

Contexto problemático:

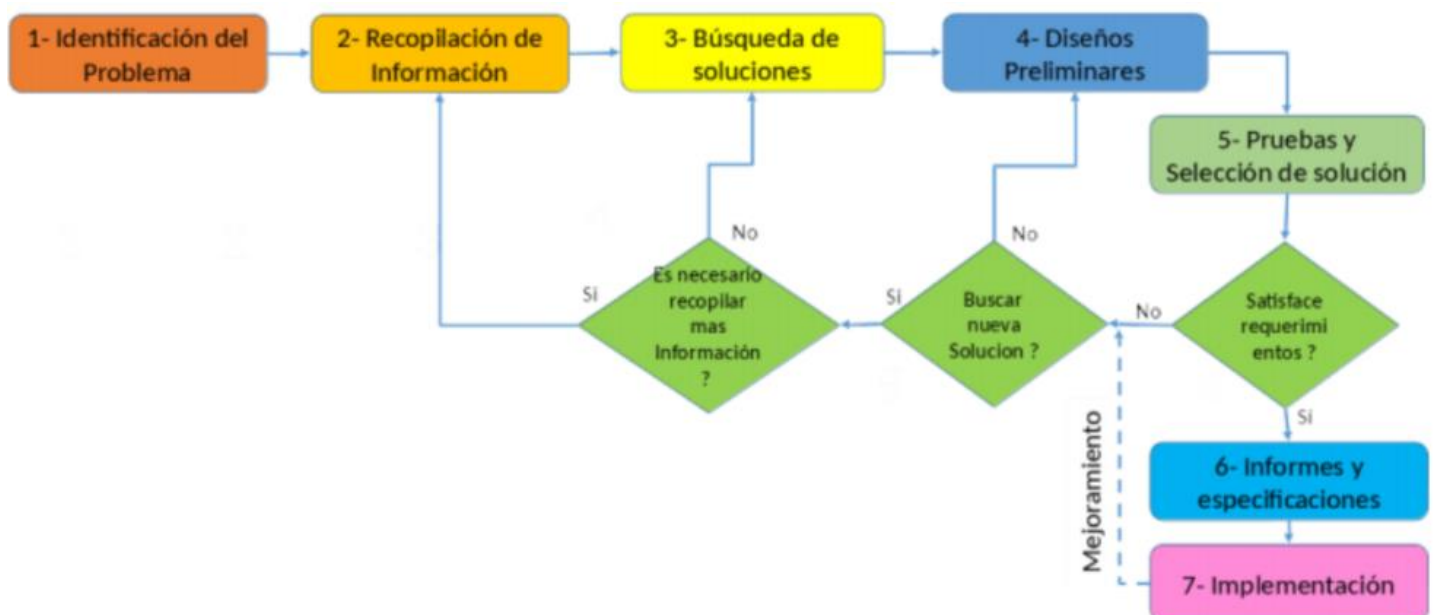
Rappi, una compañía multinacional que ha estado en auge desde 2015 como una de las mejores compañías de comercio y distribución electrónica, se ha enfrentado a una serie de problemas en este último año que han generado inconformidades en sus clientes, estas, con respecto al servicio que la empresa está prestando; una de las causas de la inconformidad de los clientes radica en el tiempo de entrega de cada uno de los productos.

Es por esto que Rappi ha tomado la decisión de contratar a un grupo de jóvenes desarrolladores de software para llevar a cabo la creación de una nueva aplicación que le permita a los encargados del domicilio cumplir con las entregas de manera eficaz; y así ir mejorando su servicio y mermar las inconformidades de los clientes.

Desarrollo de la solución:

Para resolver el caso problemático anterior se eligió el método de la ingeniería del libro “Introduction to Engineering” de Paul Wright.

Los pasos para conseguir el desarrollo de la solución se describen en el siguiente diagrama de flujo:



Paso 1. Identificación del problema:

El problema que nos presenta el cliente lo hemos acogido y hemos encontrado los siguientes requerimientos, que, al ser solucionados, se entregará la aplicación solicitada por el cliente.

REQUERIMIENTO FUNCIONAL # 1	
Nombre	Mostrar la ruta con la menor distancia
Resumen	Se le mostrará al usuario la ruta que tiene la menor distancia para desplazarse de un punto a otro.
Entrada	
Destino de salida del pedido y destino de llegada del usuario.	
Resultado	
Se le muestra al usuario la ruta que recorre la menor distancia para llegar a su destino.	

REQUERIMIENTO FUNCIONAL # 2	
Nombre	Mostrar el recorrido más corto de un sector.
Resumen	Se le mostrará al usuario la ruta que debe seguir para realizar el recorrido de un sector con el menor costo posible.
Entrada	
Delimitaciones del sector.	
Resultado	
Se le mostrará al usuario la ruta que tenga el menor costo para recorrer un sector determinado.	

REQUERIMIENTO FUNCIONAL # 3	
Nombre	Mostrar rutas disponibles.
Resumen	Se le mostrará al usuario las rutas disponibles para llegar a su destino.
Entrada	
Destino de salida del pedido y destino de llegada del usuario.	
Resultado	
Se le mostrará al usuario posible rutas que puede usar para llegar a un destino en específico.	

Paso 2. Recopilación de información:

Para poder dar solución al problema, primero debemos entrar en materia con el tema pedido y algunos conceptos necesarios para dar solución a él.

Recopilamos la siguiente información con el fin de aclarar todas las dudas propuestas por nuestro grupo de trabajo respecto al caso problemático:

Rappi: es una compañía multinacional colombiana de comercio electrónico, con sede principal en Bogotá, Colombia. Activa en México, Brasil, Uruguay, Argentina, Chile, Perú y Colombia, país donde fue fundada en 2015 por Felipe Villamarín, Sebastián Mejía y Simón Borrero (CEO de la compañía). Más de la mitad de sus ventas toman lugar en la Zona Metropolitana de la Ciudad de México.

Uno de las características que define Rappi es la gama ancha de los productos y los servicios disponibles para la entrega — el co-fundador Sebastián Mejía dice "queremos ser una 'tienda de todo'". La aplicación móvil permite a los consumidores pedir el mandado del supermercado, comida, y medicamentos de farmacias, pero también permite enviar dinero en efectivo a alguien, o que un corredor retire dinero de su cuenta bancaria de un cajero automático y entregárselo. Los corredores hasta pueden pasear los perros del cliente.

Su interfaz de usuario luce como estantes de supermercado y los usuarios pueden deslizar productos a su cesta. Los clientes pueden pagar en efectivo o vía tarjetas de débito y de crédito. Su plataforma incorpora Grability, una plataforma de comercio electrónico utilizada por otros detallistas como Walmart, El Corte Inglés, Grupo Exito y Cencosud. La inversión inicial en Rappi era de 2 millones de USD, y más tarde Rappi se metió a Y-Combinator lo que generó inversión adicional.

En septiembre de 2018, Rappi logró una valoración de los US\$1 000 millones, convirtiéndose así en el primer "unicornio" de Colombia.

Ruta: Se trata de un camino, carretera o vía que permite transitar desde un lugar hacia otro. En el mismo sentido, una ruta es la dirección que se toma para un propósito. En las comunicaciones, la ruta de enlaces es el conjunto de puntos que permiten unir dos puntos extremos.

Ahora traeremos los conceptos vistos en clase que nos pueden ayudar a solucionar el problema:

Árbol AVL: Un árbol AVL es un árbol binario de búsqueda que cumple con la condición de que la diferencia entre las alturas de los subárboles de cada uno de sus nodos es, como mucho 1 o -1. La denominación de árbol AVL viene dada por los creadores de tal estructura (Adelson-Velskii y Landis). Recordamos que un árbol binario de búsqueda es un árbol binario en el cual cada nodo cumple con que todos los nodos de su subárbol izquierdo son menores que la raíz y todos los nodos del subárbol derecho son mayores que la raíz. Recordamos también que el tiempo de las operaciones sobre un árbol binario de búsqueda son $O(\log n)$ promedio, pero el peor caso es $O(n)$, donde n es el número de elementos. La propiedad de equilibrio que debe cumplir un árbol para ser AVL asegura que la profundidad del árbol sea $O(\log(n))$, por lo que las operaciones sobre estas estructuras no deberán recorrer mucho para hallar el elemento deseado. El tiempo de ejecución de las operaciones sobre estos árboles es, a lo sumo $O(\log(n))$ en el peor caso, donde n es la cantidad de elementos del árbol. Sin embargo, y como era de esperarse, esta misma propiedad de equilibrio de los árboles AVL implica una dificultad a la hora de insertar o eliminar elementos: estas operaciones pueden no conservar dicha propiedad.

Teoría de Grafos: En matemáticas y en ciencias de la computación, la teoría de grafos (también llamada teoría de las gráficas) estudia las propiedades de los grafos (también llamadas gráficas). Un grafo es un conjunto, no vacío, de objetos llamados vértices (o nodos) y una selección de pares de vértices, llamados aristas (edges en

inglés) que pueden ser orientados o no. Típicamente, un grafo se representa mediante una serie de puntos (los vértices) conectados por líneas (las aristas).

Los grafos se pueden representar de diferentes maneras;

- **Lista de incidencia** - Las aristas son representadas con un vector de pares (ordenados, si el grafo es dirigido), donde cada par representa una de las aristas.
- **Lista de adyacencia** - Cada vértice tiene una lista de vértices los cuales son adyacentes a él. Esto causa redundancia en un grafo no dirigido (ya que A existe en la lista de adyacencia de B y viceversa), pero las búsquedas son más rápidas, al costo de almacenamiento extra.
- **Matriz de incidencia** - El grafo está representado por una matriz de A (aristas) por V (vértices), donde [arista, vértice] contiene la información de la arista (1 - conectado, 0 - no conectado)
- **Matriz de adyacencia** - El grafo está representado por una matriz cuadrada M de tamaño n^2 , donde n es el número de vértices. Si hay una arista entre un vértice x y un vértice y, entonces el elemento $m_{x,y}$ es 1, de lo contrario, es 0.

Camino: Se denomina camino (algunos autores lo llaman cadena si se trata de un grafo no dirigido) en un grafo dirigido a una sucesión de arcos adyacentes: $C = \{(v_1, v_2), (v_2, v_3), \dots, (v_{n-1}, v_n), v_i \in V\}$.

Circuito: Un circuito (o ciclo para grafos no dirigidos) es un camino en el que coinciden los vértices inicial y final, un circuito se dice simple cuando todos los arcos que lo forman son distintos y se dice elemental cuando todos los vértices por los que pasa son distintos. La longitud de un circuito es el número de arcos que lo componen. Un bucle es un circuito de longitud 1.

Algoritmos de búsqueda: En teoría de grafos existen dos formas de hacer búsqueda dentro del grafo: oportunidad aprenderemos a realizar **Algoritmos de Búsqueda en Anchura** (BFS, por sus siglas en inglés: "Breadth-First Search") y

de **Búsqueda en Profundidad** (DFS, por sus siglas en inglés: “Depth-First Search”). Ambas técnicas constituyen métodos sistemáticos para visitar todos los vértices y arcos del grafo, exactamente una vez y en un orden específico predeterminado, por lo cual podríamos decir que estos algoritmos simplemente nos permiten hacer recorridos controlados dentro del grafo con algún propósito. Siendo la búsqueda una de las operaciones más sencillas y elementales en cualquier estructura de datos, se han estandarizado el uso de estos algoritmos para ello, por lo que se conocen como algoritmos de búsqueda. Sin embargo, es importante resaltar que pueden utilizarse para muchísimas otras operaciones con grafos que no necesariamente incluyan la búsqueda de algún elemento dentro del grafo. Ambas técnicas constituyen métodos sistemáticos para visitar todos los vértices y arcos del grafo, exactamente una vez y en un orden específico

Árbol de expansión: Dado un grafo conexo, no dirigido G . Un árbol de expansión es un árbol compuesto por todos los vértices y algunas (posiblemente todas) de las aristas de G . Al ser creado un árbol no existirán ciclos, además debe existir una ruta entre cada par de vértices.

Algoritmo de Prim: Es un algoritmo perteneciente a la teoría de los grafos para encontrar un árbol de recubrimiento mínimo en un grafo conexo, no dirigido y cuyas aristas están etiquetadas.

En otras palabras, el algoritmo encuentra un subconjunto de aristas que forman un árbol con todos los vértices, donde el peso total de todas las aristas en el árbol es el mínimo posible. Si el grafo no es conexo, entonces el algoritmo encontrará el árbol de recubrimiento mínimo para uno de los componentes conexos que forman dicho grafo no conexo. Cabe tener en cuenta, que el algoritmo de prim necesita un nodo (Vértice) inicial.

El algoritmo fue diseñado en 1930 por el matemático Vojtech Jarník y luego de manera independiente por el científico computacional Robert C. Prim en 1957 y

redescubierto por Dijkstra en 1959. Por esta razón, el algoritmo es también conocido como algoritmo DJP o algoritmo de Jarnik.

Algoritmo de Kruskal: Cumple la misma función del algoritmo de Prim, pero este no necesita un nodo inicial, este ordena sus aristas y empieza a conectar una por una hasta obtener el árbol de recubrimiento mínimo.

Algoritmo de Dijkstra: También llamado algoritmo de caminos mínimos, es un algoritmo para la determinación del camino más corto dado un vértice origen al resto de vértices en un grafo con pesos en cada arista. Su nombre se refiere a Edsger Dijkstra, quien lo describió por primera vez en 1959.

Algoritmo de Floyd-Warshall: Es un algoritmo de análisis sobre grafos que permite encontrar el camino mínimo en grafos dirigidos ponderados. El algoritmo encuentra el camino entre todos los pares de vértices en una única ejecución, constituyendo un ejemplo de programación dinámica.

Fuentes:

<https://es.wikipedia.org/wiki/Rappi>

<https://edukavital.blogspot.com/2013/01/definicion-de-ruta-compendio-de.html>

<http://es.tldp.org/Tutoriales/doc-programacion-arboles-avl/avl-trees.pdf>

http://www.unipamplona.edu.co/unipamplona/portallG/home_23/recursos/general/11072012/grafos3.pdf

<http://decsai.ugr.es/~jfv/ed1/c++/cdrom4/paginaWeb/grafos.htm>

<https://www.bibliadelprogramador.com/2014/04/algoritmos-de-busqueda-en-anchura-bfs-y.html>

<https://jariasf.wordpress.com/2012/04/19/arbol-de-expansion-minima-algoritmo-de-kruskal/>

[https://www.ecured.cu/Algoritmo de Prim](https://www.ecured.cu/Algoritmo_de_Prim)

[https://www.ecured.cu/Algoritmo de Dijkstra](https://www.ecured.cu/Algoritmo_de_Dijkstra)

<https://www.ecured.cu/Floyd-Warshall>

Paso 3. Búsqueda de soluciones creativas:

Paso 4. Transición de las ideas a los diseños preliminares:

Paso 5. Evaluación y selección de la mejor solución: