UniUbi

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

## Introducción

El problema a resolver es mediante la introducción de datos crear un grafo, que contenga Nodos que sean una colección de nodos(salon,piso) un GPS que ayude a los estudiantes de algún campus universitario para no solo ayudarlos a llegar a su salón de clase si no también se busca que esté usando una base de datos personal del usuario para que sea el horario del estudiante que vaya actualizando sus ubicaciones a alcanzar.

## Problema

Desarrollar una interfaz que reciba datos para generar su propio grafo.

Desarrollar un método para vincular el horario del estudiante con su aplicación.

Desarrollar un pathFinder que con la información agregada tenga un plan de rutas a seguir.

## Estado Del Arte

Ya desde antes hemos visto como hay una necesidad de geolocalización y redireccionarlos de la manera más eficaz a nuestro destino.

* [https://www.waze.com](https://www.waze.com/es-419)

Podemos observar varios exponentes, al parecer es una idea recurrente

* <https://prezi.com/7vnefbt9rnyz/gps-universitario/>
* <https://manualbiotec.wixsite.com/inicio/gps-universitario>

## Problema Específico

Generar un árbol de recorridos, con un pathfinder que se actualice con cada horario del alumno.

## Solución Propuesta

para alojar la información se usará un formato csv para la matriz de adyacencia

Para el algoritmo de pathfinding se usará el A\* para la solucion de caminos, que se actualizará cada nodo que se avanza y se recalcula cada nodo que se aleja

## Fundamentos De La Solución

*“Como el objetivo es encontrar solamente el camino más corto entre dos nodos, parece una pérdida de tiempo y recursos procesar todos los caminos desde un nodo a todos los otros como lo hace el algoritmo de Dijkstra. El conocimiento de saber la posición del nodo destino no se ocupa para nada durante la búsqueda. El algoritmo A\*, se hace cargo de esto agregando una heurística para guiar la búsqueda hacia el destino, h[v], utilizando f[v]= g[v] + h[v] en vez de sólo g[v]. El algoritmo A\* será explicado en detalle en el próximo capítulo. [4] “*

*“Análisis de Algoritmos Pathfinding Danilo Alejandro Cuevas Guzmán INFORME FINAL DEL PROYECTO PARA OPTAR AL TÍTULO PROFESIONAL DE INGENIERO CIVIL EN INFORMÁTICA “*

*al parecer es una manera en la que nos ahorramos el trabajo de calcular el viaje a todos los caminos.*

{N,A}  
donde N es una colección de nodos, que contengan a su vez una colección de nodos que contengan el salón y el piso del objetivo

donde A son los posibles caminos a tomar

## Resultados

## Conclusión