

Institución: Instituto Tecnológico de Costa Rica

Área: Ingeniería en Computadores

Curso: Algoritmos y Estructuras de Datos I

Profesor: Antonio Gonzales Torres

Grupo: 02

Documento Anexo

Integrantes:

- **Ángelo Ceciliano Ortega | 2021035484.**
- **Byron Mata Fuentes | 2021430403.**
- **Gustavo Alvarado Aburto | 2021055170.**
- **Sebastián Chaves Ruiz | 2021032506.**

Año: 2021

Documento Anexo

Tabla de contenido

| | | |
|------|--------------------------------------|----|
| I. | Introducción..... | 1 |
| II. | Diseño..... | 2 |
| a) | Listado de requerimientos..... | 2 |
| | Requerimientos..... | 2 |
| b) | Estado del arte..... | 3 |
| □ | Algoritmo de Dijkstra..... | 3 |
| □ | Algoritmos de ordenamiento..... | 3 |
| □ | API..... | 3 |
| □ | Grafos..... | 3 |
| □ | Grafo dirigido..... | 4 |
| □ | Matrices de adyacencia..... | 4 |
| III. | Trabajo individual y en equipo..... | 4 |
| a) | Metas del proyecto..... | 4 |
| b) | Roles..... | 5 |
| c) | Reglas..... | 5 |
| d) | Cronograma..... | 6 |
| e) | Bitácora..... | 7 |
| | [15/11/2021 – 19:30 – 21:00]..... | 7 |
| | [16/11/2021 – Hora no definida]..... | 7 |
| | [17/11/2021 – 14:00 – 19:20]..... | 8 |
| | [21/11/2021 – 17:00 – 22:00]..... | 8 |
| | [23/11/2021 – 17:00 – 22:00]..... | 9 |
| | [24/11/2021 – 14:00 – 21:00]..... | 9 |
| | [25/11/2021 – 11:00 – 14:30]..... | 10 |
| IV. | Referencias..... | 10 |

I. Introducción

Este es un documento escrito de aspectos administrativos para el tercer proyecto del curso “algoritmos y estructuras de datos I”. Como se asignó por nombre este **documento anexo** tiene como **objetivo principal** el de elaborar un documento que evidencie la capacidad para la descripción de un problema complejo de ingeniería en términos de requerimientos de diseño y limitantes y la capacidad para la organización de las actividades individuales en el equipo de trabajo; así mismo como cuerpo y como **objetivos específicos** este propone determinar los requerimientos de ingeniería para un problema complejo considerando partes involucradas, estado del arte, estándares, normas, entre otras, así como también establecer los lineamientos para el trabajo en equipo (metas, roles, reglas, cronogramas, bitácoras, entre otros) y finalmente identificar el rol y las responsabilidades como miembro de un equipo de trabajo. Adicional a lo anterior se presenta el apartado de referencias en donde se encontrarán las fuentes bibliográficas utilizadas para el estado del arte, por lo que, dentro del cuerpo del documento se realizará un seccionamiento de los apartados en *Introducción, Diseño, Trabajo individual y en equipo y Referencias*.

II. Diseño

a) Listado de requerimientos

Requerimientos

1. Seleccionar un país o ciudad a modelar mediante un grafo dirigido de al menos 15 nodos.
2. El grafo debe de ser creado tomando en cuenta la distancia en km existente entre los nodos de este, para se debe hacer uso de las API's de Google Maps.
3. Los nodos del grafo dirigido deben de ser puntos de interés, como parques, monumentos, museos, restaurantes y gasolineras del país o ciudad modelada.
4. El cálculo de la mejor ruta se realizará en términos de tiempo que se tome en recorrer la ruta indicada, por lo que, se va a suponer que se viaja a una velocidad constantes de 80km/h.
5. Calcular la extracción del tiempo del recorrido tomando en consideración que *"velocidad = distancia / tiempo"*, por lo que, es posible extraer el tiempo que tomará en recorrerla la ruta teniendo en cuenta que *"tiempo = distancia / velocidad"*.
6. Inclusión de factores como el tráfico, los semáforos o accidentes de tránsito, para con ello modelar una situación de la vida cotidiana en donde se generen atrasos en términos de tiempo de viaje.
7. Uso del algoritmo de Dijkstra para realizar el cálculo de la mejor ruta entre los nodos del grafo dirigido.
8. Creación de una ventana de ayuda al viajero en donde se le permita al usuario analizar los lugares que componen el mapa, en estos se debe de mostrar cantidad de habitantes, lugares de interés, lugares de comidas y gasolineras.
9. Los lugares que componen al mapa se deben de listar y ordenar alfabéticamente utilizando el algoritmo de ordenamiento que se desee.

b) Estado del arte

- [Algoritmo de Dijkstra](#)

Según indica Bryan Salazar López (2019), el algoritmo de Dijkstra también llamado como algoritmo de caminos mínimos, este es un modelo que se ubica dentro de los algoritmos de búsqueda, en donde se define como su objetivo es determinar la ruta más corta entre dos nodos siendo uno el origen y otro cualquiera de una red de nodos, además de ello este se basa en iteraciones y su dificultad aumenta conforme el tamaño de la red de nodos o bien la cantidad de nodos aumenta.

- [Algoritmos de ordenamiento](#)

Los algoritmos de ordenamiento son aquellos que permiten ordenar información de una manera especial basándonos en un criterio de ordenamiento. Fernando Lagos (2007) acerca de este concepto indica que estos algoritmos permiten ordenar ya sean vectores, matrices o lista con valores asignados aleatoriamente.

- [API](#)

De acuerdo con el sitio web MuleSoft (2015), API es un acrónimo de Application Programming Interface, que es un intermediario de software que permite que dos aplicaciones se comuniquen entre sí, definición que coincide con la de Yúbal Fernandez el cual para el sitio web Xataka en donde además de ello y de realizar una traducción del acrónimo siendo este interfaz de programación de aplicaciones, realiza la adición que este concepto es “una especificación formal que establece cómo un módulo de un software se comunica o interactúa con otro para cumplir una o muchas funciones” (2019), sin embargo, por otro lado, en Red Hat (2017) indica que una API es un conjunto de definiciones y protocolos que se utiliza para desarrollar e integrar el software de las aplicaciones.

- [Grafos](#)

Un grafo de acuerdo con GRAPHEVERYWHERE “un grafo es una composición de un conjunto de objetos conocidos como nodos que se relacionan con otros nodos a través de un conjunto de conexiones conocidas como aristas” (2019), de forma más técnica se conoce como grafo a dos conjuntos finitos; además de lo indicado anteriormente los grafos aparecen como una

extensión del concepto de árbol como estructura de datos esto ya que ambos comparten de que cada elemento puede tener varios elementos y más de un sucesor.

- Grafo dirigido

“Un grafo dirigido conocido también como dígrafo consta de un conjunto de vértices y aristas donde cada arista se asocia de forma unidireccional a través de una flecha con otro” (Graph EveryWhere, 2019), es la anterior definición la concepción que se tiene del concepto de grafo dirigido que al final se clasifica como un tipo de grafo y que en otras palabras es aquel grafo en donde sus aristas tienen un único sentido o dirección.

- Matrices de adyacencia

Una matriz de adyacencia es la manera más sencilla y fácil de implementar un grafo, en sí es una matriz bidimensional por lo que en resumidas palabras estas son una manera de representar un grafo. Entrando en mayor detalle las filas y columnas representa un vértice en el grafo y el valor que se almacena en la celda es la intersección de la fila y la columna indicando si hay una arista desde un vértice a otro.

III. Trabajo individual y en equipo

a) Metas del proyecto

1. Realizar la entrega del proyecto para la fecha asignada.
2. Completar el proyecto en su 100%
3. Realizar la entrega del documento anexo para la fecha asignada.
4. Completar el documento anexo en su 100%
5. Completar la documentación interna y externa en su 100%
6. Seguir las reglas establecidas, así como el cronograma durante el proceso de desarrollo.

b) Roles

| Roles | Nombre |
|-------------------|--|
| Líder(es) | <ul style="list-style-type: none">• Byron Mata Fuentes• Gustavo Alvarado Aburto |
| Desarrollador(es) | <ul style="list-style-type: none">• Byron Mata Fuentes• Gustavo Alvarado Aburto• Sebastián Chaves Ruiz• Fabian Ceciliano Ortega |
| Investigador(es) | <ul style="list-style-type: none">• Gustavo Alvarado Aburto• Byron Mata Fuentes• Sebastián Chaves Ruiz• Fabian Ceciliano Ortega |

c) Reglas

1. Respeto bajo cualquier concepto a los integrantes del grupo de trabajo.
2. Promover una comunicación asertiva entre los miembros del grupo de trabajo.
3. Compromiso hacia, para y con el proyecto e integrantes del grupo de trabajo.
4. Puntualidad con las horas asignadas, para las tareas y reuniones.
5. Realizar aportes en la solución de problemas, desafíos, entre otros.
6. Mostrar iniciativa en el desarrollo del proyecto, por medio de la participación.
7. Realizar una retroalimentación a los aportes realizados por los miembros, en caso de ser necesario.
8. Mantener informados a los miembros del grupo de trabajo de cualquier actualización realizada.
9. Luego de cualquier actualización, indicar lo realizado en la bitácora, de igual manera con alguna conclusión o recomendación.
10. Trabajar de manera clara y ordenada.

d) Cronograma

| Actividad | Asignada a... | Día de asignación | Día de entrega |
|--|--|--------------------------|-----------------------|
| Documento anexo | Byron Mata F. | 15 de noviembre | 16 de noviembre |
| Selección de ciudad a modelar | Byron Mata F. Ángelo Ceciliano O. Gustavo Alvarado A. Sebastián Chaves R. | 17 de noviembre | 17 de noviembre |
| Borrador o machete del grafo dirigido | Ángelo Ceciliano O. | 17 de noviembre | 17 de noviembre |
| Interfaz visual de la aplicación (grafo dirigido y botones) | Byron Mata F. Ángelo Ceciliano O. Sebastián Chaves R. | 17 de noviembre | 23 de noviembre |
| Algoritmo Dijkstra | Gustavo Alvarado A. | 17 de noviembre | 24 de noviembre |
| Conexiones de métodos | Byron Mata F. Gustavo Alvarado A. Sebastián Chaves R. | 23 de noviembre | 24 de noviembre |
| Algoritmos solicitados (sort y patron) | Byron Mata F. Ángelo Ceciliano O. Gustavo Alvarado A. | 24 de noviembre | 24 de noviembre |

| | | | |
|---|---|-----------------|-----------------|
| Interfaz de información | Byron Mata F. Ángelo Ceciliano O. Gustavo Alvarado A. | 24 de noviembre | 24 de noviembre |
| Sistema de búsqueda (búsqueda de las locaciones) | Byron Mata F. Gustavo Alvarado A. | 24 de noviembre | 25 de noviembre |
| Diagrama de clases (Wiki de GitHub) | Ángelo Ceciliano O. Sebastián Chaves R. | 25 de noviembre | 25 de noviembre |
| Entrega del proyecto | --- | --- | 25 de noviembre |

e) Bitácora

[\[15/11/2021 – 19:30 – 21:00\]](#)

En este día se realizó una reunión donde se llevó a cabo la discusión para la definición de algunos aspectos relacionados a detalles y requerimientos del proyecto como, por ejemplo, la definición de la ciudad y los puntos que se tomaran de esta, ante esta discusión parecieron algunas dudas referentes a estos aspectos, por lo que se dejaron pendientes para realizar la consulta de estas. Por otro lado, se revisó el enunciado a seguir para el desarrollo de este proyecto en donde se establecieron algunas pautas y otros asuntos referentes al documento anexo lo que serviría para el proceso de desarrollo.

[\[16/11/2021 – Hora no definida\]](#)

Byron: Se encargo de realizar la primera tarea asignada la cual consistía en elaborar lo más completamente posible el documento anexo, lo cual fue de esta manera ya que los únicos apartados faltantes por completar fueron la bitácora y el cronograma los cuales se irán completando en el proceso de desarrollo.

[\[17/11/2021 – 14:00 – 19:20\]](#)

Ángelo: Con ayuda de Sebastián se recopiló la información de las locaciones de los Ángeles, California, la información como las distancias entre las locaciones, los nombres de las locaciones e información básica de estas. A partir de esto se crea el boceto del grafo inspirado en las locaciones más emblemáticas de L.A.

Byron: En un primer instante con ayuda de Sebastián se empezó la elaboración manual de la interfaz principal de la aplicación esto creando la clase de PathInterface la cual se ubicaría en el package llamado Interfaces. Seguido de ello, pero ahora de manera individual, se inició la representación del grafo creado por Ángelo dentro de la misma interfaz creada, finalmente a la clase creada le realizó la documentación interna pertinente.

Gustavo: Se encargó de realizar diferentes investigaciones sobre la implementación del algoritmo de Dijkstra, además empezó a desarrollar el algoritmo de Dijkstra a base de una matriz que contiene todas las distancias entre cada uno de los vértices, dicha implementación no se finalizó este día.

Sebastián: Realizo el borrador de la interfaz en la cual se va a usar para el proyecto, misma la cual resultó editada con ayuda de Byron para finalmente definirla como la interfaz que llevará la aplicación.

De forma general al final del día se debatió algunos aspectos relacionados a la interfaz en sí de la aplicación en su completitud.

[\[21/11/2021 – 17:00 – 22:00\]](#)

Byron: Durante este día trabajó en graficar en la interfaz de la pantalla principal las aristas que conforman el grafo, así como adicionar los “indicadores” de cada uno de los nodos que forman este mismo y se les dio la función de mostrar en pantalla la aparición de información del sitio al que seleccionaron.

[\[23/11/2021 – 17:00 – 22:00\]](#)

Ángelo: Con ayuda de Byron se crea los Labels contenedores de la información general de los Ángeles, California.

Byron: Se trabajó nuevamente en la documentación interna del código, además de la adición de algunos Label los cuales conformaban la interfaz principal de la aplicación, así como también se dio apoyo en la función que los botones desarrollarían creando una subclase para ello y un su respectivo método.

Gustavo: Se terminó la implementación del algoritmo de Dijkstra y a su vez se creó la matriz que contiene todas las distancias entre cada uno de los vértices. Además, aportó en la implementación de las funciones que tiene cada botón implementando la detección de presionar 2 botones en la interfaz e identificar que botón se presionó.

Sebastián: Realizó la clase "Data", en la cual se genera un JOptionpane que muestra la información del lugar en específico al presionar el botón, así mismo colaboró con otros aspectos de los cuales se trabajaron el mismo día.

[\[24/11/2021 – 14:00 – 21:00\]](#)

Ángelo: Se trabaja en la interfaz del PathFinder, en la parte de la adicción de nombres de los botones.

Byron: Se encargo crear la interfaz que sería usada para mostrar la información de las locaciones dadas, así mismo los botones y los label que esta debería de contener, además de ello, junto con Gustavo se encargó de darle la función respectiva a los diferentes botones que en esta pantalla o ventana se encontraban.

Gustavo: Se encargó de realizar la implementación del algoritmo de ordenamiento Quicksort para ordenar los nombres de lugares alfabéticamente, así como colaborar con Byron en la asignación de la función respectiva a cada botón de la lista y realizar algunos cambios y depuración del algoritmo de Dijkstra.

Sebastián: Se encargo de realizar algunas correcciones a la clase “Data” en donde seguidamente empezó a realizar los encabezados de los diferentes métodos implementados en las demás clases y reviso atentamente los ya existentes, es decir, continuó con la documentación interna.

[25/11/2021 – 11:00 – 14:30]

Byron & Gustavo: Se encargaron de realizar el sistema de búsqueda, en donde y debido a ello se tuvo que realizar un cambio en la clase de “inf_Interface” ya que al agregar el JTable se necesito cambiar la clase en su completitud, además de ello ambos realizaron algunos métodos para mejorar algunos detalles de esta interfaz, finalmente se realizó el JavaDoc, así como también la revisión de la documentación y la adición de la restante, para concluir con el último commit.

Ángelo & Sebastián: Se trabajó asincrónicamente a los compañeros Gustavo y Byron en la creación del diagrama de clases, donde seguido de ello se realizó la adición o publicación de este en el apartado de Wiki del repositorio de GitHub.

IV. Referencias

Graph EveryWhere. (8 de julio, 2019). Qué son los grafos. Graph EveryWhere.
<https://www.grapheverywhere.com/que-son-los-grafos/>

Fernández, Y. (23 de agosto, 2019). API: qué es y para qué sirve. Xataka.
<https://www.xataka.com/basics/api-que-sirve>

Lagos, F. (2007). Algoritmos de Ordenamiento[PDF].
http://tigger.itc.mx/conacad/cargas/VITD591124JC2/CF13/Informe_Ordenamiento.pdf

MuleSoft. (26 de noviembre, 2015). What is an API (Application Programming Interface)?
MuleSoft. <https://www.mulesoft.com/resources/api/what-is-an-api>

Salazar, B. (12 de junio, 2019). Algoritmo de Dijkstra. Ingeniería Industrial Online.
<https://www.ingenieriaindustrialonline.com/investigacion-de-operaciones/algoritmo-de-dijkstra/>

Red Hat. (31 de octubre, 2017). ¿QUÉ ES UNA API? Qué son las API y para qué sirven. Red Hat.
<https://www.redhat.com/es/topics/api/what-are-application-programming-interfaces>

Tema 6: Grafos. (sf). Tema 6: Grafos[PDF].

https://www6.uniovi.es/usr/cesar/Uned/EDA/Apuntes/TAD_apUM_07.pdf