# Weekly Progress Report - Week 3

## Project: Campus Graph Modeling for Autonomous Navigation

This week, the project transitioned from basic verification to the implementation and refinement of the core navigation algorithms. The focus was on building a robust pathfinding system capable of producing clear and efficient routes.

### 1. Overview

Following the verification of the campus graph model, this week was dedicated to operationalizing the data structure. The primary goal was to fully implement the **Dijkstra’s and A\* algorithms** to handle route requests and to begin converting the raw algorithmic output into a user-friendly format. This phase is crucial for transforming the static graph model into a dynamic navigation tool.

### 2. Process Followed

a) **Advanced Algorithm Implementation:** While Dijkstra's was initially tested in Week 2, this week focused on a more robust implementation to handle complex path queries across the entire network. Additionally, the **A\* search algorithm**was implemented to find optimal routes more efficiently by incorporating a heuristic based on the straight-line distance to the target node.

b) **Route Generation & Formatting:** A module was developed to process the sequence of nodes returned by the pathfinding algorithms. This module generates a simplified, textual representation of the route, listing key landmarks and turns. This ensures the output is more intuitive than a raw list of coordinates or path points.

c) **Performance Optimization:** Based on the issues with high node density identified in Week 2, the graph's data structure was optimized to reduce redundant edges and simplify pathfinding calculations, leading to faster query times.

### 3. Key Findings

* The A\* algorithm consistently found optimal paths with fewer computational steps compared to a pure Dijkstra’s search, demonstrating improved efficiency.
* The system can now generate a clear, step-by-step route between any two major landmarks on campus.
* The graph optimization successfully addressed the issue of redundant edges, improving both the speed and clarity of the route results.

### 4. Issues Encountered

* Fine-tuning the heuristic for the A\* algorithm required careful testing to ensure it remained both efficient and accurate.
* Developing a clear and concise route description proved challenging, as some paths contain many intermediate points that need to be summarized intelligently.

### 5. Week 3 Outcomes

* Fully implemented **Dijkstra’s and A\* pathfinding algorithms**.
* Developed a module for **human-readable route generation**.
* Optimized graph structure for improved performance.
* Completed the core navigation functionality of the proje