias, Password. Guarda los cambios e ingresa al sistema con la vista de tutor.
Flujo básico	
Flujos Alternativos	--

CU09 Creación de paciente: El tutor crea un paciente a su cargo

Caso de Uso	Creación de paciente CU-09
Actor	Tutor
Requerimientos relacionados	--
Precondiciones	El tutor debe estar logueado y activo en el sistema.
Objetivo en contexto-Descripción	Crear un paciente al cual dará seguimiento El tutor ingresa a su tablero en la sección de pacientes. Ingresa a la sección alta de paciente. Se valida número máximo de pacientes que puede tener. Ingresa los siguientes datos del paciente Nombre completo, Fecha Nacimiento, Usuario Alias. Guarda los cambios.
Flujo básico	
Flujos Alternativos	En caso de rebasar el número máximo de pacientes el sistema no le permitirá dar de alta un nuevo paciente.

CU10 Modificación de paciente: El tutor modifica un paciente a su cargo

Caso de Uso	Modificación de paciente CU-10
Actor	Tutor
Requerimientos relacionados	--
Precondiciones	El tutor debe estar logueado y activo en el sistema. Debe tener un paciente dado de alta.
Objetivo en contexto-Descripción	Modificar un paciente dado de alta.
Flujo básico	El tutor ingresa a su tablero en la sección de pacientes. Ingresa a la sección modificación de paciente. Puede modificar los siguientes datos del paciente Fecha Nacimiento y/o Usuario Alias. Guarda los cambios.
Flujos Alternativos	--

CU11 Eliminación de paciente: El tutor elimina un paciente a su cargo

Caso de Uso	Modificación de paciente CU-10
Actor	Tutor
Requerimientos relacionados	--
Precondiciones	El tutor debe estar logueado y activo en el sistema. Debe tener un paciente dado de alta.
Objetivo en contexto-Descripción	Eliminar un paciente dado de alta.
Flujo básico	El tutor ingresa a su tablero en la sección de pacientes. Selecciona un paciente y da clic en eliminar paciente.
Flujos Alternativos	--

CU12 Creación de cuestionarios: El tutor crea un cuestionario al paciente

Caso de Uso	Creación de cuestionarios CU-12
Actor	Tutor
Requerimientos relacionados	--
Precondiciones	Un tutor debe estar activo, de estar logueado en el sistema. El tutor debe tener al menos un paciente dado de alta.
Objetivo en contexto-Descripción	Crear un cuestionario
Flujo básico	El tutor ingresa a la vista de pacientes. Selecciona un paciente. El sistema valida el número máximo de cuestionarios Selecciona Crear cuestionario Ingresa Nombre del cuestionario y descripción y lo guarda. El sistema crea un nuevo cuestionario con la fecha del momento de la creación.
Flujos Alternativos	En caso de rebasar el número máximo de cuestionarios por paciente, el sistema no le permitirá agregar otro.

CU13 Modificación de cuestionarios: El tutor modifica un cuestionario al paciente

Caso de Uso	Modificación de cuestionarios CU-13
Actor	Tutor
Requerimientos relacionados	--
Precondiciones	Un tutor debe estar activo, de estar logueado en el sistema. El tutor debe tener al menos un paciente dado de alta. Debe tener un cuestionario en su paciente
Objetivo en contexto-Descripción	Modificar un cuestionario
Flujo básico	El tutor ingresa a la vista de pacientes. Selecciona un paciente. Selecciona Modificar cuestionario Modifica descripción y lo guarda.
Flujos Alternativos	--

CU14 Eliminación de cuestionarios: El tutor elimina un cuestionario al paciente

Caso de Uso	Eliminación de cuestionarios CU-14
Actor	Tutor
Requerimientos relacionados	--
Precondiciones	Un tutor debe estar activo, de estar logueado en el sistema. El tutor debe tener al menos un paciente dado de alta. Debe tener un cuestionario en su paciente
Objetivo en contexto-Descripción	Eliminación de un cuestionario
Flujo básico	El tutor ingresa a la vista de pacientes. Selecciona un paciente. Selecciona eliminar cuestionario
Flujos Alternativos	--

CU15 Creación de reactivos: El tutor crea un reactivo al cuestionario

Caso de Uso	Creación de reactivos CU-15
Actor	Tutor
Requerimientos relacionados	--
Precondiciones	Un tutor debe estar activo, de estar logueado en el sistema. El tutor debe tener al menos un paciente dado de alta. Debe tener un cuestionario en su paciente
Objetivo en contexto-Descripción	Creación de reactivos.
Flujo básico	El tutor ingresa a la vista de pacientes. Selecciona un paciente. Selecciona un cuestionario. El sistema valida número máximo de reactivos El tutor debe agregar a cada reactivo la siguiente información (datos reactivo): Pregunta a realizar al Paciente, Imagen asociada a la pregunta, Opcional: Un sonido asociado a la pregunta.
Flujos Alternativos	En caso de rebasar el máximo de reactivos por cuestionario el sistema no le permitirá crearlo.

CU15 Modificación de reactivos: El tutor modifica un reactivo al cuestionario

Caso de Uso	Modificación de reactivos CU-15
Actor	Tutor
Requerimientos relacionados	--
Precondiciones	Un tutor debe estar activo, de estar logueado en el sistema. El tutor debe tener al menos un paciente dado de alta. Debe tener un cuestionario en su paciente, debe tener un reactivo a editar.
Objetivo en contexto-Descripción	Modificación de reactivos.
Flujo básico	El tutor ingresa a la vista de pacientes. Selecciona un paciente. Selecciona un cuestionario. Selección un reactivo Modifica la pregunta asociada y/o la imagen asociada.
Flujos Alternativos	Guarda los cambios.
	--

CU16 Eliminación de reactivos: El tutor elimina un reactivo al cuestionario

Caso de Uso	Eliminación de reactivos CU-16
Actor	Tutor
Requerimientos relacionados	--
Precondiciones	Un tutor debe estar activo, de estar logueado en el sistema. El tutor debe tener al menos un paciente dado de alta. Debe tener un cuestionario en su paciente, debe tener un reactivo a editar.
Objetivo en contexto-Descripción	Modificación de reactivos.
Flujo básico	El tutor ingresa a la vista de pacientes. Selecciona un paciente. Selecciona un cuestionario. Selección un reactivo Y da clic en eliminar el reactivo.
Flujos Alternativos	--

CU17 Reporte resultados por paciente: Generar un resumen de resultados por paciente

Caso de Uso	Reporte resultados por paciente CU-17
Actor	Tutor
Requerimientos relacionados	--
Precondiciones	Un tutor debe estar activo, de estar logueado en el sistema. El tutor debe tener al menos un paciente dado de alta. El paciente debe tener contestado al menos un cuestionario.
Objetivo en contexto-Descripción	Generar reportes de resultados por paciente
Flujo básico	El tutor ingresa a la vista de pacientes. Selecciona un paciente. Selecciona un cuestionario. Selecciona resultados. El resumen tendrá Nombre del Cuestionario, Fecha Aplicación, Calificación, Aciertos y Errores por pregunta
Flujos Alternativos	--

CU18 Reporte resumen de pacientes: Generar un resumen de pacientes

Caso de Uso	Reporte resumen de pacientes CU-18
Actor	Tutor
Requerimientos relacionados	--
Precondiciones	Un tutor debe estar activo, de estar logueado en el sistema. El tutor debe tener al menos un paciente dado de alta.
Objetivo en contexto-Descripción	Generar reportes de pacientes asignados
Flujo básico	El tutor ingresa a la vista de pacientes. Selección reporte de pacientes El resumen de pacientes tendrá los siguientes datos id Paciente, Nombre Paciente, Alias Paciente, Fecha Alta Paciente
Flujos Alternativos	--

CU19 Logueo de tutor: El tutor accede al sistema

Caso de Uso	Logueo de tutor CU-19
Actor	Tutor
Requerimientos relacionados	--
Precondiciones	Un tutor debe estar activo en el sistema.
Objetivo en contexto-Descripción	Acceder al sistema. El tutor ingresa a la vista de Loguin del sistema. Ingresa su usuario y password Se valida que el tutor este activo Se valida sus claves y accede al sistema con su tablero de tutor.
Flujo básico	
Flujos Alternativos	--

CU20 Tablero de tutor: Tablero con información de pacientes cuestionarios y pacientes.

Caso de Uso	Tablero de tutor CU-19
Actor	Tutor
Requerimientos relacionados	--
Precondiciones	Un tutor debe estar activo en el sistema y logueado.
Objetivo en contexto-Descripción	Mostrar tablero del tutor. El tutor se loguea en el sistema. El sistema muestra en el tablero del tutor la siguiente información: El tutor visualizara en su tablero los datos de pacientes que tiene: Nombre paciente, Fecha Alta El tutor visualizara los datos de cuestionarios que tiene cada paciente: Nombre del cuestionario, descripción del cuestionario, Fecha Alta El tutor visualizara los reactivos de cada cuestionario
Flujo básico	
Flujos Alternativos	--

3.3 Diagrama de clases

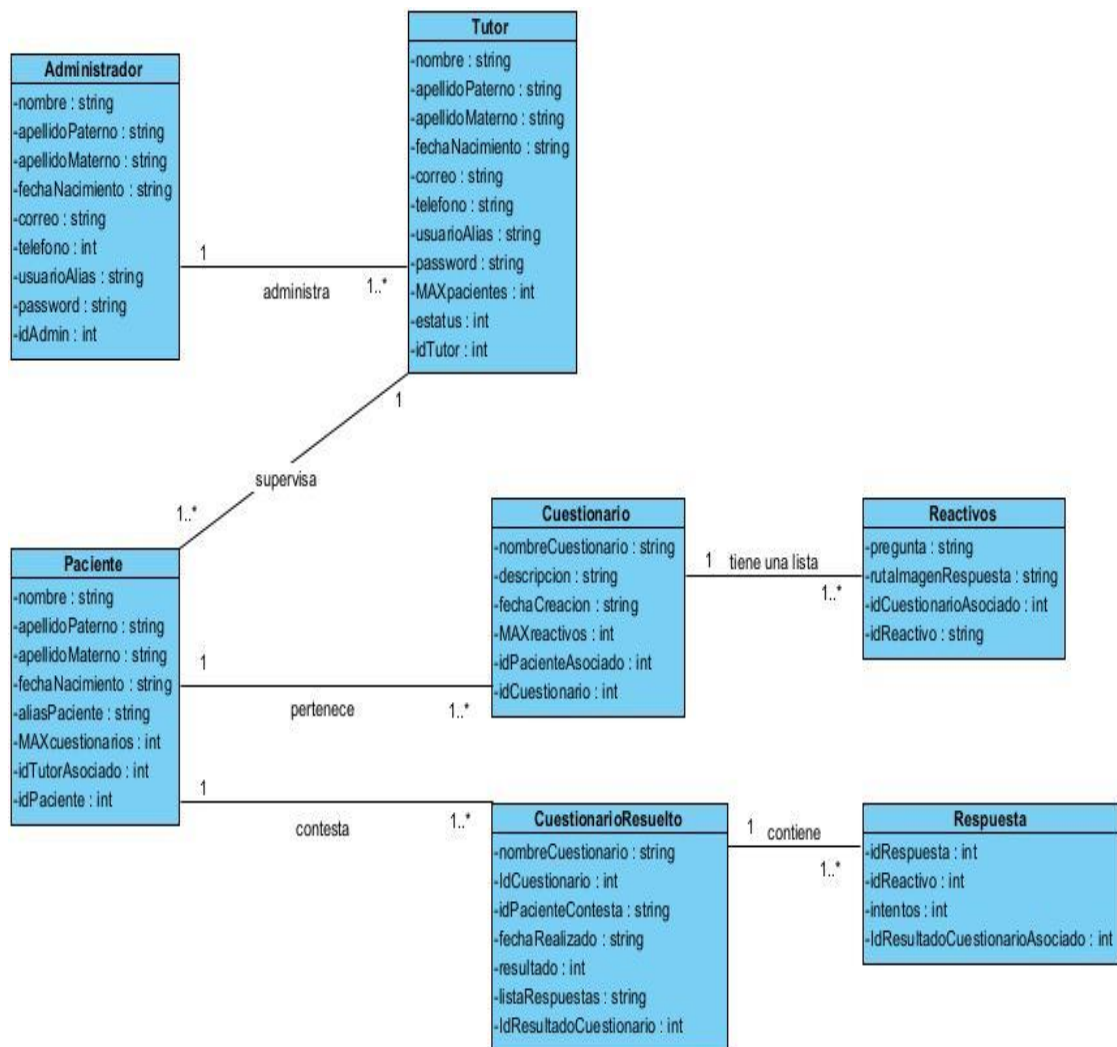
En el diagrama de clases modelaremos las estructuras que se usaran en nuestra aplicación donde se plasma la clase, sus relaciones y atributos a modelar.

En principio tenemos nuestra tabla de clases:

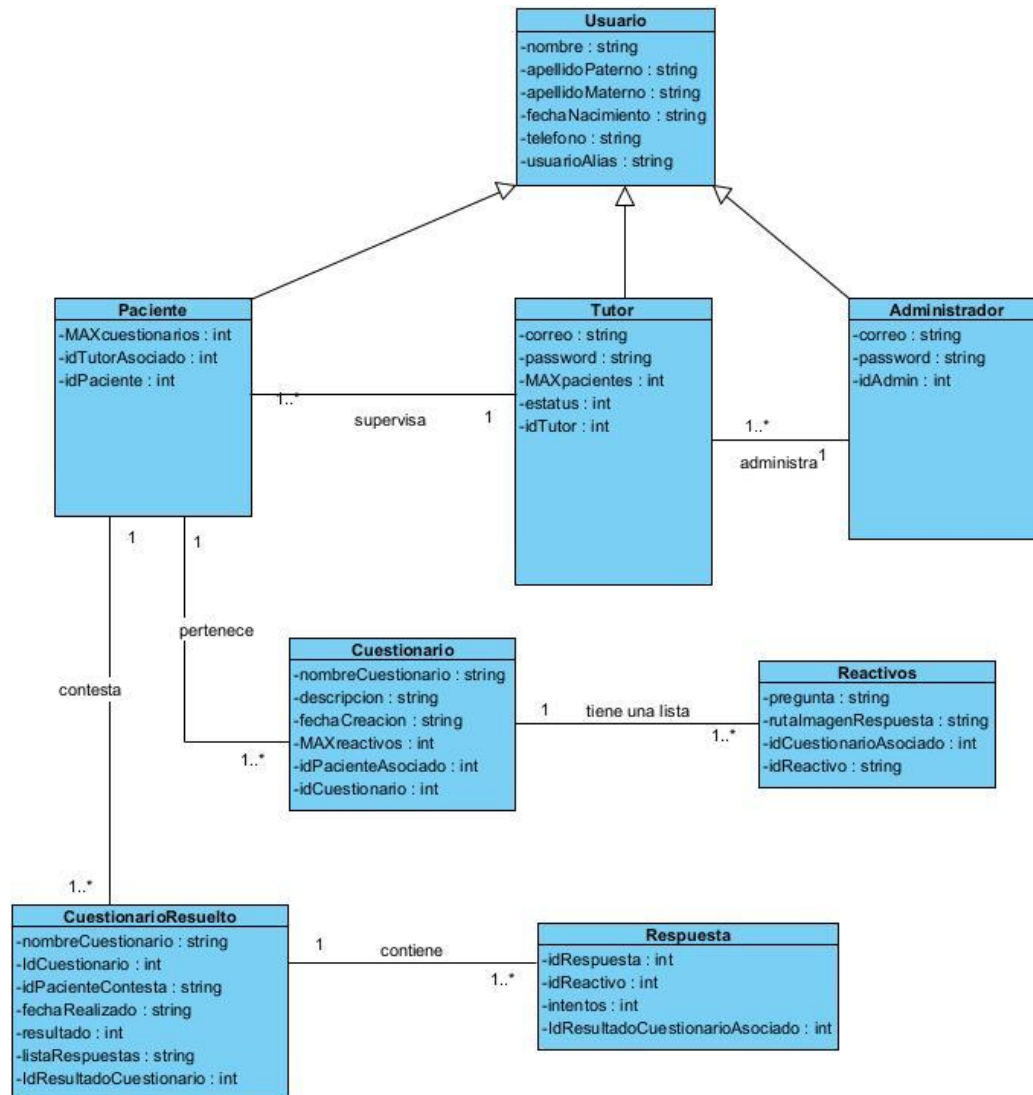
Clases	Atributos	Tipo de dato	Método
Administrador	Nombre Completo* Fecha Nacimiento* Alias admin* Password Admin* Correo Telefono IdAdmin	String Date String String String Numérico Numerico	DAO Admin
Tutor	Nombre Completo* Fecha Nacimiento* Alias admin* Password Admin* Correo Teléfono Máximo de Pacientes IdTutor estatus	String Date String String String Numérico Numérico Numerico Numetico	DAO Tutor
Paciente	Nombre Completo* Fecha Nacimiento* Alias paciente* Máximo de cuestionarios IdPaciente IdTutorAsociado	String Date String Numérico Numérico Numerico	DAO Pacien te
Cuestionario	Nombre cuestionario Descripción Fecha Creación Máximo de reactivos IdCuestionario IdPAcienteAsociado	String String Date Numérico Numérico Numerico	DAO Cuesti onario
Reactivo	Pregunta Ruta Imagen Respuesta idReactivo idCuestionarioAsociado	String String Numerico Numerico	DAO Reacti vo
ResultadoCuestionario	Nombre de Cuestionario IdCuestionario IdPaciente Contesta	String Numérico Numérico	Repor tes Cuesti

	Fecha Realizado Resultado (se puede calcular) Lista Respuestas IdResultadoCuestionario	Date Numérico Respuestas Numérico	onarios
Respuesta	idRespuesta* idReactivo* intentos IdResultadoCuestionarioAsociado	Numérico Numérico Numérico Numérico	

A continuación, en la (Img 5), se muestra nuestro primer acercamiento de Diagrama de clases.



Img 5 Diagrama de clases v1 Propio



Img 6 Diagrama de Clases v2 Propio

De donde podemos ver un cambio entre (Img 5) y (Img 6) debido a una generalización.

En nuestro diagrama tenemos las siguientes clases:

- **Usuario** de la cual se le generalizan Administrador, Tutor y Paciente.

Usuario cuenta con datos generales: nombre completo, fecha de nacimiento, teléfono y alias.

- En la clase **Paciente**, se le agregan los atributos de tutor asociado que corresponde al ID de su tutor y el número máximo de cuestionarios que se le pueden configurar.
- La clase **Tutor** tiene atributos como numero Max de pacientes indicando cuantos pacientes puede administrar, su propio estatus en el sistema, correo y password. Y la clase Administrador que solo se le especializa correo y contraseña.
- La clase **Reactivo**, contiene la imagen a asociada a la pregunta y la pregunta a realizar.
- La clase **Cuestionario**, contiene descripción, fecha, máximo de reactivos que puede tener y el id del tutor que usara ese cuestionario.
- La clase **Respuesta** contiene el id correspondiente al reactivo que se está respondiendo y el número de intento en el cual fue respondido correctamente.
- La clase **CuestionarioResuelto** contiene la lista de respuestas, fecha de ejecución, id del paciente que contesta y su resultado.

3.4. Diagrama entidad relación del sistema

Para el diagramado de nuestro sistema en lo que concierne a las Bases de Datos, ocuparemos el modelo entidad relación ya que nos permite tener una abstracción del mundo real hacia lo que plasmaremos en objeto y relaciones entre estos. En la (Img 7) se muestra el diagrama entidad relación.

Universo del discurso de una Empresa:

Administrador (con identificador No_Admin, Nombre completo, Fecha de Nacimiento, Teléfono, Correo, UsuarioAlias, password);

Tutor(con identificador No_Tutor, Nombre completo, Fecha de Nacimiento, Teléfono, Correo, UsuarioAlias, password, estatus, MAXPacientes);

Paciente(con identificador No_Paciente, Nombre completo, Fecha de Nacimiento, Telefono, UsuarioAlias, MAXCuestionarios);

Cuestionario(con identificador No_Cuestionario, Nombre, Descripción, FechaCreacion, MAXReactivos);

Reactivo(con identificador No_Reactivo, pregunta, rutalmgRespuesta);

CuestionarioRespuesta(con identificador No_CuestionarioRespuesta, fechaRealizado, resultado);

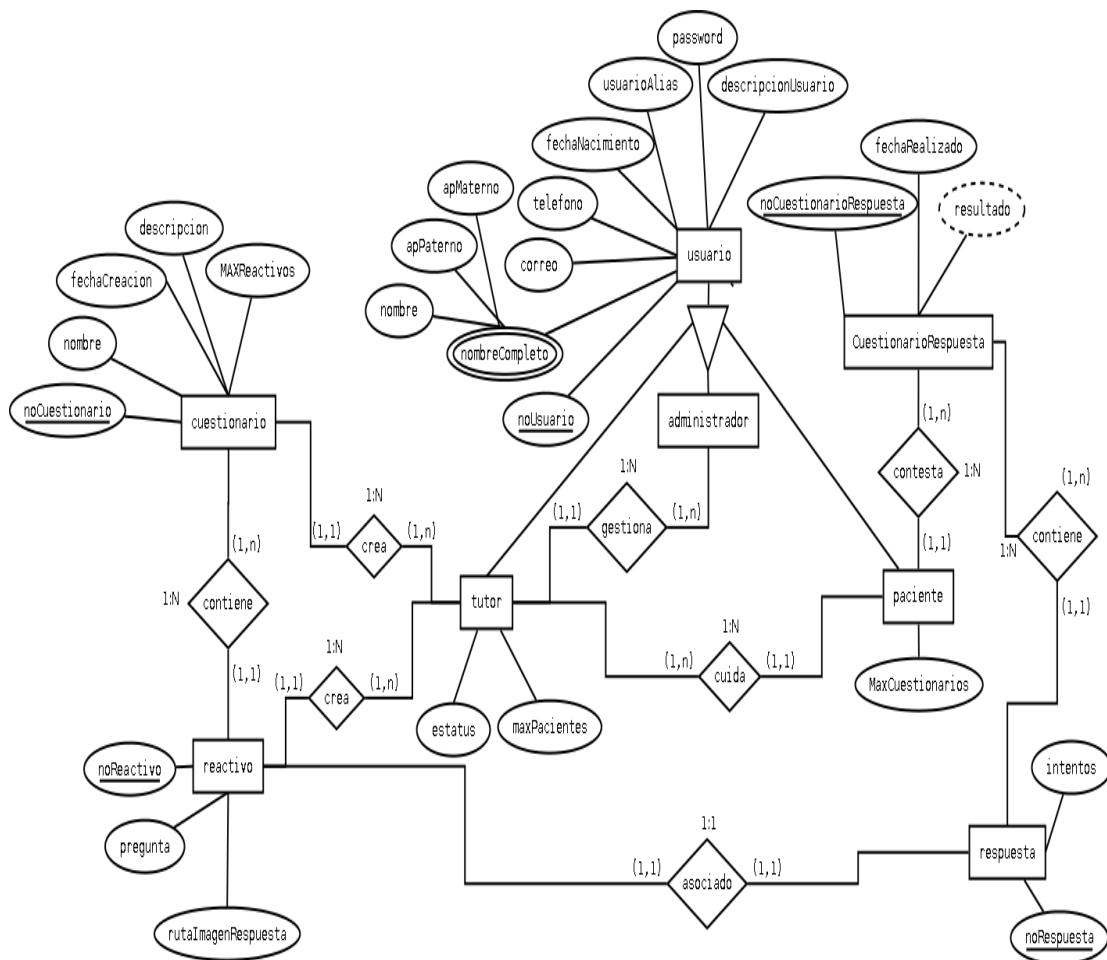
Respuesta(con identificador No_Respuesta, intentos);

Se identifica que se puede realizar una generalización con Administrador, Tutor y Paciente, donde el padre es usuario, se procede a redistribuir los atributos generales:

Usuario(con identificador No_Usuario, Nombre completo, Fecha de Nacimiento, Telefono, UsuarioAlias,descripcionUsuario) y especializamos

a Tutor (estatus, MAXPacientes) y paciente(MAXCuestionarios)

Obtenido el siguiente diagrama (Img 7):



Img 7 Diagrama Entidad Relación Propio

De las relaciones encontradas:

- Un Admin, gestiona uno o varios Tutores y un Tutor es gestionado por un solo Admin
- Un Tutor cuida a uno o varios Pacientes y un Paciente es cuidado por un solo Tutor.
- Un Tutor crea uno o varios Cuestionario y un Cuestionario es creado por un Tutor
- Un Tutor crea uno o varios reactivos de un cuestionario y un reactivo esta creado por un Tutor
- Un Cuestionario contiene varios reactivos y un reactivo está solo en un Cuestionario
- Un Paciente contesta uno o varios CuestionariosRespuesta y un CuestionariosRespuesta es contestado solo por un Paciente.
- Un CuestionariosRespuesta contiene varias Respuestas, y una Respuesta está solo en un CuestionariosRespuesta.
- Una Respuesta está asociada a un Reactivo y un Reactivo está asociada a una Respuesta.

3.5. Diagrama Relacional del Sistema

Al diagrama relacional llegamos tras aplicar el proceso de normalización de Edgar F. Codd, se hace la mención de que el proceso lo acotamos hasta la 3er Forma normal.

Trabajaremos con

- -Primera Forma Normal (1NF),
- -Segunda Forma Normal (2NF)
- -Tercera Forma Normal (3NF).

Esto con el fin de no tener problemas de redundancia de los datos., evitar anomalías de inserción, modificación y eliminación de datos y garantizar la integridad de los datos.

1. Primera forma normal: Una tabla está en Primera Forma Normal (1FN), si cada uno de sus campos tiene valores atómicos con respecto a su sistema de gestión de base de datos (DBMS), es decir que ese campo no se pueda descomponer en más “sub-campos”

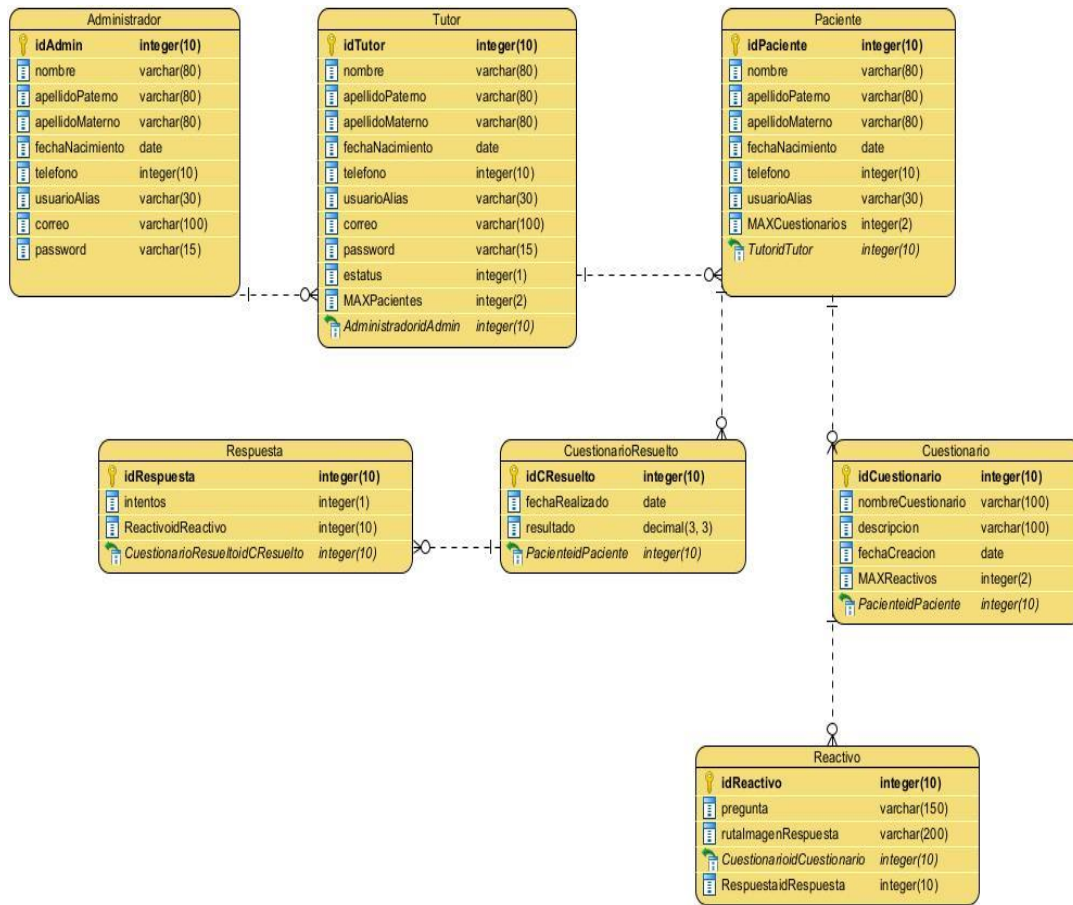
Por ejemplo, al detectar los atributos multivaluados de una entidad del Modelo E-R.

Tampoco debe haber registros repetidos.

2. Segunda forma Normal: Primero debe estar en 1NF, Una tabla está en Segunda Forma Normal (2FN), que no existen dependencias parciales. Todos los atributos que no son clave principal deben depender únicamente de la clave principal. En otras palabras, que todas sus columnas deben estar ligadas a una clave primaria compuesta.
3. Tercera Forma Normal: Una tabla está en Tercer Forma Normal (3FN), si no existe ninguna dependencia entre algunos de sus campos con otros campos que no son ni hacen parte de su llave primaria. Esta forma normal se garantiza al seleccionar solamente los atributos que están directamente relacionados con la entidad que describen y con el IP que la identifica.

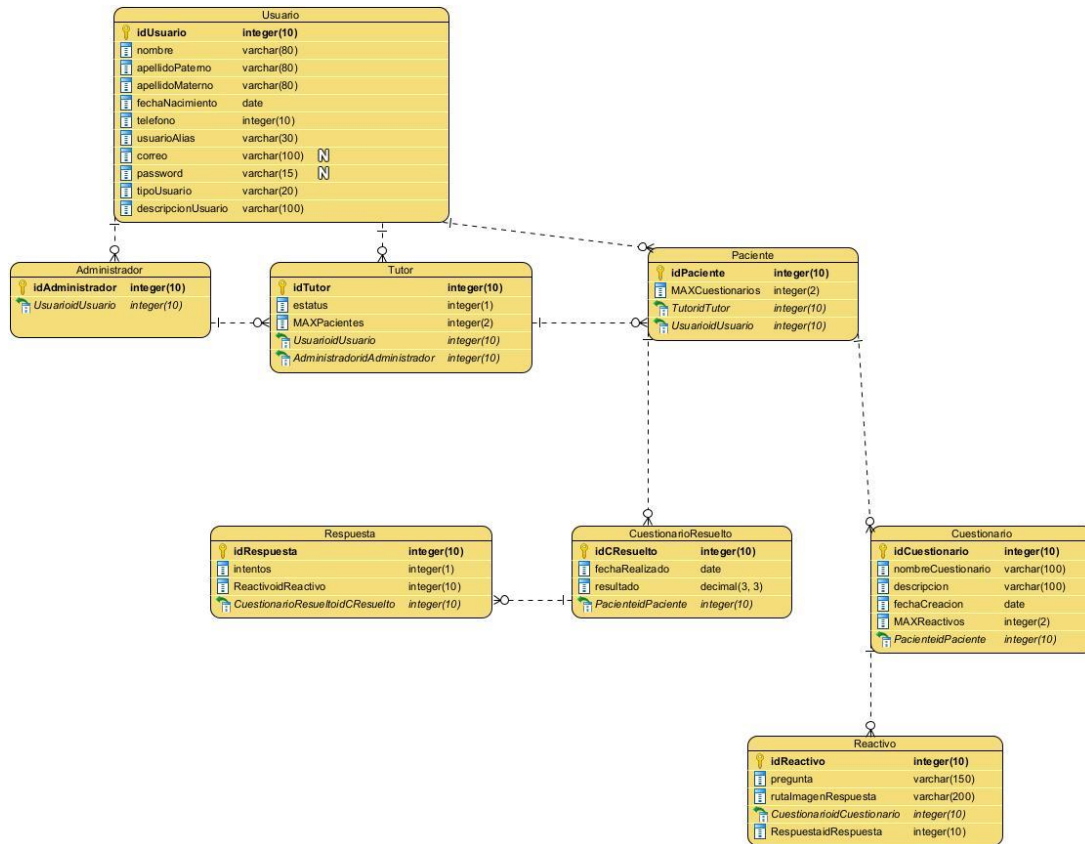
Para que una tabla esté en 3FN, debe haber cumplido la Regla de la 2FN.

Para el modelo relacional utilizamos la herramienta VisualParadigm y obtuvimos el primer acercamiento (Img 8)



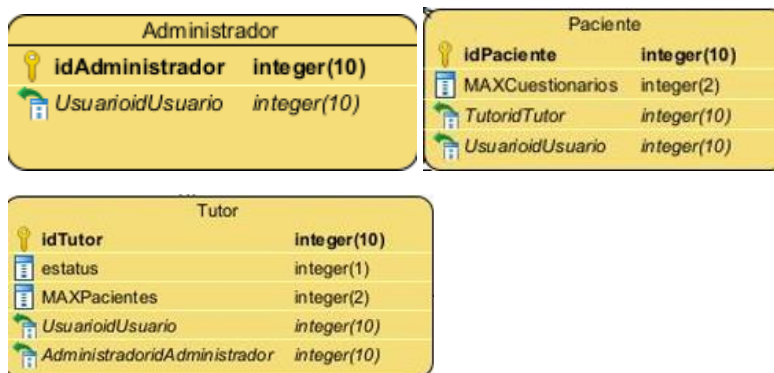
Img 8 Diagrama Relacional v1 Propio

Hacemos notar que con respecto al D-ER (Img 7) se debe pasar (Img 8) al (Img 9), debido la generalización expresada.



Img 9 Diagrama Relacional v2 Propio

Sobre las Tablas Administrador, Paciente y Tutor:



Se encuentra en 1NF, 2NF y 3NF.

Cuenta con PK no hay nullos, los campos son atómicos,

Los campos dependen de su PK y no hay transitividad.

Para la tabla Usuario (Img 10) mostrada en (Img 9)

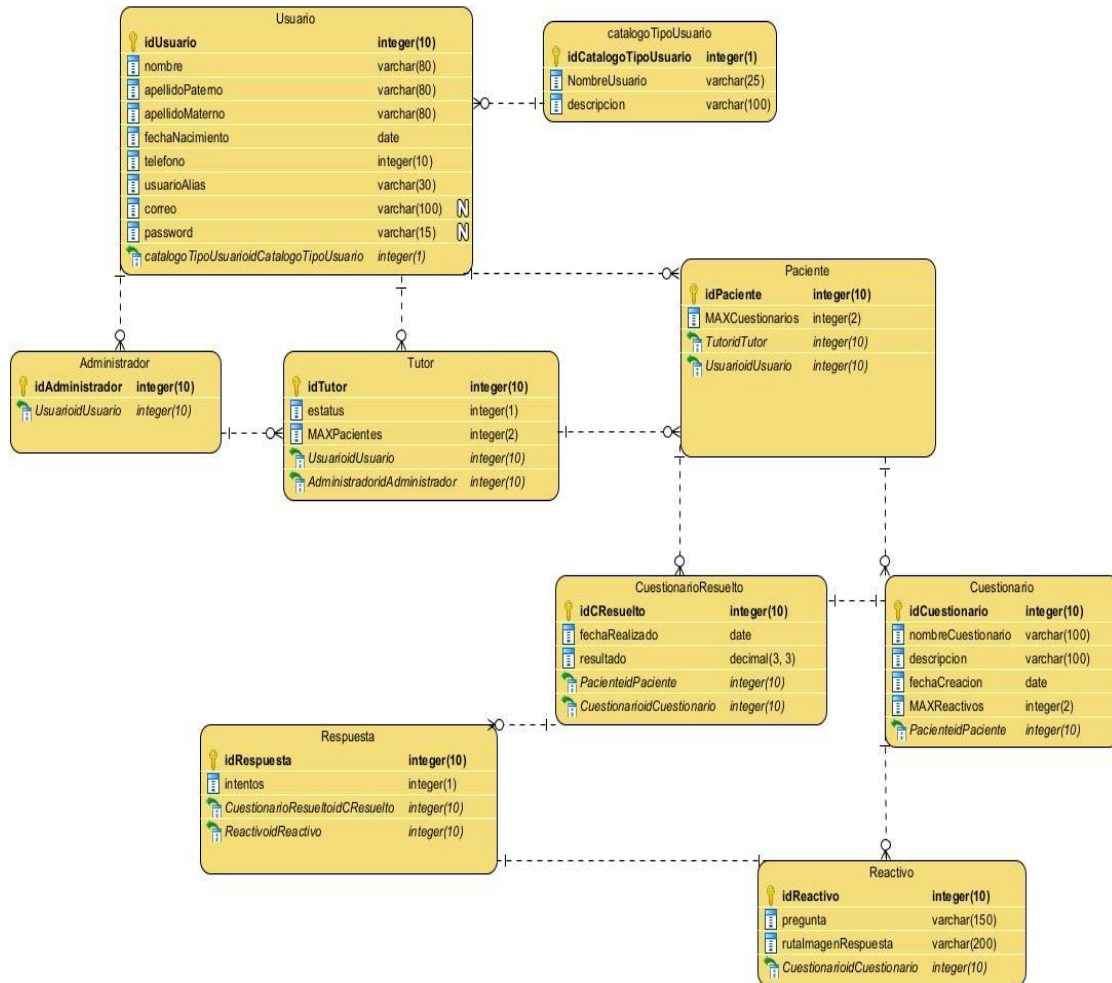
Se detecta que el campo descripcionUsuario es dependiente de tipoUsuario por lo cual se procede a generar: catalogoTipoUsuario(Img 11). Esto para cumplir con 3NF sobre la transitividad de los campos. Obtenido así (Img 12)

Usuario		
	idUsuario	integer(10)
	nombre	varchar(80)
	apellidoPaterno	varchar(80)
	apellidoMaterno	varchar(80)
	fechaNacimiento	date
	telefono	integer(10)
	usuarioAlias	varchar(30)
	correo	varchar(100) N
	password	varchar(15) N
	tipoUsuario	varchar(20)
	descripcionUsuario	varchar(100)

Img 10 Tabla Usuario Propio

catalogoTipoUsuario		
	idCatalogoTipoUsuario	integer(1)
	NombreUsuario	varchar(25)
	descripcion	varchar(100)

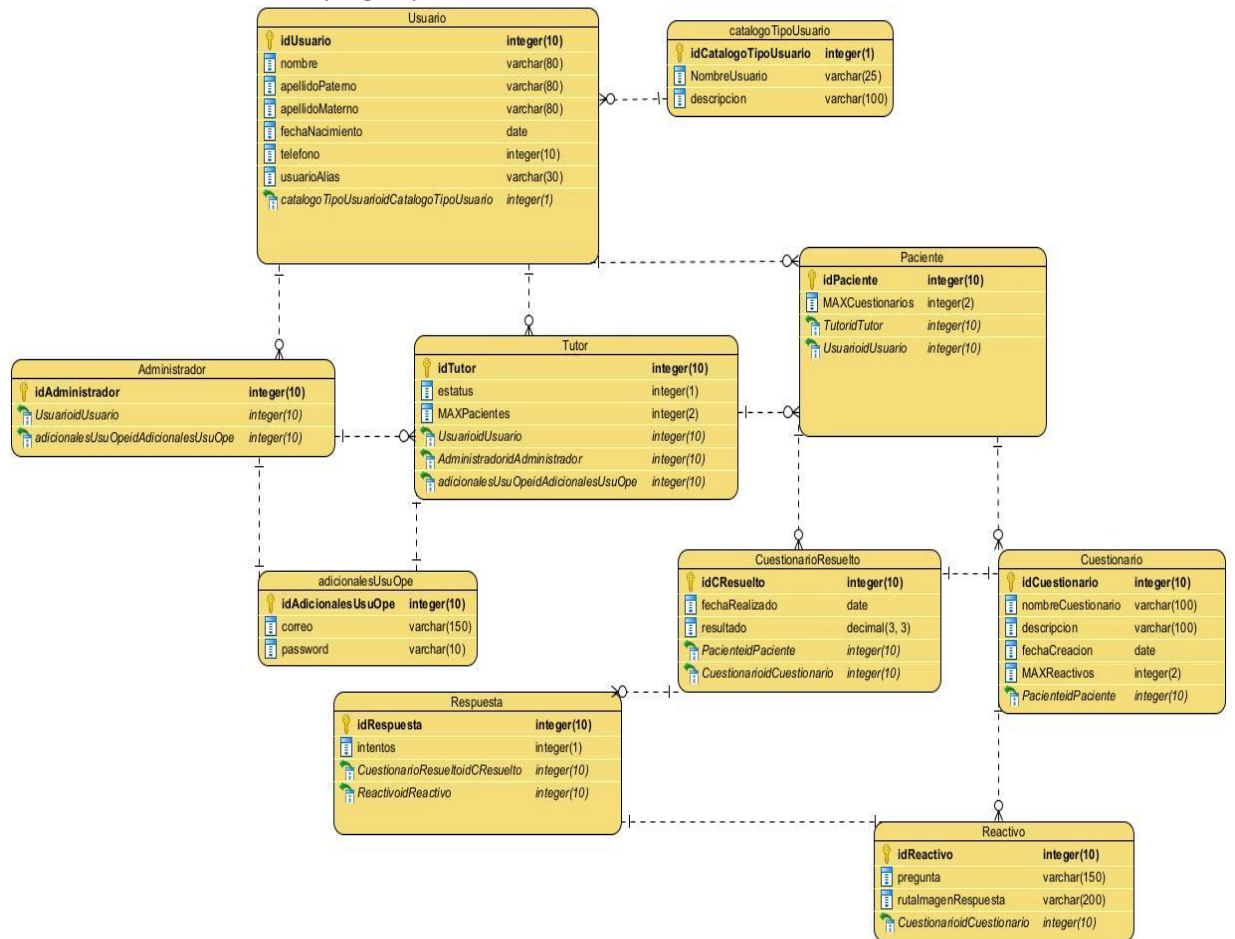
Img 11 Catalogo tipoUsuario Propio



Img 12 Diagrama Relacional v3 Propio

Por último, procedemos a realizar un ajuste en la tabla Usuario de la (Img 12) ya que detectamos que PUEDE tener campos nulos. Una tabla debe soportar estos valores nulos, pero se hace la propuesta de generar la nueva tabla adicionalesUsuOpe (datos adicionales para usuarios operativos) esta acción puede ser discutible bajo diferentes literaturas.

Obteniendo finalmente (Img 13)



Img 13 Diagrama Relacional v4 Propio

Además de que para el tema de seguridad será más eficiente y práctico encriptar solo la tabla adicionalUsuOpe que toda la tabla Usuario.

Capítulo 4. Implementación y pruebas del sistema

Para la implementación del sistema trabajaremos bajo el modelo MVC

Modelo-Vista-Controlador.

Este modelo de diseño se eligió debido a que tener por capas un sistema, trae múltiples ventajas, como mayor entendimiento en el diseño, menor coste de mantenimiento es de los modelos mas usados en la actualidad por tal motivo se puede introducir a la industria más rápidamente y su tiempo de implementación es menor.

4.1 Implementación de la base de datos

Como motor de base de datos ocuparemos MySQL, ya que nos permite la creación de BD basado en el modelo entidad relación, es openSource y trabaja bajo sentencias SQL las cuales son un estándar en la industria dándonos la versatilidad de uso para su creación y mantenimiento.

En la (Img) se muestran las tablas creadas de la base de datos del sistema de ejercicios para débiles visuales.

```
mysql> use ev_a1;
Database changed
mysql> show tables;
+-----+
| Tables_in_ev_a1 |
+-----+
| adicionalesusuo |
| administrador   |
| catalogotipousu |
| cuestionario     |
| cuestionarioresu |
| paciente        |
| reactivo        |
| respuesta       |
| tutor          |
| usuario         |
+-----+
10 rows in set (0.00 sec)
```

Img 14 Creación de las tablas de la base de datos del sistema web [Propio]

En las siguientes (Img 15) imágenes se muestran los atributos de cada una de las tablas implementadas en la base de datos para el sistema web de ejercicios para débiles visuales.

```
mysql> desc adicionalesusuope;
```

Field	Type	Null	Key	Default	Extra
idAdicionalesUsuOpe	int	NO	PRI	NULL	auto_increment
correo	varchar(150)	NO		NULL	
password	varchar(10)	NO		NULL	

3 rows in set (0.01 sec)

```
mysql> desc administrador;
```

Field	Type	Null	Key	Default	Extra
idAdministrador	int	NO	PRI	NULL	auto_increment
UsuarioidUsuario	int	NO	MUL	NULL	
adicionalesUsuOpeidAdicionalesUsuOpe	int	NO	MUL	NULL	

3 rows in set (0.00 sec)

```
mysql> desc catalogotipousuario;
```

Field	Type	Null	Key	Default	Extra
idCatalogoTipoUsuario	int	NO	PRI	NULL	auto_increment
NombreUsuario	varchar(25)	NO		NULL	
descripcion	varchar(100)	NO		NULL	

3 rows in set (0.00 sec)

```
mysql> desc cuestionario;
```

Field	Type	Null	Key	Default	Extra
idCuestionario	int	NO	PRI	NULL	auto_increment
nombreCuestionario	varchar(100)	NO		NULL	
descripcion	varchar(100)	NO		NULL	
fechaCreacion	date	NO		NULL	
MAXReactivos	int	NO		NULL	
PacienteidPaciente	int	NO	MUL	NULL	

6 rows in set (0.01 sec)

```
mysql> desc cuestionarioresuelto;
```

Field	Type	Null	Key	Default	Extra
idCResuelto	int	NO	PRI	NULL	auto_increment
fechaRealizado	date	NO		NULL	
resultado	decimal(3,3)	NO		NULL	
PacienteidPaciente	int	NO	MUL	NULL	
CuestionarioidCuestionario	int	NO	MUL	NULL	

5 rows in set (0.00 sec)

```
mysql> desc paciente;
```

Field	Type	Null	Key	Default	Extra
idPaciente	int	NO	PRI	NULL	auto_increment
MAXCuestionarios	int	NO		NULL	
TutoridTutor	int	NO	MUL	NULL	
UsuarioidUsuario	int	NO	MUL	NULL	

4 rows in set (0.01 sec)

```
mysql> desc reactivo;
```

Field	Type	Null	Key	Default	Extra
idReactivo	int	NO	PRI	NULL	auto_increment
pregunta	varchar(150)	NO		NULL	
rutaImagenRespuesta	varchar(200)	NO		NULL	
CuestionarioidCuestionario	int	NO	MUL	NULL	

4 rows in set (0.01 sec)

```
mysql> desc respuesta;
```

Field	Type	Null	Key	Default	Extra
idRespuesta	int	NO	PRI	NULL	auto_increment
intentos	int	NO		NULL	
CuestionarioResueltoidCResuelto	int	NO	MUL	NULL	
ReactivoReactivo	int	NO	MUL	NULL	

4 rows in set (0.00 sec)

```
mysql> desc tutor;
```

Field	Type	Null	Key	Default	Extra
idTutor	int	NO	PRI	NULL	auto_increment
estatus	int	NO		NULL	
MAXPacientes	int	NO		NULL	
UsuarioidUsuario	int	NO	MUL	NULL	
AdministradoridAdministrador	int	NO	MUL	NULL	
adicionalesUsuoPeidAdicionalesUsuoPe	int	NO	MUL	NULL	

6 rows in set (0.01 sec)

```
mysql> desc usuario;
```

Field	Type	Null	Key	Default	Extra
idUsuario	int	NO	PRI	NULL	auto_increment
nombre	varchar(80)	NO		NULL	
apellidoPaterno	varchar(80)	NO		NULL	
apellidoMaterno	varchar(80)	NO		NULL	
fechaNacimiento	date	NO		NULL	
telefono	int	NO		NULL	
usuarioAlias	varchar(30)	NO		NULL	
catalogoTipoUsuarioidCatalogoTipoUsuario	int	NO	MUL	NULL	

8 rows in set (0.00 sec)

Img 15 Descripción de las tablas ejercicios e imágenes propuestas para la base de datos del sistema web.
[Propio]

Usando los siguientes Scripts.

```
CREATE TABLE Usuario (idUsuario int(10) NOT NULL AUTO_INCREMENT,
nombre varchar(80) NOT NULL, apellidoPaterno varchar(80) NOT NULL,
apellidoMaterno varchar(80) NOT NULL, fechaNacimiento date NOT NULL,
telefono int(10) NOT NULL, usuarioAlias varchar(30) NOT NULL,
catalogoTipoUsuarioidCatalogoTipoUsuario int(1) NOT NULL, PRIMARY
KEY (idUsuario)) ENGINE=InnoDB;
```

```
CREATE TABLE Tutor (idTutor int(10) NOT NULL AUTO_INCREMENT, estatus
int(1) NOT NULL, MAXPacientes int(2) NOT NULL, UsuarioidUsuario
int(10) NOT NULL, AdministradoridAdministrador int(10) NOT NULL,
adicionalesUsuoPeidAdicionalesUsuoPe int(10) NOT NULL, PRIMARY KEY
(idTutor)) ENGINE=InnoDB;
```

```
CREATE TABLE Paciente (idPaciente int(10) NOT NULL AUTO_INCREMENT,
MAXCuestionarios int(2) NOT NULL, TutoridTutor int(10) NOT NULL,
```

```
UsuarioidUsuario int(10) NOT NULL, PRIMARY KEY (idPaciente))
ENGINE=InnoDB;
```

```
CREATE TABLE Cuestionario (idCuestionario int(10) NOT NULL
AUTO_INCREMENT, nombreCuestionario varchar(100) NOT NULL,
descripcion varchar(100) NOT NULL, fechaCreacion date NOT NULL,
MAXReactivos int(2) NOT NULL, PacienteidPaciente int(10) NOT NULL,
PRIMARY KEY (idCuestionario)) ENGINE=InnoDB;
```

```
CREATE TABLE Reactivo (idReactivo int(10) NOT NULL AUTO_INCREMENT,
pregunta varchar(150) NOT NULL, rutaImagenRespuesta varchar(200) NOT
NULL, CuestionarioidCuestionario int(10) NOT NULL, PRIMARY KEY
(idReactivo)) ENGINE=InnoDB;
```

```
CREATE TABLE CuestionarioResuelto (idCResuelto int(10) NOT NULL
AUTO_INCREMENT, fechaRealizado date NOT NULL, resultado decimal(3,
3) NOT NULL, PacienteidPaciente int(10) NOT NULL,
CuestionarioidCuestionario int(10) NOT NULL, PRIMARY KEY
(idCResuelto)) ENGINE=InnoDB;
```

```
CREATE TABLE Respuesta (idRespuesta int(10) NOT NULL AUTO_INCREMENT,
intentos int(1) NOT NULL, CuestionarioResueltoidCResuelto int(10)
NOT NULL, ReactivoidReactivo int(10) NOT NULL, PRIMARY KEY
(idRespuesta)) ENGINE=InnoDB;
```

```
CREATE TABLE catalogoTipoUsuario (idCatalogoTipoUsuario int(1) NOT
NULL AUTO_INCREMENT, NombreUsuario varchar(25) NOT NULL, descripcion
varchar(100) NOT NULL, PRIMARY KEY (idCatalogoTipoUsuario))
ENGINE=InnoDB;
```

```
CREATE TABLE Administrador (idAdministrador int(10) NOT NULL
AUTO_INCREMENT, UsuarioidUsuario int(10) NOT NULL,
adicionalesUsuOpeidAdicionalesUsuOpe int(10) NOT NULL, PRIMARY KEY
(idAdministrador)) ENGINE=InnoDB;
```

```
CREATE TABLE adicionalesUsuOpe (idAdicionalesUsuOpe int(10) NOT NULL
AUTO_INCREMENT, correo varchar(150) NOT NULL, password varchar(10)
NOT NULL, PRIMARY KEY (idAdicionalesUsuOpe)) ENGINE=InnoDB;
```

```
ALTER TABLE Paciente ADD CONSTRAINT FKPaciente300880 FOREIGN KEY
(TutoridTutor) REFERENCES Tutor (idTutor);
```

```
ALTER TABLE Cuestionario ADD CONSTRAINT FKCuestionar385867 FOREIGN
KEY (PacienteidPaciente) REFERENCES Paciente (idPaciente);
```

```
ALTER TABLE Reactivo ADD CONSTRAINT FKReactivo500414 FOREIGN KEY
(CuestionarioidCuestionario) REFERENCES Cuestionario
(idCuestionario);
```

```
ALTER TABLE Respuesta ADD CONSTRAINT FKRespuesta853838 FOREIGN KEY
(CuestionarioResueltoidCResuelto) REFERENCES CuestionarioResuelto
(idCResuelto);
```

```
ALTER TABLE CuestionarioResuelto ADD CONSTRAINT FKCuestionar709218
FOREIGN KEY (PacienteidPaciente) REFERENCES Paciente (idPaciente);
```

```
ALTER TABLE Usuario ADD CONSTRAINT FKUsuario629813 FOREIGN KEY
(catalogoTipoUsuarioidCatalogoTipoUsuario) REFERENCES
catalogoTipoUsuario (idCatalogoTipoUsuario);
```

```
ALTER TABLE Administrador ADD CONSTRAINT FKAdministra539211 FOREIGN
KEY (UsuarioidUsuario) REFERENCES Usuario (idUsuario);
ALTER TABLE Paciente ADD CONSTRAINT FKPaciente614682 FOREIGN KEY
(UsuarioidUsuario) REFERENCES Usuario (idUsuario);
```

```
ALTER TABLE Tutor ADD CONSTRAINT FKTutor999170 FOREIGN KEY
(UsuarioidUsuario) REFERENCES Usuario (idUsuario);
```

```
ALTER TABLE Tutor ADD CONSTRAINT FKTutor622859 FOREIGN KEY
(AdministradoridAdministrador) REFERENCES Administrador
(idAdministrador);
```

```
ALTER TABLE CuestionarioResuelto ADD CONSTRAINT FKCuestionar94772
FOREIGN KEY (CuestionarioidCuestionario) REFERENCES Cuestionario
(idCuestionario);
```

```
ALTER TABLE Respuesta ADD CONSTRAINT FKRespuesta955713 FOREIGN KEY
(ReactivoidReactivo) REFERENCES Reactivo (idReactivo);
```

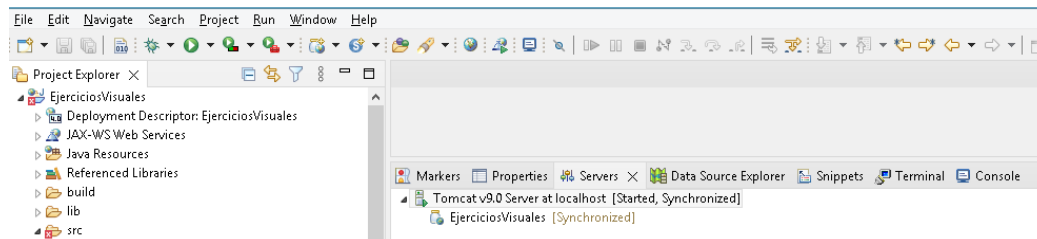
```
ALTER TABLE Administrador ADD CONSTRAINT FKAdministra808124 FOREIGN
KEY (adicionalesUsuOpeidAdicionalesUsuOpe) REFERENCES
adicionalesUsuOpe (idAdicionalesUsuOpe);
```

```
ALTER TABLE Tutor ADD CONSTRAINT FKTutor348165 FOREIGN KEY
(adicionalesUsuOpeidAdicionalesUsuOpe) REFERENCES adicionalesUsuOpe
(idAdicionalesUsuOpe);
```

4.2 Implementación de las funcionalidades del sistema

La implementación se realizó sobre lenguaje Java sobre IDE Eclipse, NetBeans sobre servidor Tomcat 9 usando JSP con Hibernate.

<http://localhost:8080/EjerciciosVisuales/>



Configuración Hibernate:

```

1 <?xml version="1.0" encoding="utf-8" ?>
2 <!--
3  AUTOR VICTOR CASTILLO
4  PROYECTO TESINA EV [Ejercicios Visuales]
5  -->
6 <!DOCTYPE hibernate-configuration PUBLIC
7  "-//Hibernate/Hibernate Configuration DTD 3.0//EN"
8  "http://www.hibernate.org/dtd/hibernate-configuration-3.0.dtd">
9 <hibernate-configuration>
10 <session-factory>
11 <!-- properties -->
12 <property name="dialect">org.hibernate.dialect.MySQLInnoDBDialect</property>
13 <property name="connection.driver_class">com.mysql.cj.jdbc.Driver</property>
14 <property name="connection.url">jdbc:mysql://localhost/ev_a1</property>
15 <property name="connection.username">root</property>
16 <property name="connection.password">root</property>
17 <property name="hibernate.connection.provider_class">org.hibernate.service.jdbc.connections.internal.C3P0ConnectionProvider</property>
18 <property name="hibernate.c3p0.acquire_increment">1</property>
19 <property name="hibernate.c3p0.idle_test_period">0</property>
20 <property name="hibernate.c3p0.min_size">1</property>
21 <property name="hibernate.c3p0.max_size">15</property>
22 <property name="hibernate.c3p0.timeout">0</property>
23 <property name="show_sql">>false</property>
24 <property name="query.substitutions">>true 1, false 0, yes 'Y', no 'N'</property>
25 <!-- mapping files -->
26 <mapping resource="ormmapping/AdicionalesUsuOpe.hbm.xml" />
27 <mapping resource="ormmapping/Administrador2.hbm.xml" />
28 <mapping resource="ormmapping/CatalogoTipoUsuario.hbm.xml" />
29 <mapping resource="ormmapping/Cuestionario2.hbm.xml" />
30 <mapping resource="ormmapping/CuestionarioResuelto2.hbm.xml" />
31 <mapping resource="ormmapping/Paciente2.hbm.xml" />
32 <mapping resource="ormmapping/Reactivo.hbm.xml" />
33 <mapping resource="ormmapping/Respuesta2.hbm.xml" />
34 <mapping resource="ormmapping/Tutor2.hbm.xml" />
35 <mapping resource="ormmapping/Usuario2.hbm.xml" />
36 </session-factory>
37 </hibernate-configuration>
38

```

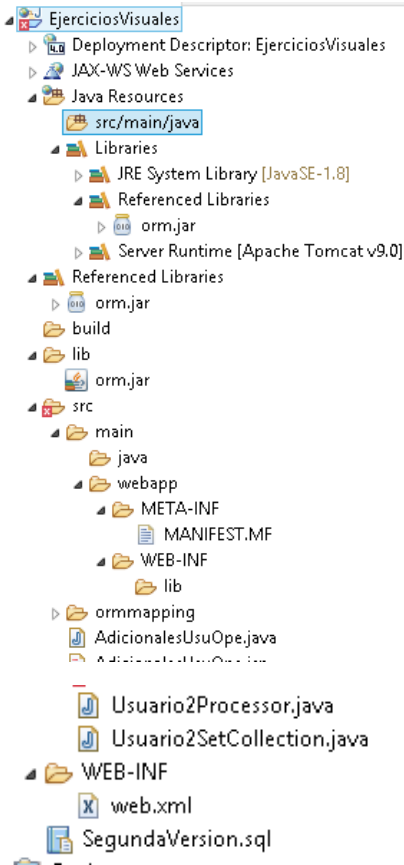
Muestra del Mapeo Hibernate:

```

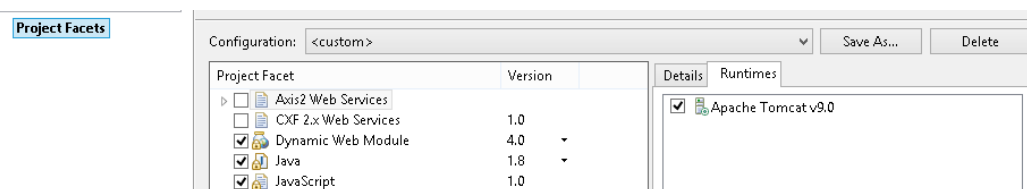
AdicionalesUsuOpe.hbm.xml X
1 <!--
2  AUTOR VICTOR CASTILLO
3  PROYECTO TESINA EV [Ejercicios Visuales]
4  -->
5  -->
6 <!DOCTYPE hibernate-mapping PUBLIC "-//Hibernate/Hibernate Mapping DTD 3.0//EN" "http://www.hibernate.org/dtd/hibernate-mapping-3.0.dtd">
7 <hibernate-mapping>
8 <class name="AdicionalesUsuOpe" table="adicionalesUsuOpe" lazy="false">
9 <id name="idAdicionalesUsuOpe" column="idAdicionalesUsuOpe" type="integer" unsaved-value="0">
10 <generator class="native">
11 </generator>
12 </id>
13 <property name="correo" column="correo" type="string" length="150" not-null="true" lazy="false"/>
14 <property name="password" column="password" type="string" length="10" not-null="true" lazy="false"/>
15 <one-to-one name="administrador2" class="Administrador2" cascade="save-update,lock" property-ref="adicionalesUsuOpeidAdicional
16 <one-to-one name="tutor2" class="Tutor2" cascade="save-update,lock" property-ref="adicionalesUsuOpeidAdicionalesUsuOpe" lazy="
17 </class>
18 </hibernate-mapping>
19
Administrador2.hbm.xml X
1 <?xml version="1.0" encoding="utf-8" ?>
2 <!--
3  AUTOR VICTOR CASTILLO
4  PROYECTO TESINA EV [Ejercicios Visuales]
5  -->
6 <!DOCTYPE hibernate-mapping PUBLIC "-//Hibernate/Hibernate Mapping DTD 3.0//EN" "http://www.hibernate.org/dtd/hibernate-mapping-3.0.dtd">
7 <hibernate-mapping>
8 <class name="Administrador2" table="Administrador" lazy="false">
9 <id name="idAdministrador" column="idAdministrador" type="integer" unsaved-value="0">
10 <generator class="native">
11 </generator>
12 </id>
13 <many-to-one name="usuarioidUsuario" cascade="lock" column="UsuarioidUsuario" class="Usuario2" not-null="true" lazy="no-proxy"
14 </many-to-one>
15 <many-to-one name="adicionalesUsuOpeidAdicionalesUsuOpe" column="adicionalesUsuOpeidAdicionalesUsuOpe" class="AdicionalesUsuOpe
16 </many-to-one>
17 <set name="ORM_Tutor2" lazy="true" cascade="save-update,lock" inverse="true">
18 <key column="AdministradoridAdministrador" not-null="true"/>
19 <one-to-many class="Tutor2"/>
20 </set>
21 </class>
22 </hibernate-mapping>
23

```

Estructura del Proyecto java:



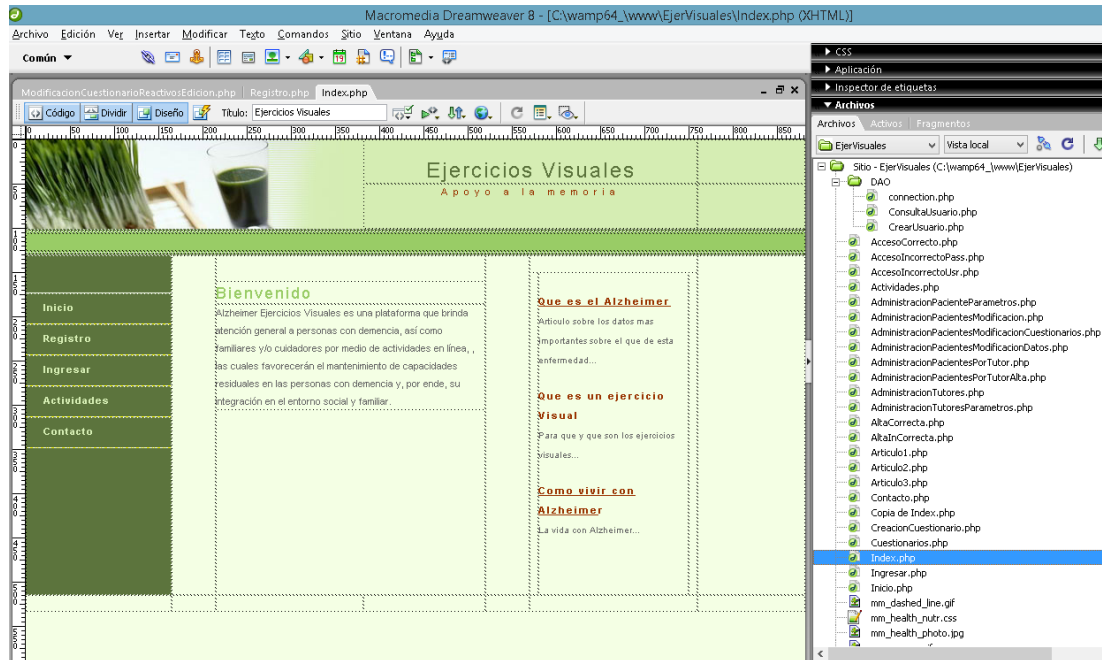
Facetas



4.3 Maquetado y pruebas de funcionalidad del sistema

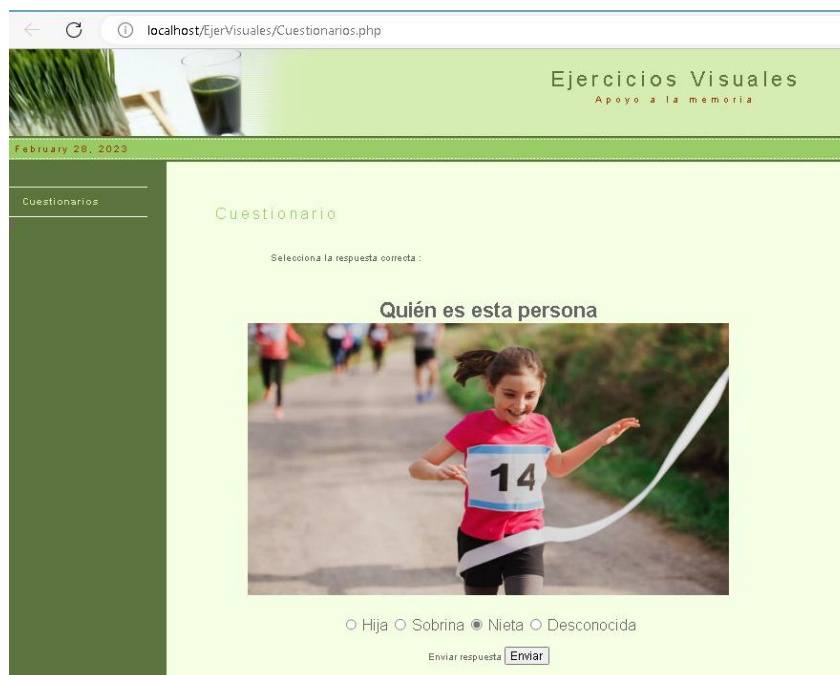
Prototipos de vistas

El maquetado de estas vistas es generado por el aplicativo Dreamweaver8 con esta estructura de manera general:



Se mostrarán las vistas de acuerdo con el caso de uso indicado.

CU1 RESOLVER CUESTIONARIOS: El paciente resuelve los cuestionarios que se le presentan.



CU2 Alta inicial del administrador: Cuando nace el sistema tiene un usuario administrador por default, se debe de actualizar los datos del admin default con los datos del administrador en turno.

← localhost/EjerVisuales/Registro.php

Ejercicios Visuales
Apoyo a la memoria

February 28, 2023

Registro

Te invitamos a que te registres:

Nombre

Apellido Paterno

Apellido Materno

Dirección

Teléfono

Correo

Tipo Usuario

Alias

Password

[Que es el Alzheimer](#)
Artículo sobre los datos mas importantes sobre el que de esta enfermedad...

[Que es un ejercicio Visual](#)
Para que y que son los ejercicios visuales...

[Como vivir con Alzheimer](#)
La vida con Alzheimer...

CU3 Activa/des Activa tutores: El tutor activa/des activa tutores para el acceso al sistema.

← localhost/EjerVisuales/AdministracionTutores

Ejercicios Visuales
Apoyo a la memoria

February 28, 2023

Administracion

Activacion/Desactivacion tutores

Activa o desactiva tutores::

Tutores

Nombre	Estatus	Edicion
Victor Castillo	☐	EDITAR
Rosa Isela	☒	EDITAR
Daniel Alberto	☒	EDITAR
Juan Perez	☐	EDITAR

Enviar acción

CU4 Edición de parámetros de los tutores: El administrador puede editar los parámetros de los tutores.

localhost/EjerVisuales/AdministracionTutoresParametros

Ejercicios Visuales
Apoyo a la memoria

February 28, 2023

Administracion

Edición parametros

Edita parametros del tutor:

Victor Castillo

MAX Pacientes	MAX Cuestionarios	MAX Preguntas/Cuestionario
3 ▼	3 ▼	5 ▼

Enviar acción

CU5 Logueo del Administrador: El administrador accede al sistema.

localhost/EjerVisuales/Ingresar.php

Ejercicios Visuales
Apoyo a la memoria

February 28, 2023

Inicio
Registro
Ingresar
Actividades
Contacto

Ingresar

Ingresa tus datos de acceso:

Usuario

Password

[Que es el Alzheimer](#)
Articulo sobre los datos mas importantes sobre el que de esta enfermedad...

[Que es un ejercicio Visual](#)
Para que y que son los ejercicios visuales...

[Como vivir con Alzheimer](#)
La vida con Alzheimer...

CU7 Config. de paciente inicial: El sistema configura con sus valores iniciales.

← → ↻ ⓘ localhost/EjerVisuales/AdministracionPacienteParametros

Ejercicios Visuales
Apoyo a la memoria

February 28, 2023

Administracion

Edicion parametros

Edita parametros del paciente:

Maria de la Luz Tomas

Número reactivos iniciales	Número de cuestionarios
5 ▼	3 ▼

Enviar acción

CU8 Alta de tutor: El tutor genera su registro en el sistema

← ↻ ⓘ localhost/EjerVisuales/Registro.php

Ejercicios Visuales
Apoyo a la memoria

February 28, 2023

Inicio

Registro

Ingresar

Actividades

Contacto

Registro

Te invitamos a que te registres:

Nombre

Apellido Paterno

Apellido Materno

Direccion

Telefono

Correo

Tipo Usuario **TUTOR ▼**

Alias

Password

[Que es el Alzheimer](#)
Articulo sobre los datos mas importantes sobre el que de esta enfermedad...

Que es un ejercicio Visual
Para que y que son los ejercicios visuales...

[Como vivir con Alzheimer](#)
La vida con Alzheimer...

Pacientes por tutor:



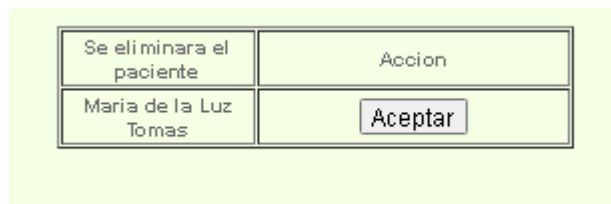
AGREGAR/EDITAR/ELIMINAR PACIENTE:

Agregar:





Eliminación de tutorado:



Modificación de tutorado:



Modificación de datos generales:

localhost/EjerVisuales/AdministracionPacientesModificacionDatos

Ejercicios Visuales
Apoyo a la memoria

February 28, 2023

Administracion

Administracion de pacientes

Edita datos generales del paciente:

Victorino Castillo Sebastian

Nombre

Apellido Paterno

Apellido Materno

Fecha nacimiento

Usuario Alias

Actualizar paciente

Modificación de cuestionarios:

localhost/EjerVisuales/AdministracionPacientesModificacionCuestionarios

Ejercicios Visuales
Apoyo a la memoria

March 1, 2023

Administracion

Administracion de pacientes

Edita cuestionarios del paciente:

Victorino Castillo Sebastian

Nombre cuestionario	Descripción	
		Agregar cuestionario
Familia	Preguntas relacionadas con la familia.	Editar Borrar
Lugares	Preguntas relacionadas con lugares conocidos.	Editar Borrar

CU12 Creación de cuestionarios: El tutor crea un cuestionario al paciente

← ↻ ⓘ localhost/EjerVisuales/CreacionCuestionario.php



Ejercicios Visuales

Apoyo a la memoria

March 1, 2023

Administracion

Administracion de cuestionarios

Se crea nombre y descripcion del cuestionario para el paciente :

Victorino Castillo

Nombre del cuestionario

Descripcion del cuestionario

CU13 Modificación de cuestionarios: El tutor modifica un cuestionario al paciente

← ↻ ⓘ localhost/EjerVisuales/ModificacionCuestionario.php



Ejercicios Visuales

Apoyo a la memoria

March 1, 2023

Administracion

Administracion de cuestionarios

Se modifica informacion del cuestionario para el paciente :

Victorino Castillo

Datos generales:

The screenshot shows a web browser at the URL `localhost/EjerVisuales/ModificacionCuestionario.php`. The page has a green header with the title "Ejercicios Visuales" and the subtitle "Apoyo a la memoria". Below the header, the date "March 1, 2023" is displayed. A sidebar on the left contains the menu item "Administracion". The main content area is titled "Administracion de cuestionarios" and includes the instruction "Se modifica nombre y descripcion del cuestionario para el paciente :". The patient's name, "Victorino Castillo", is prominently displayed. Below this, there are two input fields: "Nombre del cuestionario" with the value "Familia" and "Descripcion del cuestionario" with the value "Preguntas de la Familia". A red button labeled "Modificar cuestionario" is positioned below the description field.

CU14 Eliminación de cuestionarios: El tutor elimina un cuestionario al paciente

Se eliminara el cuestionario	Accion
Familia	Borrar

CU15 Creación de reactivos: El tutor crea un reactivo al cuestionario

Reactivos

← ↻ ⓘ localhost/EjerVisuales/ModificacionCuestionarioReactivos

Ejercicios Visuales
Apoyo a la memoria

March 1, 2023

Administracion

Administracion de cuestionarios

Se edita reactivos del cuestionario para el paciente :

Victorino Castillo

cuestionario:Familia

Reactivo	Nuevo Reactivo	
¿Quién es esta persona?	Editar	Borrar
¿Dónde es este lugar?	Editar	Borrar
¿Como se llama tu esposa?	Editar	Borrar
¿Como se llama tu hijo?	Editar	Borrar
¿Como se llama tu hija?	Editar	Borrar

Borrado

Reactivo	Accion
¿Quién es esta persona?	Borrar

Edición de reactivo

← ↻ ⓘ localhost/EjerVisuales/ModificacionCuestionarioReactivosEdicion



Ejercicios Visuales
Apoyo a la memoria

March 1, 2023

Administración

Administración de cuestionarios

Se edita reactivos del cuestionario para el paciente :

Victorino Castillo

cuestionario:Familia

Reactivo	¿Quién es esta persona?	
Respuesta 1	Desconocida	
Respuesta 2	Hija	
Respuesta 3	Sobrina	
Respuesta 4*	Nieta	
Imagen relacionada	Elegir archivo No se ha seleccionado ningún archivo	
GUARDAR		

CU20 Tablero de tutor: Tablero con información de pacientes cuestionarios y pacientes.

← ↻ ⓘ localhost/EjerVisuales/TableroTutores



Ejercicios Visuales
Apoyo a la memoria

March 1, 2023

Tutores

Tablero General

Muestra resumen de pacientes y sus cuestionarios::

Nombre Paciente	Fecha Alta	Cuestionarios	Resultados
Victorino Castillo Sebastian	12 /12/2022	Familia Lugares	Ir
Maria de la Luz Tomas	21/12/2023	AUN NO TIENE CUESTIONARIOS	Ir

CU17 Reporte resultados por paciente: Generar un resumen de resultados por paciente



Ejercicios Visuales

Apoyo a la memoria

March 1, 2023

Tutorías

Tablero por paciente

Muestra resumen cuestionario del paciente:

Victorino Castillo Sebastian

Cuestionario	Fecha aplicacionn	Resultado	Errores
Lugares	12/12/2022	60%	Reactivo 3 Reactivo 1
Familia	24 /12/2022	80%	Reactivo 1
Lugares	30/12/2023	80%	Reactivo 3
Familia	01 /01/2023	100%	NA

Conclusiones y trabajo futuro

Hasta el momento tenemos la parte del análisis, diseño e inicios de la implementación, con el CORE y el prototipado, este documento representa la parte fundamental del sistema, donde faltaría la puesta en producción.

Como plan a futuro se tiene la finalización de la implantación en desarrollo para pasar a la etapa de testeo y poder ejecutar una batería de pruebas dar más certeza a su correcto funcionamiento.

Se tiene en mente el agregar el algoritmo flash Card a el cuestionario que se le presenta al paciente, esto con el objetivo de poder ayudar a recordar de mejor manera las respuestas de cada cuestionario.

Así también la implementación de un front más actualizado para dar una mejor estilizado al proyecto.

Gracias!!!!

Bibliografía

- [1] Andrea Slachevsky, P. F. (2007). Enfermedad del Alzheimer y otras demencias. Santiago de Chile: LOM Ediciones.
- [2] inn.salud.gob.mx. (14 de 03 de 2018). Recuperado el octubre de 2021, de <http://www.innn.salud.gob.mx/interna/medica/padecimientos/alzheimer.html>
- [3] Precius C. Godson, A. I. (2018). Los ultimos dias de la Demencia del Alzheimer. Bebelcube, Inc.
- [4] Abraham Gutiérrez, Ginés Bravo, PHP5, Alfaomega Ra-MA, México 2005.
- [5] Braude, Ingeniería de Software, una perspectiva orientada a objetos, Alfaomega, México 2003.
- [6] Booch, Grady / Rumbaugh, James / Jacobson, Ivar. E Proceso Unificado De Desarrollo De Software. Addison Wesley, 2000
- [7] Carlos Coronel, Steven Morris, Peter Rob. Bases de datos: diseño, implementación y administración, México 2002.
- [8] Elizabeth Freeman & Erick Freeman. Head First HTML with CSS & XHTML–2006.
- [9] Gabriel Baca Urbina, Proyectos de Sistemas de Información, Primera Edición Ebook, México 2015
- [10] Helma Spona, Programación de bases de datos con MySQL y PHP, Alfaomega, México, 2010.
- [11] Ian Gilfillan. La biblia MySQL. Editorial ANAYA, 2003.
- [12] Ian Sommerville, Ingeniería del software, Séptima edición, Pearson Educación. S.A. Madrid 2005.
- [13] Jacobo Pavon Puertas. Creación de un portal con PHP y MySQL.4ta Edición. Alfaomega. 2010.
- [14] Luke Welling, Laura Thomson. Desarrollo Web con PHP y MySQL Editorial Anaya. 2009
- [15] Philippe Kruchten, The Rational Unified Process An Introduction Third Edition, Addison-Wesley
- [16] Silberschatz, Korth, Sudarshan, Fundamentos de Diseño de Bases de Datos, quinta edición, Mc Graw Hill, España, 2007.
- [17] G. / J. R. J. / J. I. Booch, Proceso Unificado De Desarrollo De Software., Addison Wesley, 2000.

- [17] Steve Suehring, Janet Valade. PHP, MySQL, JavaScript & HTML5 All-in-One For Dummies. 1st Edición. 2013.
- [18] A. I. S. Precius C. Godson, Los ultimos dias de la Demencia del Alzheimer, Bebelcube, Inc., 2018.
- [19] P. F. L. J. P. A. Andrea Slachevsky, Enfermedad del Alzheimer y otras demencias, Santiago de Chile: LOM Ediciones, 2007.
- [20] K. S. Silberschatz, Fundamentos de Diseño de Bases de Datos, Quinta edición, ed., España: Mc Graw Hill, 2007.
- [21] E. F. & E. Freeman., Head First HTML with CSS & XHTML, 2006.
- [22] «ISES Instituto Superior de Estudios Sociales y Sociosanitarios,» 2021. [En línea]. Available: <https://www.isesinstituto.com/noticia/5-tecnicas-de-estimulacion-para-enfermos-de-alzheimer>. [Último acceso: 10 2021].
- [23] P. N. y. B. R. Eds, «SOFTWARE ENGINEERING,» de *Software Engineering: Report of a conference sponsored by the NATO Science Committee*, Garmisch, Germany, 1987.
- [24] R. S. P. y. B. R. Maxim, Software Engineering: A practitioner's approach, 8th ed. ed., New York, NY, USA: McGraw-Hill Education, 2015.
- [25] S. C. Florman, The Existential Pleasures of Engineering., New York: St. Martin's Press, 1976.
- [26] D. L. Parnas, Education for Computing Professionals, IEEE Computer 23, 1990, pp. 17-22.
- [27] G. Ford, 1990 SEI Report on Undergraduate Software Engineering Education, Pittsburgh, PA, USA, 1990.
- [28] B. e. B. A.U.S. Gustavo Torossi, «El Proceso Unificado de Desarrollo de Software,» Addison Wesley , PERSON EDUCACION, S.A., Madrid, 2000.
- [29] M. J. G. Fernández, UF1471: Base de datos relacionales y modelado de datos, EDITORIAL ELEARNING S.L. .