**Week 1: Project Understanding, Planning, and Campus Modeling**

**Project Title:** BotBrain: An Intelligent Campus Navigator for Chanakya University.

**Abstract:** This project involves the design and implementation of "BotBrain," an intelligent agent to assist students in navigating the Chanakya University campus. The system will model the campus as a graph and utilize various search algorithms like Breadth-First Search (BFS), Depth-First Search (DFS), Uniform Cost Search (UCS), and A\* Search, to identify optimal routes between any two points on campus. The agent will be delivered as a complete application with a user-friendly graphical interface, processing navigation queries to provide the shortest path, total distance, and an estimated walking time. The project's primary goal is to apply theoretical AI concepts to solve a practical, real-world navigation problem.

**Problem Statement:** New and existing students often face challenges in finding the quickest or most efficient routes between various buildings on a large university campus. This project aims to solve this by creating an intelligent navigation tool that provides optimal paths based on user queries, saving time and effort.

**Objectives**:

To model the Chanakya University campus as a weighted graph with at least 12 key locations.

To implement four search algorithms: BFS, DFS, UCS, and A\* Search, for pathfinding.

To develop a simple text-based interface for users to input source and destination queries.

To display the calculated path, total distance, and estimated walking time for the user.

Allow users to select their source and destination, choose a search algorithm, and view the results visually.

**Scope**: The project is limited to the digital navigation of the Chanakya University campus. The core functionality will include pathfinding between 12 specified locations using four distinct search algorithms. The agent will operate from the text-based interface. The final submission will include the source code, a technical report, a live demonstration, and a GitHub repository with documentation

**Functional Requirements:**

Back-End:

Model the campus as a graph with 12 locations and weighted paths.

Implement BFS, DFS, UCS, and A\* search algorithms.

Calculate path, distance, and walking time.

Graphical User Interface (GUI):

Dropdowns for selecting start and destination points.Option to choose a search algorithm.A "Find Path" button to start the search.Display a visual map of the campus. Highlight the calculated route on the map. Show results: the path taken, total distance, and estimated time.Allow users to click on locations to get more information.

Algorithm Analysis: Display the number of nodes each algorithm explored to find the path.

**Data Requirements**

Campus Graph Data:

Locations (Nodes): A list of 12 specific buildings (Main Gate, Admin Building, etc.).

Paths (Edges): A defined set of connections between buildings, including distances in meters.

Coordinates: X, Y coordinates for each location to calculate the A\* search heuristic.

Building Information:A simple key-value store or dictionary mapping each location to a brief description (e.g., "Library: Open 8 AM - 10 PM").

**Tools and technical list:**

Programming language – python will handle everything: the data, the calculations, the visuals, and the user interaction.  
for project’s UI we can use libraries like PySimpleGUI, Tkinter, if possible PyQt or PySide

To use a real world Map API from google maps api or openstreetmap: These services can provide a satellite view of the Chanakya University campus

Ms word,powerpoint for documentation and others.