

Telegraph

Requirement Specification

Equipo:

Martínez González Luis Alejandro	lu1s.al3jandro@gmail.com	214588764
Pérez Peregrina Oscar Alejandro	oscarp.peregrina@gmail.com	214589228
Zárate Núñez Carlos Alberto	carlos.zn07@gmail.com	214404074

Contenido

Contenido	1
Agradecimientos	2
Sinopsis del problema	2
Solución propuesta.....	3
Alcance del sistema	3
Componentes del sistema.....	4
Especificaciones de las interfaces.....	5
Restricciones del sistema.....	6
Calendario revisado del proyecto.....	7
Tabla de Versionamiento.....	7

Agradecimientos

Todos los miembros de equipo queremos agradecer al maestro Gustavo Arrona Córdova por asignarnos este gran proyecto, además de confiar en nosotros para dicha tarea, darnos las oportunidades necesarias para poder avanzar en él, y las asesorías y correcciones a este, ya que, sin esa retroalimentación, este proyecto no sería lo que es. También queremos agradecerle al maestro Felipe Sención Echauri por su asesoría respecto a este proyecto, quien nos brindó sus conocimientos cada que lo requeríamos. También a nuestras familias, que siempre nos apoyaron en nuestros estudios y nos alentaron a no dejarlos de lado, además de apoyarnos para poder centrarnos en aprender.

Agradecemos a la benemérita Universidad de Guadalajara, nuestra casa de estudios, principalmente al Centro Universitario de Ciencias Exactas e Ingenierías, ya que en ella fue donde aprendimos y descubrimos lo apasionante que puede ser el mundo de la informática. Gracias a cada uno de los profesores que nos transmitieron su conocimiento y siempre nos alentaron a seguir aprendiendo. Por último, pero no menos importante, le damos las gracias a cada uno de los integrantes del equipo, ya que sin el trabajo en equipo nada de esto hubiese sido posible.

Sinopsis del problema

Las aplicaciones de mensajería han tomado fuerza en los últimos 20 años, pero muchos de estos servicios necesitan de mucho espacio en el disco o en caso de ser en la nube, dependen mucho de la velocidad de la red, dejando a los consumidores con un mal sabor de boca. Dentro del mundo de las aplicaciones de mensajería, son muy pocas las que contienen un método de pago para renovar la suscripción, por lo que queremos ofrecer otra alternativa de manera gratuita para la mensajería dentro de computadoras.

La presentación también es importante; pensamos que las aplicaciones de moda utilizan diseños que si bien, son novedosos y estéticos, han perdido la esencia de aquellas aplicaciones de mensajería de antaño, por lo que, ofrecemos una interfaz que mezcla las nuevas tendencias con una presentación que apela a la nostalgia.

Las aplicaciones de este tipo tienen un gran impacto las comunicaciones, pero hay muchos inconvenientes en este tipo de sistemas, los cuales se encuentran listados abajo:

- 1.- Requerimientos para una buena conexión.
- 2.- Limitaciones de uso de la aplicación por persona.
- 3.- Hay muchas funciones que el usuario promedio no utiliza o ni siquiera tiene conciencia de.
- 4.- La falta de mantenimiento del sistema provoca que se vuelvan obsoletas en un determinado periodo de tiempo.

Solución propuesta

Con este proyecto, buscamos presentar una aplicación de mensajería más simple y amigable para su uso cotidiano, atacando la mayoría de los problemas mencionados anteriormente, por medio de un sistema de Mensajería Instantánea (IM). El sistema tiene una instalación simple y limpia, presenta una interfaz bastante intuitiva, además de ofrecer funciones básicas pero concisas.

Decidimos incursionar en un terreno con competencia bastante dura, sin embargo, nuestro principal objetivo no es competir directamente con las empresas que lideran actualmente este ámbito. Se busca ofrecer una aplicación bastante diferente, diseñada para aquellos que buscan una aplicación ágil y eficaz, con alta precisión en lo que ofrece. Para aquellos que no requieren otras características que vuelven un tanto complejo el manejo de la aplicación.

Alcance del sistema

Este proyecto será basado en dos grandes paradigmas de la programación, por un lado, está la programación orientada a objetos y por el otro, la programación estructural. Algunas de las ventajas sobre la programación orientada a objetos se pueden enlistar de la siguiente manera:

- Los objetos pueden tener varios atributos, por ejemplo, que lea un sensor y a la vez encienda.
- Son más fáciles de entender los códigos.
- La ejecución del programa es rápida y sencilla, todo se encuentra en una sola ventana.
- Los componentes se pueden reutilizar.
- Facilidad de mantenimiento y modificación de los objetos existentes.

Compu Mundo Híper Mega Red

- Una estructura modular clara se puede obtener, la cual no revelará el mecanismo detrás del diseño.
- Se proporciona un buen marco que facilita la creación de rica interfaz gráfica de usuario aplicaciones (GUI).
- Se acopla bien a la utilización de bases de datos, debido a la correspondencia entre las estructuras.

Y como se pudo mencionar en la última de las muchas ventajas, nuestro proyecto se basa en una base de datos de tipo JSON, por lo tanto, orientarlo a objetos nos resultará muy sencillo a la hora de poder adaptarlo.

Sin embargo, no será el único paradigma que utilizaremos, ya que, usaremos la programación estructural de la cual podemos tener los siguientes beneficios:

- Datos separados del Diseño.
- Entendimiento del programa en el mundo real.
- Fácil entendimiento de la lógica del programa.
- Fácil documentación y diseño del programa.
- Dinamismo en el manejo de los datos.
- Facilidad en el mantenimiento y expansión.

Componentes del sistema

Este sistema manejará una base de datos la cual tendrá operaciones como: crear usuario, acceder, añadir contacto, mandar mensaje, entre otras más para la funcionalidad del sistema en el caso de la ventana de acceso o la ventana principal.

Administrador:

- Usuarios
- Iniciar sistema

Usuario:

- Nombre de usuario
- Teléfono
- Contraseña
- Contactos

Contacto:

- Nombre de contacto
- Mensajes

Mensaje:

- Texto
- Fecha

Acceso al Sistema:

- Crear usuario
 - *Nombre de usuario*
 - *Teléfono*
 - *Contraseña*
- Acceder usuario
 - *Nombre de usuario*
 - *Contraseña*

Sistema:

- Añadir contacto
 - *Nombre de usuario*
 - *Nombre de contacto*
- Mandar Mensaje
 - *Nombre de usuario*
 - *Nombre de contacto*
 - *Texto*
 - *Fecha*
- Buscar contacto
 - *Nombre de contacto*
- Buscar mensaje
 - *Texto*

Especificaciones de las interfaces

Se utilizarán reglas básicas para la interconectividad de todos los objetos y los vectores que se utilicen, como pueden ser:

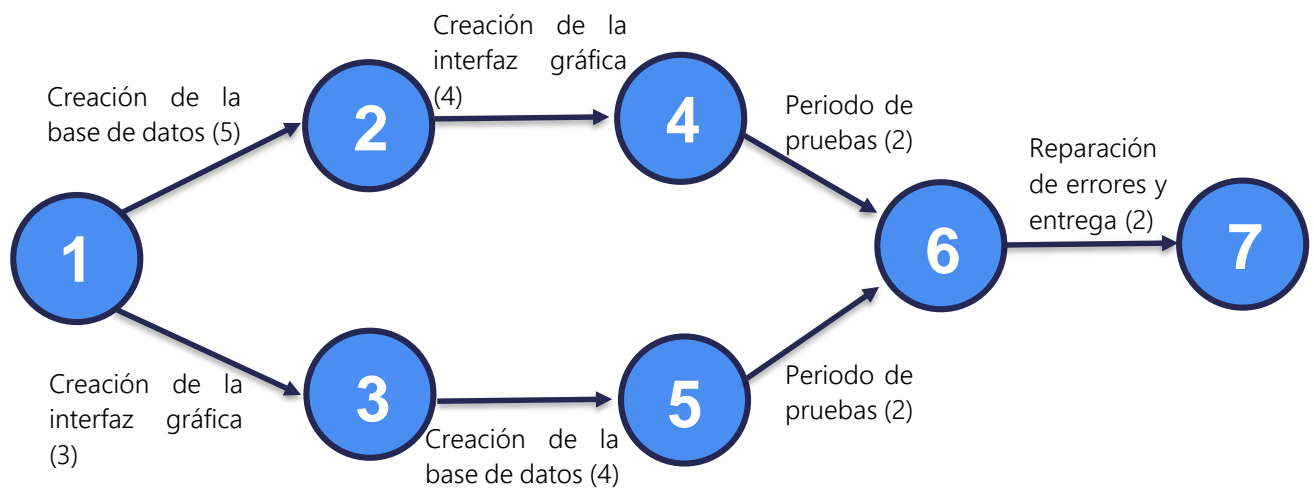
- La clase Administrador se encarga de todas las operaciones que se realicen por el usuario.
- La clase Usuario sólo será manipulada en la interfaz de acceso del sistema.
- La clase Contacto sólo será manipulada en la interfaz principal del sistema.
- La clase Mensaje sólo será manipulada en la interfaz principal del sistema.
- En la parte superior derecha de la interfaz principal se mostrará el nombre de usuario y teléfono del usuario actual.
- Las búsquedas sólo serán posibles en la interfaz principal del sistema.

- En la interfaz principal, se mostrarán todos los usuarios existentes, con la posibilidad de añadir a cualquiera de ellos.
- La clase Mensaje funcionará sólo si el contacto ha sido añadido.
- Siempre que se seleccione un contacto que haya sido agregado, se mostrará la conversación que tenga con él.

Restricciones del sistema

- Al ser un sistema que depende de contenido almacenado en la base de datos, es necesario que este archivo siempre exista en el almacenamiento interno del dispositivo. No es necesario el acceso a internet, pero sí es necesario contar con una cuenta en el sistema, de modo que el usuario pueda acceder a este.
- La ejecución del programa es fluida, ya que el sistema no tiene un peso elevado, así como el tiempo de ejecución de cada una de las funciones, utilizando métodos que, si bien no son los más eficaces a la hora de realizar búsquedas, son completamente funcionales.
- No existe límite como tal en cuanto a la cantidad de usuarios que puedan ser agregados, sino que se limita a la cantidad de usuarios existentes en la base de datos. Además, un usuario no puede ser agregado más de una vez ni removido. Tampoco se puede agregar a sí mismo como contacto.
- No existe un número de usuarios máximos, limitado únicamente por el tamaño en disco que genere. No puede haber usuarios con el mismo nombre, pero sí con el mismo número y contraseña.
- No existe límite para el número de mensajes enviados a un contacto o mensajes recibidos; los mensajes pueden ser iguales pero la fecha puede variar. Son ordenados respecto a la fecha que fueron enviados.
- El programa no es en tiempo real. Para poder recibir mensajes, deben ser enviados por otro usuario, permitiendo el único acceso a un usuario.

Calendario revisado del proyecto



Unidad de Tiempo: Semanas.

Tabla de Versionamiento

1.0	Versión Inicial	Pérez Peregrina Oscar Alejandro	17/Abril/2019
1.1	Revisión y ajustes	Zárate Núñez Carlos Alberto	27/Abril/2019
1.2	Versión final	González Martínez Luis Alejandro	28/Abril/2019