

Administración de Tecnologías de Información

TI-2402 Algoritmos y Estructuras de Datos

Prof: Andréi Fuentes Leiva.

Tercera Tarea Programada en Java

Daniel Navarro Romero
Carné: 2014044199

Ernesto Brenes Calderón
Carné: 2014044135

Steven Rojas Cortes
Carné: 2014071312

Contenido

Contenido	2
Descripción del problema	3
Diseño del programa	4
Decisiones de diseño	4
Descripción de algoritmos	5
Librerías utilizadas	5
Análisis de Resultados	6
Objetivos Alcanzados	6
Objetivos no alcanzados	6
Manual de usuario	7
1. Página principal	7
Apartado 1	7
Apartado 2	7
2. Perfil	8
3. Búsqueda Persona	8
4. Encontrar Camino	9
5. Error de usuario y contraseña	9
6. Registro Correcto	10
7. Error de búsqueda	11
Conclusión Personal	12

Descripción del problema

Cada tarea programada, independiente de si se haga en el mismo lenguaje o no, representa una dificultad y problemas distintos. Esta tarea programada consiste en crear una red social por medio de una aplicación web java, la cual debe implementarse en la plataforma Heroku. Heroku es un servicio en la nube, el cual permite desplegar aplicaciones web de diversos lenguajes.

Para esta tarea programada se establecen cinco puntos principales, los cuales más que problemas son retos a vencer. Estos cinco puntos se determinaron por la dificultad que representa realizarlos. Además se tomó en cuenta lo necesario que es para la conclusión de la tarea programada. A continuación se describen estos cinco puntos:

1. Grafos: la red social estará implementada por medio de esta estructura de datos, en donde n nodos tienen diversas relaciones entre sí. Este punto es fundamental para el desarrollo de la red social, porque cada nodo representará un perfil de un usuario. Sin embargo no se considera de gran complejidad crear el grafo, ya que se basa en estructura de datos vistas anteriormente y ya sean realizado prácticas y tareas. Cabe recalcar que un aspecto a considerar en este punto es encontrar el camino más corto de un nodo a otro, pero de igual manera ya sea estudiado varios algoritmos al respecto.
2. Aplicaciones Web Java: Una de las características particulares de esta tarea programada es que debe ser una aplicación web java, pero ninguno de los integrantes del grupo ha realizado aplicaciones web antes. Aun cuando sea trabajado con java en las tareas programadas anteriores, crear una aplicación web en este lenguaje implica la investigación de diversas tecnologías como el uso de servlets, JSP, HTML, entre otras. También es importante tomar en cuenta que la aplicación debe adaptarse al tamaño de la pantalla del dispositivo donde se visualice.
3. Uso de Heroku: otro requisito para realizar esta red social es usar la plataforma Heroku, la cual es desconocida del todo para los integrantes del grupo. La complejidad en este punto es cumplir con el formato requerido por Heroku para desplegar la aplicación. Esta situación se deriva el cuarto punto.
4. Uso de maven: Al momento de escribir esta descripción aún no se tenía un claro concepto de lo qué es maven y su funcionamiento, aunque se haya investigado. Lo comprendido hasta el momento es que maven estructura el proyecto para gestionarlo, usando el concepto de dependencias y artefacto. También se encontró que ciertos IDEs como Eclipse crean proyectos basados en maven, pero esto implica considerar el punto cinco.
5. Elegir método de programación: las tareas programadas anteriores fueron realizadas sin la ayuda de IDEs, pero esta tarea programada implica el uso de varias tecnologías que son facilitas por los IDEs. Este punto más que un reto o un problema, es una

decisión de desarrollo. Por esta razón se contará con la ayuda de Netbeans para el desarrollo de esta tarea programada.

Diseño del programa

Decisiones de diseño:

La primera decisión de diseño que se analizó fue si se iba a utilizar un IDE java o un editor de texto. Las variables que se tomaron en cuenta para esta decisión fueron las siguientes:

- Complejidad de la tarea programada
- Tiempo límite y tiempo necesario para concluir la tarea programada
- Beneficios de los diferentes IDEs

La variable más importante que se tomó en cuenta fue la dificultad que resultaba desarrollar la tarea programada en un editor de texto, comparada con la dificultad de desarrollarla en un IDE. Crear la aplicación web implicaba hacer gran uso de elementos como servidores, JSP, maven, HTML, servlets, los cuales resultaban complejos de gestionar manualmente, es decir, mediante código sin la ayuda de un IDE. Por esta razón se decidió utilizar un IDE.

La segunda variable contribuyó en gran parte a decidir usar un IDE, ya que a mayor complejidad se necesitaba más tiempo para terminar la tarea. Después de haber considerado las dos primeras variables seguía analizar los beneficios de los diferentes IDEs. Esto llevó a tomar una segunda decisión que responde a la pregunta ¿Qué IDE elegir? Las dos opciones que se consideraron fueron eclipse y netbeans, pero al final se eligió netbeans por los beneficios que otorgaba.

Otra decisión que se realizó fue realizar la interfaz en Adobe Dreamweaver SC6. Esta decisión se llevó a cabo porque este software permite crear fácilmente archivos HTML, CSS, JSP, JS y la interfaz de la tarea es un archivo con extensión JSP, por lo tanto se utilizó esta herramienta.

Secuenciar el desarrollo de la tarea programada fue una de las decisiones más importantes, ya que existían actividades muy importantes para la realización de esta tarea. Para esto se establecieron prioridades sobre qué actividades debían realizarse primero. Producto de esta decisión se obtuvo la siguiente lista:

1. Desarrollar la estructura de datos, grafos.
2. Crear el algoritmo para encontrar el camino más corto entre dos elementos del grafo.
3. Investigar sobre la creación de aplicaciones web con java.
4. Investigar sobre el uso de Heroku y maven.
5. Implementar el grafo en la aplicación web.
6. Desarrollar las demás funcionalidades de la aplicación.
7. Mejorar el diseño interfaz.
8. Hostear la aplicación en Heroku.
9. Realizar la documentación.

Descripción de algoritmos:

1. Grafo:

El grafo se implementó mediante estructuras de datos conocidas, las cuales fueron listas doblemente enlazadas y nodos. Al grafo se le agregó una lista enlazada principal que contiene a todos los nodos agregados al grafo. Cada nodo representa un perfil y contiene todos los datos de la persona, mediante atributos. Las relaciones que pueda tener una persona con otra, es decir, un nodo con otro, se almacenan en una lista doblemente enlazada que posee cada nodo como atributo. Por ejemplo cuando el nodo A agrega como amigo al nodo B, a nivel de código lo que sucede es que se realiza una copia del nodo B y se agrega en la lista de amigos del nodo A.

2. Hallar el camino más corto:

Para elaborar este algoritmo se basó en el algoritmo de dijkstra, aunque sería más correcto decir que se basó en la búsqueda por anchura, ya que el peso entre todas las aristas de los nodos es de 1. La búsqueda por anchura implementada toma el nodo desde el cual se va a partir y se almacena en una cola. Luego se almacena en la cola todos los nodos de la lista de amigos del nodo de partida y se establece el nodo de partida como visitado. Posteriormente se van agregando a la cola los amigos de los amigos del nodo de partida y así sucesivamente, hasta que se establecen las rutas más cortas del nodo de partida hacia los demás nodos. Por último se obtiene la ruta que se necesita mostrar.

Librerías utilizadas

Las librerías utilizadas fueron muy pocas y las que se usaron son propias de java. La librería más relevante es la javax.servlet. Un Servlet es una clase en lenguaje Java usada para ampliar la funcionalidad de los servidores web a los que se accede vía modelo de programación request-response. Entre las funcionalidades de los Servlets se encuentran las siguientes:

1. Leer los datos enviados por el cliente.
2. Extraer cualquier información útil incluida en la cabecera HTTP o en el cuerpo del mensaje de petición enviado por el cliente.
3. Generar dinámicamente resultados.
4. Formatear los resultados en un documento HTML.
5. Establecer los parámetros HTTP adecuados incluidos en la cabecera de la respuesta.
6. Enviar el documento final al cliente.

Análisis de Resultados

Objetivos Alcanzados

1. Se logró realizar la tarea programada en aplicación web.
2. Se implementó efectivamente el uso de grafos, mediante código propio.
3. Se pudo comprender como funciona la búsqueda por anchura y aplicarla a la tarea programada.
4. Se obtuvo una aplicación web responsive, es decir, se adapta a la pantalla del dispositivo.
5. Liberación de cajeros, al seleccionar el cajero que se desea asignar como disponible, el sistema de forma automática establecerá el cajero al cliente que se encuentre en la cola, según su prioridad como tipo de cliente.

Objetivos no alcanzados

1. No se pudo hostear la aplicación en Heroku, ya que no se logró cumplir con el formato requerido por Heroku para desplegar la aplicación. Tampoco se pudo instanciar el servidor Heroku.

Manual de usuario

1. Página principal:

Al ejecutar la aplicación web se puede visualizar la página principal. Esta página principal permite crear un perfil o bien iniciar sesión con un perfil ya creado. A continuación se muestra la esta página:

Red Social GeekTec

1 Ingresa a su perfil

Correo: Formato: algo@dir.com Contraseña: Iniciar Sesión

2 Regístrate

Nombre:

Correo: Formato: algo@dir.com

Contraseña:

Edad:

Telefono: Formato: 88765434

Año de Carrera: dd/mm/aaaa

Seleccione su Carrera: Administración de Empresas

Aceptar Cancelar

Apartado 1:

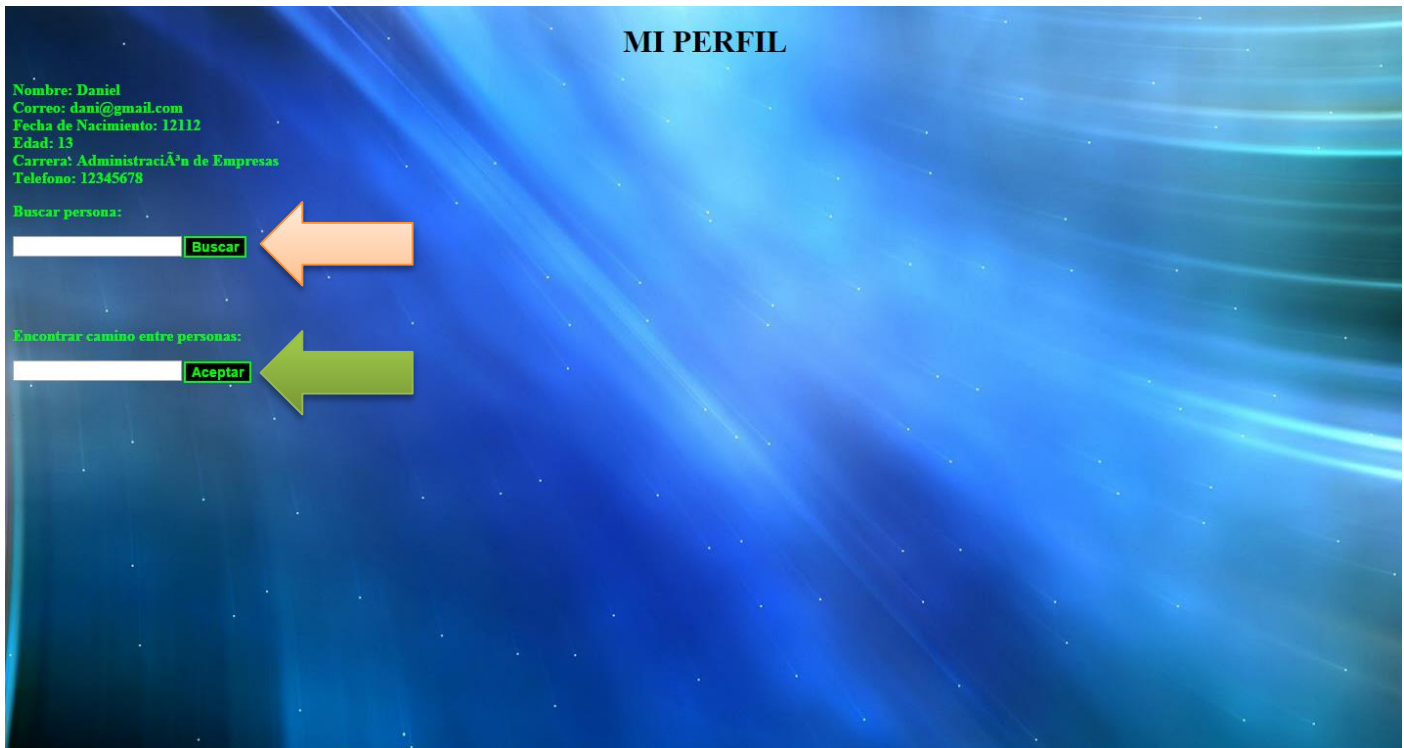
En este apartado la aplicación web permite iniciar sesión con un perfil creado anteriormente. Para ingresar solo es necesario digitar el correo electrónico y la contraseña correctamente. Luego se pulsa el botón Iniciar Sesión y podrá ingresar a su perfil.

Apartado 2:

En este apartado se solicitan todos los datos necesarios para crear un nuevo perfil. Todos los campos de textos deben estar llenados correctamente. Una vez que se han ingresado todos los datos se pulsa el botón aceptar y el perfil quedará creado.

2. Perfil

Cuando el usuario inicie sesión, automáticamente se estará ejecutando la siguiente página; la cual consiste en ser la página de perfil del usuario que inicio sesión, esta visualizara cada uno de los datos del usuario:



Como se logra apreciar, en esta página estará mostrando la información del usuario que ha iniciado sesión, el nombre del usuario, su correo electrónico, fecha de nacimiento, edad, carrera, y número telefónico, además de ofrecerle a este las siguientes opciones:

- ➡ Esta opción, va a permitir al usuario buscar una persona que desea agregar como amigo, únicamente se indica el nombre de esa persona y posteriormente aparecerá una página con la información de esa persona, si esa persona ya es amigo del usuario, aparecerá toda la información de esa persona en caso contrario únicamente se mostrará el nombre y correo.
- ➡ Aquí se estaría implementando la funcionalidad de encontrar el camino más corto a la persona que se indique en el campo de texto disponible, según sea la relaciones que el usuario utilizara para llegar a esa persona se estaría determinando el camino más corto.

3. Búsqueda de Persona

Esta página constituye en mostrar la información de la persona que el usuario busco, como se menciona anteriormente si este es amigo del usuario manifestara toda la información de este, y si no solamente retornara el nombre y su correo electrónico respectivamente.



4. Encontrar camino más corto

En este apartado consiste en identificar el camino más corto de una persona a otra, en este caso el usuario con el individuo que el mismo indique, se identificará en la página, por los amigos de los amigos, y así sucesivamente hasta llegar a la persona que se esté buscando.

5. Error de usuario y contraseña

En el momento que el usuario desee iniciar sesión, y de alguna manera digito de manera incorrecta su usuario o su contraseña o simplemente este no este registrado, automáticamente la siguiente página será mostrada:



Lo sentimos el nombre de usuario o contraseña son incorrectos!!!

6. Registro Correcto

Si el usuario ha decidido registrarse, y ha digitado sus datos de una forma correcta, inmediatamente se le estaría notificando que su registro se ha concretado de manera correcta de la siguiente manera:



7. Error de búsqueda

En este apartado se estará notificando al usuario que la persona que intento buscar no se realizó de manera correcta, debido a que digito de mal manera sus datos o esta persona no está registrada:



No se encontraron resultados para la búsqueda

Conclusión Personal

Esta tarea programada represento un mayor nivel de complejidad respecto a las tareas programadas anteriores, debido al uso de nuevas tecnologías y a la implementación en aplicación web. El punto de mayor complejidad fue el desarrollo de la aplicación web en java y cumplir con el formato necesario para desplegar la aplicación en Heroku. Esto último implicaba usar maven, ya que es requisito de Heroku para poder desplegar la aplicación.

Esta situación nos llevó a solicitar más tiempo para el desarrollo de la tarea programada, porque aún no se superaba la curva de aprendizaje, tanto en cómo crear aplicaciones web en java, cómo funcionaba maven y cómo utilizar Heroku. Aún después de superar esta curva de aprendizaje el desarrollo de la aplicación no estaba exenta de problemas e incidentes inesperados. Los algoritmos como el grafo, los recorridos, encontrar el camino más corto de un nodo a otro, todos estos no presentaron mayor complejidad al realizarlos. Lo difícil fue comprender como utilizarlos en la aplicación web.

Un punto que se discutió y se analizó fue el enfoque de la tarea programada, es decir el objetivo de la tarea programada. En las especificaciones de la tarea programada se menciona en la introducción lo siguiente “el objetivo de la tarea es implementar un sistema de red social de estudiantes, usando grafos para representar la información”, sin embargo la tarea programada giró alrededor de aplicaciones web, Heroku y maven.

Se sabe que como estudiantes de Administración de Tecnología de Información debemos estar abiertos al aprendizaje de tecnologías y herramientas nuevas, pero se opina que se debe mantener un balance entre las herramientas nuevas implementadas en la tarea programada y el objetivo principal de esta. Esto porque con la interacción con los compañeros se pudo determinar que la mayoría se estaba centrando más en implementar la aplicación web y el uso de Heroku, restando importancia a la creación del grafo y la gestión de este. Por ejemplo con GitHub y Ant se mantuvo una continuidad en las dos tareas programadas, por lo que se pudo superar la curva de aprendizaje y realizar un buen uso de estas dos herramientas.

Esta perspectiva acerca del enfoque de la tarea programada no fue la razón para no usar Heroku, se investigó, se realizaron prácticas y ejemplos, pero al final no se pudo desplegar la aplicación en Heroku. En síntesis la creación de una red social está muy acorde al uso de la estructura de datos llamadas grafos. Sin embargo la complejidad de utilizar nuevas herramientas le quitó importancia al uso y manejo de las estructura de datos, las cuales son el tema principal de la tarea programada e incluso del curso.