

Universidad del Valle de Guatemala
Facultad de Ciencias y Humanidades
Departamento de Computación
Ingeniería de Software 1
Lic. Lynette García
Sección 10



meeting

(corte 3)

Orlando Cabrera 19943
Diana Corado 191025
Fernando Garavito 18071
Pablo Méndez 19195
Eduardo Ramírez Herrera 19946

Guatemala, Ciudad de Guatemala 5 de abril del 2021

Índice

Resumen	1
Introducción	1
Design Thinking	2
Prototipos	2
Análisis	3
Requisitos funcionales	3
Elabore una lista de requisitos funcionales de su sistema	3
Historias de usuario	3
Para el Backend	4
Persistencia de datos:	6
Diseño	7
Selección de la tecnología a utilizar basándose en los requisitos no funcionales:	7
-Diana Zaray Corado López-	13
-Orlando Osberto Cabrera Mejía-	13

Resumen

El tema del proyecto será una red social en donde estudiantes de la universidad del Valle podrán conocer personas que tengan gustos similares. El proyecto estará enfocado hacia los estudiantes de primer año de la universidad del Valle de Guatemala, esto se decidió así ya que la mayoría de las veces, las personas no conocen a nadie de la universidad cuando empiezan sus estudios universitarios.

El principal objetivo del proyecto es permitir a los estudiantes de primer año conocer personas con hobbies similares y puedan crear relaciones de amistades con distintas personas.

Como objetivos secundarios está sacar a los estudiantes de primer año de su zona de confort y que conozcan gente nueva.

Introducción

Los estudiantes de primer año de la universidad experimentan diversos cambios, entre ellos se encuentra el cambio de amigos y el nuevo reto nuevos grupos sociales, actualmente la *nueva normalidad* ha vuelto más difícil el poder conocer nuevas personas dentro de la universidad, sumado a esto, cada persona cuenta con características que les permiten ser extrovertidos o introvertidos, facilitando o complicando un poco más la manera de interactuar con las demás personas.

A diferencia de las clases presenciales, en las cuales los estudiantes tenían la oportunidad de compartir más ideas mientras se estaba recibiendo una clase, en las clases virtuales es más condicionada la interacción entre estudiantes, por lo que se vuelve más difícil hacer nuevos amigos y entablar conversaciones. Debido a esto a continuación se detalla el desarrollo de un proyecto que permita a los estudiantes de primer año de la Universidad del Valle de Guatemala a conocer e interactuar con nuevas personas que tengan hobbies y características en común. El principal objetivo de este proyecto es lograr que los estudiantes, en medio de la pandemia, puedan encontrar una manera más dinámica y entretenida de conocer nuevos amigos, así mismo el poder formar grupos sociales de personas con características en común que luego puedan trabajar en equipo en diversos proyectos de clase

Design Thinking

1. Prototipos

- a. Elaborar al menos dos versiones más de prototipos del sistema a desarrollar. Haga prototipos para diferentes historias de usuario.
- b. Mostrar evidencias de cada versión del prototipo. Esto incluye evidencias de que se le mostró a las personas entrevistadas. (Fotos y/o vídeos)
- c. Elaborar una lista de cambios hechos a cada versión de prototipo de acuerdo a observaciones hechas por futuros usuarios entrevistados.
 - i. Versión 1
 1. Falta ver la pantalla de iniciar sesión
 2. El color de fondo si bien es blanco, sería mejor utilizar un color un poco más oscuro
 3. Al momento de recomendar los usuarios sería que bueno que se pudiera redireccionar no a su perfil de usuario sino a una ventana que proporcione la información mejor estructurada
 4. Incluir las redes sociales cuando se redirecciona a una vista de información del usuario
 - ii. Versión 2
 1. Búsqueda de amigos manualmente (ingresando el nombre de la persona)
 2. Recomendación de amigos por medio del colegio en el que se estudió
 3. La ventana de registro no contiene espacios para incluir información como hobbies, colegio o cursos. ¿O eso se agrega luego de registrarse?
 4. Ventana para agregar información para recomendación de amigos
 5. Redireccionamiento hacia las redes sociales de las personas. Con opción que permita redireccionar el número de teléfono a WhatsApp
 6. Opción para actualizar información, como los cursos que se están llevando y el semestre que se está cursando.
- d. Cambios a la lista de requisitos funcionales, si hubiere, de acuerdo a las observaciones hechas a los prototipos.
 - i. Redireccionamiento a redes sociales → agregar redireccionamiento por el número de teléfono directamente a WhatsApp
 - ii. Actualización de información de usuario (como cursos, semestre, carrera)

Análisis

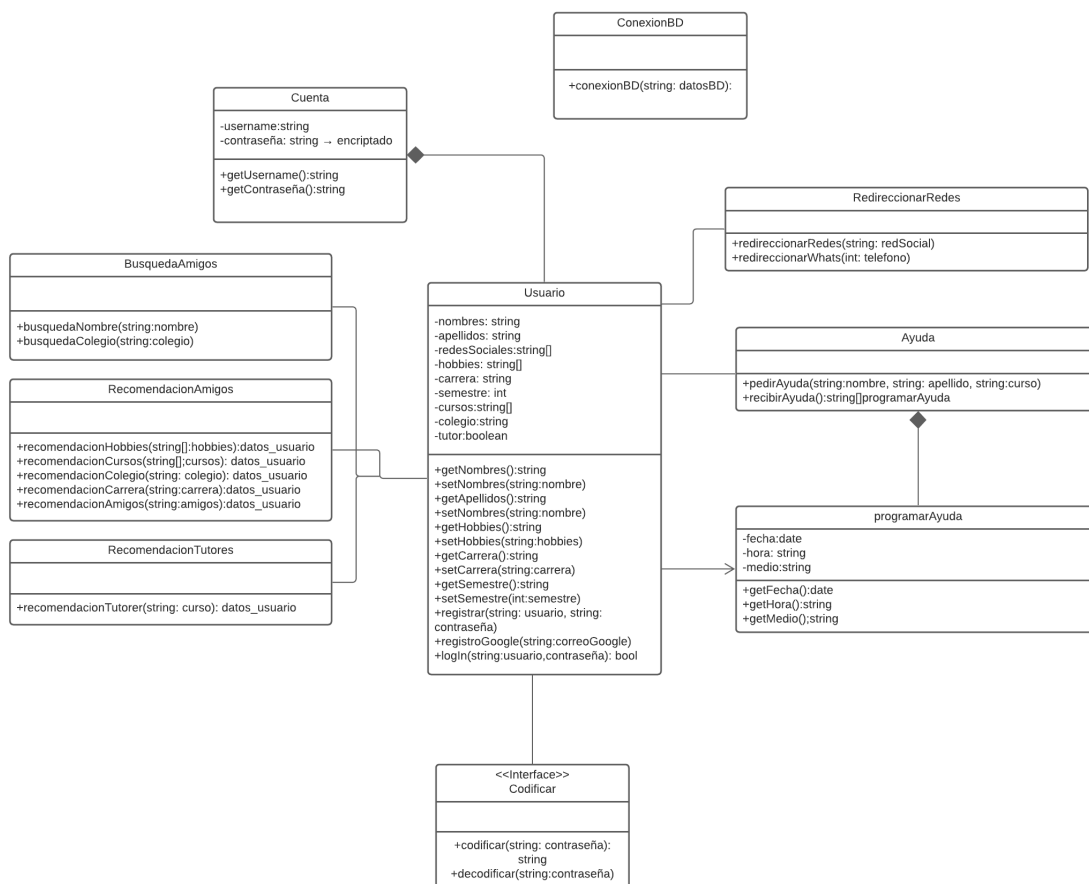
1. Requisitos funcionales

- a. Elabore una lista de requisitos funcionales de su sistema
 - i. Registrarse (facilidad con google o crear cuenta con correo manualmente)
 - ii. Iniciar sesión
 - iii. Mantener sesión abierta
 - iv. Agregar información de usuario (cursos, carrera, hobbies, colegios en los que ha estudiado)
 - v. Buscar candidatos a amigos (por recomendación, por colegio o por búsqueda manual)
 - vi. Redireccionamiento a redes sociales de otras personas
 - vii. Notificación para dar apoyo en cursos del semestre
 - viii. Notificación para recibir apoyo en algún curso del semestre

- b. Historias de usuario
 - i. Como usuario quiero registrarme para poder tener un perfil de usuario que me facilite encontrar personas
 - ii. Como usuario quiero registrarme por medio de google para no tener que ingresar todos mis datos manualmente y aprender una nueva contraseña.
 - iii. Como usuario quiero poder insertar mis hobbies para que la aplicación me recomiende amigos.
 - iv. Como usuario quiero poder insertar mis cursos para que la aplicación me recomiende amigos que lleven los mismos cursos.
 - v. Como usuario quiero poder insertar el último colegio en el que he estudiado para que la aplicación me recomiende amigos que pudieron estudiar conmigo.
 - vi. Como usuario quiero poder ver las redes sociales y los números de teléfono de las personas que me salen en las recomendaciones.
 - vii. Como usuario quiero poder insertar mis hobbies para que la aplicación me recomiende amigos.
 - viii. Como usuario quisiera poder ayudar a personas en algún curso que se le dificulte.
 - ix. Como usuario quisiera pedir ayuda a auxiliares para poder mejorar en algún curso.

2. Backend

- a. Elabore un diagrama de clases, si utiliza un patrón Modelo Vista Controlador deben estar identificadas los tipos de clases.

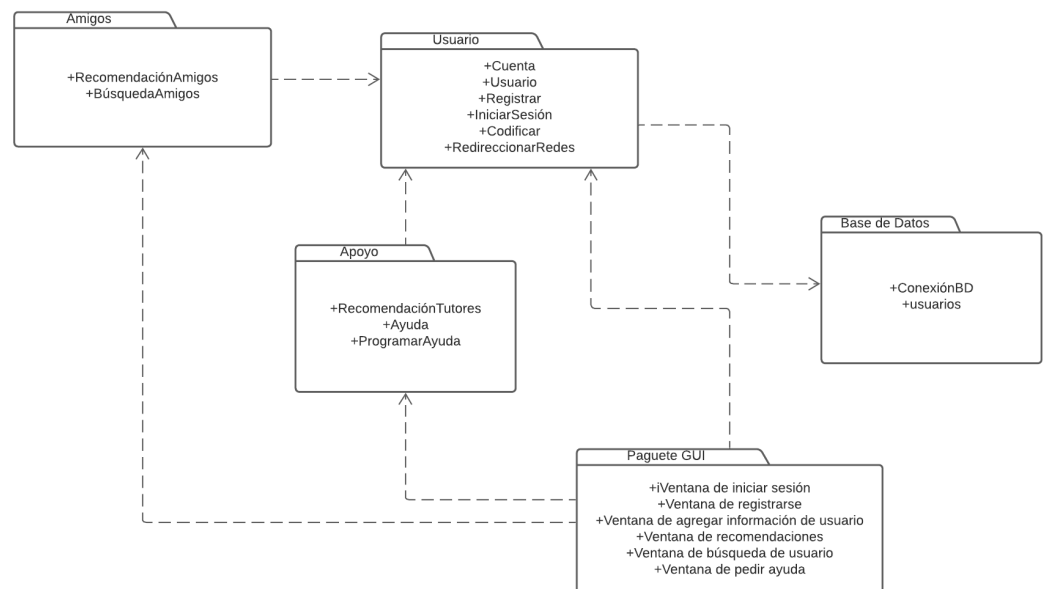


- b. Descripción de las clases. Recuerde que si no puede describirla seguramente no la necesita.
- ConexionBD: se encarga de la conexión a la base de datos para poder extraer la información de la misma.
 - Cuenta: modela los datos necesarios para la creación de una cuenta. Así como los datos de una cuenta ya registrada en el sistema.
 - Usuario: modela los datos de un usuario registrado en el sistema
 - RecomendaciónAmigos: se encarga de la recomendación de amigos, ya sea por hobbies, cursos, carrera, semestre, etc.
 - BúsquedaAmigos: se encarga de la búsqueda manual de amigos ya sea por nombres o por colegio en el que estudió.
 - RecomendaciónTutor: recomienda un tutor dependiendo el curso del cual se necesite tutoría
 - Registrar: guarda un nuevo usuario en la base de datos
 - IniciarSesión: verifica los datos ingresados en el sistema con los datos guardados en la base de datos
 - Ayuda: hace las peticiones de ayuda a otros usuarios (tutores) así como las respuestas a los pedidos de ayuda.

- x. ProgramarAyuda: Modela las respuestas a los pedidos de ayuda
- xi. RedireccionarRedes: se encarga del redireccionamiento a las redes sociales de los usuarios, tanto a IG, Facebook como a whats.
- xii. Codificar(Interface): se encarga del proceso de encriptación de una contraseña al momento de crear una cuenta y de desencriptar dicha contraseña al momento de iniciar sesión para la verificación.
- xiii. Singleton Registrar: Se utilizará esta clase para asegurarnos de que el usuario pueda registrarse con sus datos sin que exista la posibilidad de que haya un error.
- xiv. Singleton IniciarSesión: Se utilizará esta clase para asegurarnos de que el usuario pueda iniciar sesión con sus datos sin que exista la posibilidad de que haya un error.

(Estas dos últimas clases se crean para no arriesgar los datos personales y la contraseña del usuario)

- c. Diagrama de paquetes (paquetes y relaciones entre los mismos. Recuerde mantener alta cohesión y bajo acoplamiento).

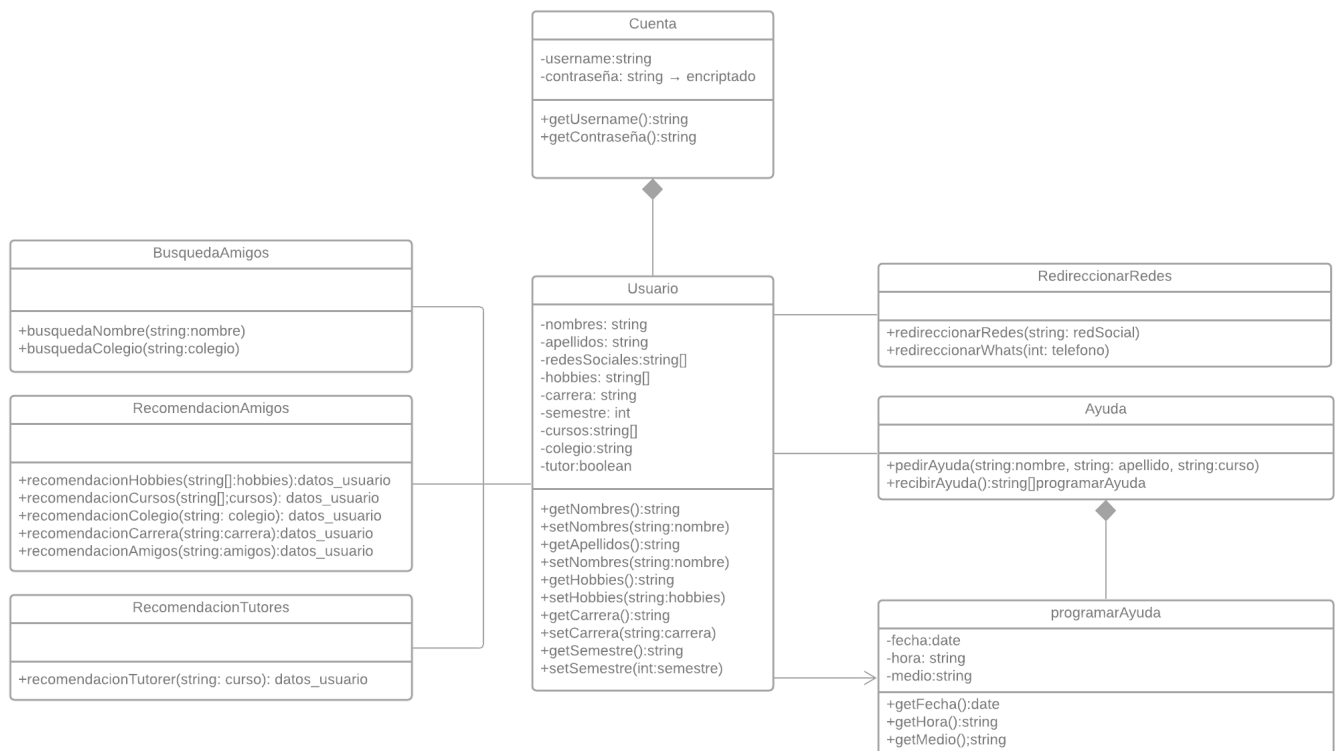


- d. Descripción de cada paquete y de sus componentes.
 - i. Base de datos: agrupa todo lo relacionado a la conexión de con la base de datos, así como un componente usuarios que guarda el cypher de la base de datos.
 - ii. Usuario: guarda los datos relacionados con el usuario, como sus datos de la cuenta en el sistema, la información que proporcionó al crear su cuenta, así como para redireccionar a las redes sociales de los usuarios.
 - iii. Amigos: se encarga de la búsqueda y recomendación de amigos, ya sea por hobby o cursos.
 - iv. Apoyo: agrupa a todos los elementos que se encargan de pedir ayuda a otro usuario, así como la recomendación de tutores por los cursos de los cuales se quiera recibir una tutoría. De igual manera,

- agrupa la programación de una ayuda, en la cual le confirma al auxiliado la fecha y la hora en la cual le puede apoyar en un curso.
- v. Autenticación: se encarga de registrar un nuevo usuario y de verificar que los datos ingresados por un usuario sean válidos para iniciar sesión.
- vi. Paquete GUI: almacena todos los elementos encargados de mostrar la información al usuario (básicamente es toda la vista del sistema)

3. Persistencia de datos

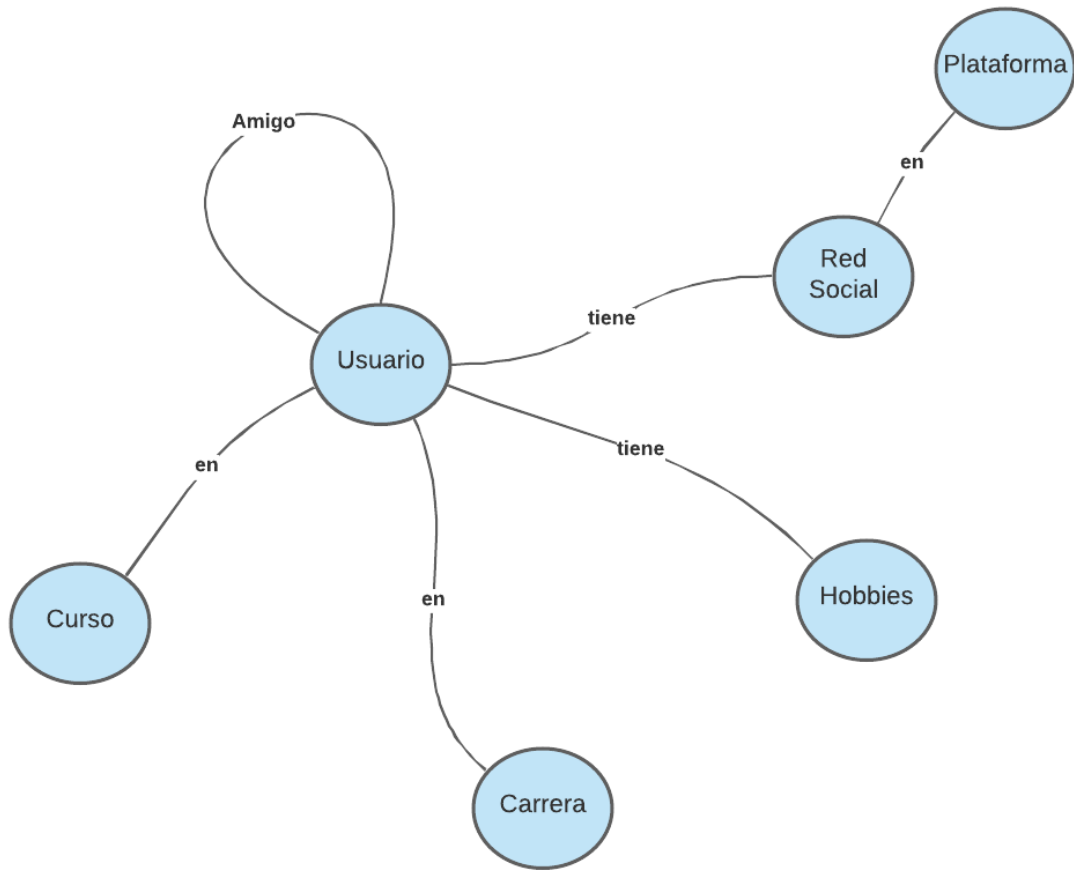
- a. Diagrama de clases persistentes.



- b. Selección del tipo de almacenamiento de datos que hará (Relacional, no relacional, documentos, grafos, etc.) y las razones por las cuales considera que es lo más adecuado. Debe basarse en los requisitos funcionales y no funcionales de su sistema.

Se trabajará con una base de datos de grafos. Esto es debido a que la aplicación trabaja en torno de recomendaciones, de manera que se tendrá que acceder constantemente a otras entidades. Cabe destacar que las bases de datos de grafos suelen ser más eficientes que las bases de datos relacionales al tener que acceder constantemente a otras entidades a través de su conexión.

- c. Diagrama Entidad Relación, en caso de que determine hacer una base de datos relacional o forma y/o elementos que estará almacenando en caso de que no lo sea.



Diseño

1. Selección de la tecnología a utilizar basándose en los requisitos no funcionales:

- *Cantidad usuarios probables pudiera llegar a tener el sistema.*

No conocemos qué cantidad de usuarios utilizaran constantemente la aplicación, pero podemos definir como límite superior a la cantidad de estudiantes del Campus Central activos. De acuerdo con los datos de Fundación de la Universidad del Valle de Guatemala, en el año 2021, 863 personas ingresaron al campus Central de la Universidad del Valle (FUVG, 2021). Asumiendo que la cantidad de ingresantes se mantiene constante, nuestro límite superior de usuarios es de 4315 personas. De nuevo, esto es un límite superior; es improbable que todas los estudiantes utilicen nuestra aplicación y que ninguno de ellos abandone la universidad.

- Tiempo promedio, en segundos que se necesitaría que el sistema demora en dar respuesta.

El tiempo máximo de procesamiento en el servidor de cada respuesta es de tres segundos. En caso de que alguna consulta se tarde más del tiempo alocado, se terminará forzosamente, se devolverá un error y se pasará a la siguiente consulta.

- Cantidad de usuarios promedio trabajando simultáneamente con el sistema.

Debido a que la aplicación está hecha para recomendar grupos sociales, no se espera que la cantidad de usuarios promedio trabajando simultáneamente sea alta. Una vez están armados los grupos sociales, en la mayoría de los casos se dejará de utilizar la aplicación. Esto induce la idea de que la aplicación será utilizada mayoritariamente al inicio del semestre y sobretodo, al inicio del año.

Se puede volver a definir un límite superior de usuarios utilizando la cantidad de personas que ingresan cada año. De tal manera que se espera que el máximo de usuarios trabajando simultáneamente sea de 863 personas.

- Escalabilidad del sistema

Se espera que por cada usuario se debe almacenar el nombre, contraseña, redes sociales y la relación con cursos, amigos y sus hobbies. Todos estos campos utilizan poco espacio de memoria. En la actualidad, no está planeado permitir subir archivos multimedia. Aunque esto significa que no se mostrará imágenes de perfil, consideramos que estas son innecesarias, pues se mostrará las redes sociales en la plataforma.

Asimismo, de acuerdo con Rocha (s.f.), para una base de datos relacional se utilizan 15B por nodo, 34 B por cada relación y un promedio de 41 B por cada atributo. Para calcular la cantidad de espacio que se va a necesitar para esta base de datos, se realizarán las siguientes suposiciones:

- 100 Cursos, cada uno con 3 atributos
- 4315 Personas, cada uno con 4 atributos
- Cada estudiante asiste a 6 cursos.
- Cada persona tiene 100 amigos
- Hay 5 redes sociales
- Cada usuario tiene 3 cuentas de redes sociales.
- Hay 20 carreras universitarias.

$100 \text{ cursos} + 4315 \text{ estudiantes} + 5 \text{ redes sociales} + 4315 * 3 \text{ cuentas de redes sociales} + 20 \text{ carreras} = 17,385 \text{ nodos.}$

$100 * 4315 / 2 \text{ amistades} + 4315 * 6 \text{ asistencias a cursos} + 4315 * 3 * 2 \text{ relaciones redes sociales} + 4315 \text{ conexiones a carreras} = 271,775 \text{ relaciones}$

$$100 * 3 \text{ atributos curso} + 4315 * 4 \text{ atributos usuarios} + 4315 * 3 * 2 \text{ atributos de cuentas redes sociales} + 5 * 2 \text{ atributos redes sociales} + 40 \text{ atributos carrera} = 43500$$

$$17,385 \text{ nodos} * \frac{15 B}{1 \text{ nodo}} + 271,775 \text{ relaciones} * \frac{34 B}{1 \text{ relación}} + 43500 \text{ atributos} * \frac{41 B}{\text{atributo}} = 11284625 B = 11020 MB = 11 GB$$

Asimismo, se le debe agregar un 33% por el indexado de la base de datos:

$$11 GB * 133\% = 15 GB$$

Además, para cada usuario extra se necesitaría:

$$1 \text{ nodo} * \frac{15 B}{1 \text{ nodo}} + (100 + 1 + 3) \text{ relaciones} * \frac{34 B}{1 \text{ relación}} + 6 \text{ atributos} * \frac{41 B}{\text{atributo}} = 18200 B = 18.2 MB$$

$$1.8 MB * 133\% = 2.4 MB$$

En conclusión, se espera que la base de datos utilice 15GB. Asimismo, para cada usuario adicional se necesitará 2.4 MB.

- *Confiabilidad, tolerancia a fallos masivos, replicación de la información.*

Las tecnologías y lenguajes de programación que serán utilizados en el desarrollo de la aplicación serán Python y Django ya que debido al gran flujo de datos que serán manejados estos dos nos brindan eficiencia en la ejecución de la aplicación. Como segunda opción se tiene contemplado utilizar Express.js que nos permite realizar enrutamientos y desarrollo de una aplicación web de una manera mas rapida

Python es un lenguaje de programación altamente efectivo para construir aplicaciones web, analizar datos, automatizar procesos y operaciones y crear aplicaciones con alta escalabilidad y fiabilidad por lo que es bastante útil ya que nuestra aplicación realizará frecuentemente los análisis de datos necesarios para crear relaciones entre usuarios y hacer recomendaciones de posibles amigos.

Django es un framework web de alto nivel que permite desarrollar aplicaciones web de manera rápida, segura y sostenible. Es un framework que nos brinda un alto rendimiento en el flujo de datos extenso y de una manera eficiente por lo que los tiempos de ejecución de los procesos en la aplicación pueden ser cortos gracias a la implementación de este framework.

React

Ventajas	Desventajas
Al momento de representar un componente es bastante sencillo.	Se debe conectar el código para varias solicitudes.
Llega a asegurar la legibilidad y facilita el mantenimiento.	Es una biblioteca bastante amplia.
Se puede usar en cualquier marco porque es una capa vista.	Debido a que es bastante grande, el aprendizaje para manipularlo puede complicarse.

(Javaer, 2020)

- Django:

Ventajas	Desventajas
Cuenta con un sistema de autenticación de usuarios.	Algunos problemas que nos puede dar Django para desarrollar aplicaciones web con Python es que podemos destacar su extensa documentación, que a veces puede llegar a ser confusa y dispersa. También, puede resultar difícil a la hora de realizar API Rest (estilo de arquitectura software para crear aplicaciones web respetando el protocolo HTTP) o resultar complejo cuando se trabaja con sockets
Manejo de versiones que permite una distribución simple de actualizaciones.	
Ofrece un gran rendimiento y flexibilidad, pudiendo escalar proyectos de forma sencilla.	
Trabajar bajo un patrón MVC (Modelo Vista Controlador), lo que permite un desarrollo ágil y reutilizable.	
Incorpora una amplia variedad de paquetes de librerías (más de 4000).	

Dispone de una inmensa comunidad de usuarios en internet.	
Incluye opciones de protección para las aplicaciones, por ejemplo, contra ataques de SQL injection o ataques XSS (cross site scripting).	
Proporciona una estructura de código autogenerado.	
Cuenta con panel de administración para bases de datos.	

(s.f. ilimit)

Python.

Ventajas	Desventajas
<p>Es multiplataforma, ya que funciona en cualquier tipo de sistema el cual tenga un interpretador.</p> <p>Tiene frameworks las cuales nos ayudan tanto para web como móvil.</p> <p>Es de código abierto, donde se puede distribuir los binarios.</p> <p>La sintaxis la cual es simple y que también se reduce la cantidad de caracteres.</p> <p>Nos ofrece varios usos de conceptos tales como polimorfismo, cohesión, herencia, abstracción, entre otros.</p>	<p>Tiene una ejecución lenta a comparación de otros lenguajes de programación.</p> <p>Cuando programamos en web llega a ser un poco más complicado debido a sus funcionalidades.</p> <p>El servicio como tal de Hosting no están tan preparadas.</p>

(BeJob, 2016)

Express Js.

Ventajas	Desventajas
Tiene gran rendimiento,e incluso	Llega a ser asíncrono al momento de

<p>escalabilidad para aplicaciones web.</p> <p>No se pierde tanto el tiempo para las conmutaciones de texto.</p> <p>Se puede acceder a cientos de paquetes reutilizables.</p> <p>Es portable para varias versiones.</p>	<p>realizarlo.</p> <p>Hay ciertas cosas que no necesariamente tienes que hacerlas aquí sino en otro lenguaje.</p>
---	---

(MDN, 2021))

- Neo4j:

Ventajas	Desventajas
Almacena grafos nativos y cuenta con un motor de búsqueda propia que garantiza una velocidad	Nivel previo de capacitación con el lenguaje Cypher. Por lo que puede ser una carga extra de trabajo.
Sus operaciones son de alta fiabilidad ya que garantizan propiedades ACID	El diseño de las bases de datos de grafos es más complejos que los modelos relacionales.
Soporte clúster y conmutación de errores en tiempo de ejecución	

(Orlando, 2021)

- Amazon Neptune:

Ventajas	Desventajas
Lenguajes asociados para consultas: Apache TinkerPop Gremlin y SparQL, lo que facilita la creación de consultas.	No admite consultas analíticas de PageRank
Posee un almacenamiento de datos de hasta 64TB de escala automática en 6 réplicas y 3 zonas de alta disponibilidad.	Sus almacenes de datos de Gremlin y SPARQL no se relacionan entre sí, lo que representa un reto de operatividad
	Las consultas de grafos están limitadas a un máximo de tres saltos lo que limita en cierto grado sus funciones

(Orlando, 2021)

Usaremos Amazon Neptune ya que este nos permite la creación automática de backups, además de que cuenta con un almacenamiento de hasta 64TB, lo que nos permite guardar a una gran cantidad de usuarios. Agregado a esto, Amazon Neptune admite diversos lenguajes de consulta como Gremlin y SPARQL los cuales cuentan con una amplia documentación, por lo que, si bien, actualmente, ningún integrante cuenta con conocimiento de Amazon Neptune, es posible aprenderlo fácilmente. Agregado a esto, python cuenta con librerías fácilmente accesibles para una rápida y fácil conexión a bases de datos creadas en Amazon Neptune.

Nota: Debe incluir una descripción de todas las herramientas consideradas. La explicación de la seleccionada debe ser convincente, y basada en las ventajas y desventajas propias y de las demás.

Informe de gestión

- Desglose el proyecto en tareas que luego serán asignadas a uno o más miembros del grupo.
 - Prototipos
 - Prueba de prototipos
 - Diagrama de clases
 - Diagrama de paquetes
 - Diagrama de persistencia de clases
 - Ejemplo de grafo para guardar datos
 - Diseño
 - Selección de tecnologías → ventajas y desventajas
 - Selección de gestor de datos → ventajas y desventajas
- Elabore una lista con las tareas propuestas y el miembro del equipo que será responsable de cumplirla.

Actividad	Involucrados
Prototipos y Sketches	Orlando Cabrera y Fernando Garavito
Prueba de prototipos	Orlando Cabrera, Diana Corado
Diagrama de clases	Grupo completo
Diagrama de paquetes	Diana Corado
Diagrama de persistencia de clases	Eduardo Ramirez
Ejemplo de grafo para el almacenamiento de datos	Pablo Alejandro Méndez
Diseño	Orlando Cabrera y Pablo Méndez
Selección de tecnologías → ventajas y desventajas de los frameworks y	Eduardo Ramirez

lenguajes escogidos	
Selección de gestor de bases de datos → ventajas y desventajas de diversos gestores de bases de datos basadas en grafos	Fernando Garavito y Diana Corado

-Diana Zaray Corado López-

Fecha	Inicio	Fin	Interrupción	▲ Tiempo	Fase	Comentarios
4-04-21	19:00	20:00	10 min	50 min	Análisis → requisitos funcionales con historias de usuario	-
5-04-21	7:30	15:00	60 min	390 min	Prototipado → cambios en las versiones de los prototipos y modificaciones de los requisitos funcionales Análisis → Diagrama de clases y diagrama de paquetes así como descripción de gestor de bases de datos	Descanso para desayunar y almorzar

-Orlando Osberto Cabrera Mejía-

Fecha	Inicio	Fin	Interrupción	▲ Tiempo	Fase	Comentarios
3-04-21	12:30	18:00	120 min	210 min	Prototipos	Almorcé y comí demasiado, me dolió el estómago :(
4-03-21	8:00	15:00	120 min	270 min	Prototipado → cambios en las versiones de los prototipos y modificaciones de los requisitos funcionales Análisis → Diagrama de clases y diagrama de paquetes así como descripción de gestor	Me puse a hacer otra tarea

-Pablo Alejandro Méndez Morales

Fecha	Inicio	Fin	Interrupción	▲ Tiempo	Fase	Comentarios
4-05-21	7:30	15:00	30 min	420 min	Backend, presentación, diagrama de clases y revisión del documento final.	Desayuné y almorcé, pero comí mientras trabajaba así que no fue tanto tiempo.

-FERNANDO JOSE GARAVITO OVANDO

Fecha	Inicio	Fin	Interrupción	▲ Tiempo	Fase	Comentarios
05-04-21	14:00	16:45	-	165	Ventajas y desventajas Python, React y Express Js	-

-Edwin Eduardo Ramírez Herrera

Fecha	Inicio	Fin	Interrupción	▲ Tiempo	Fase	Comentarios
05-04-21	12:30	15:35	15 min	180 min	Diagrama de Clases persistentes, tecnologías a utilizar, ventajas y desventajas, Django y Python	Almorcé y fui algunas veces al baño.

Bibliografía

Orlando. (2021, 11 febrero). *Comparativa Bases de Datos de Grafos / ¿Cuál es mejor? ¿Cuál elegir?* GraphEverywhere.
<https://www.grapheverywhere.com/comparativa-bases-de-datos-de-grafo-s-cual-es-mejor-cual-elegir/>

Rocha, J. (s.f.). Understanding Neo4j
<https://neo4j.com/developer/kb/understanding-data-on-disk/>

Fundación de la Universidad del Valle [FUVG] (2021). Bienvenidos Nuevos Estudiantes UVG 2021. <https://fuvg.org.gt/mentor30-2021.php>

llimit (s.f.) flask vs. django extraído de:
[https://www.ilimit.com/blog/flask-vs-django/#::-:text=Ventajas%20de%20Django&text=Ofrece%20un%20gran%20rendimiento%20y,librer%C3%ADas%20\(m%C3%A1s%20de%204000\).](https://www.ilimit.com/blog/flask-vs-django/#::-:text=Ventajas%20de%20Django&text=Ofrece%20un%20gran%20rendimiento%20y,librer%C3%ADas%20(m%C3%A1s%20de%204000).)

BeingJob. (2016, 18 septiembre)- 7 RAZONES PARA PROGRAMAR EN PYTHON. Extraído de: <https://www.bejob.com/7-razones-para-programar-en-python/>

Discoder. (2021, 23 enero). Python: Ventajas y Desventajas. Extraído de: <https://www.discoder.tech/python-ventajas-desventajas/>

Javaer. (2020). Ventajas y desventajas de usar ReactJs. Extraído de: <https://www.javaer101.com/es/article/2211628.html>