## Campus Navigator Project: Week 4 Final Report

Project Title: BotBrain Campus Navigator  
Week 4 Focus: Documentation, Live Demo Preparation, Final Submission

## 1. Week 4 Objectives

* Finalize technical documentation according to IEEE format and submission guidelines.
* Prepare campus navigation chatbot for live demo: ensure full functionality of the Flask web application.
* Complete code commenting, README, and test cases.
* Write and organize the final project report, covering agent analysis, algorithm implementation, and feature comparison.
* Record screencast demo (optional, not included in this document).

## 2. Final System Overview

* Environment: Chanakya University campus graph modeled with 8 key building nodes and weighted edges for walking distances.
* Agent Architecture (PEAS Analysis):
  + Performance: Fast, accurate navigation, minimum path length, correct building information
  + Environment: Digital campus map; real-world campus navigation
  + Actuators: Text responses; path visualization on map
  + Sensors: User chat/input query (source/destination/building info)
* Agent Type: Goal-based agent using Uniform Cost Search (UCS) for route planning, with simple reflex for FAQ responses.

## 3. Search Algorithm Implementation

* Implemented: Uniform Cost Search (UCS)
* Data Structures: Priority queue (heapq), graph represented as adjacency dictionary.
* Sample Query: "How to go from Main Gate to Admin Block?"
  + UCS path: Main Gate → Admin Block (260 meters, estimated time: 3 min 7 sec)
  + API: /find\_path returns steps, total distance, estimated walking time, and nodes explored.

## 4. Web Application Features

* Interface: Responsive web app with campus map visualization (Leaflet.js), search box, dropdown selectors, and campus logo.
* Navigation: User selects source & destination, and system displays optimal UCS path with distance, time, nodes explored.
* Chatbot: NLP-based assistant answers navigation and building info queries, e.g., “Where is Admin Block?” or “Hostel to Main Gate directions.”

## 5. Documentation and Code Quality

* README: Usage instructions, setup guide, and example queries.
* Code Comments: Inline explanations for algorithms, class structures, API endpoints.
* Modularity: Separation of campus graph, search logic, Flask API, and frontend interface for maintainability.

## 6. Testing and Validation

* Functional Tests: Checked all valid source-destination pairs. Verified correct path and output for edge cases (same source/destination, unreachable locations).
* Unit Tests: For UCS algorithm, graph path calculations, chat response logic.

## 7. Comparison Table & Discussion

|  |  |  |  |
| --- | --- | --- | --- |
| Query | UCS Path | Distance | Nodes Explored |
| Main Gate → Cricket Ground | Main Gate→Admin Block→Cricket Gr. | 1260 m | 3 |
| Hostel → Admin Block | Hostel→Faculty Housing→Engg.→Admin | 530 m | 4 |
| Main Gate → Hostel | Main Gate→Admin Block→Food Court→Hostel | 910 m | 4 |

* Result: UCS consistently finds the shortest path for each query, along with node exploration counts.

## 8. Final Submission Details

* Code: Flask Python app with UCS implementation, web frontend (HTML, CSS, JS), and chatbot logic.
* Report: PDF file following IEEE format, summarizing all technical and agent aspects, sample runs, and code snippets.
* Test Cases: Inputs/outputs and description included in the documentation.

**TEST CASES EXECUTED:**

* Navigation Queries: 15+ location combinations tested
* FAQ Responses: 20+ question variations validated
* Edge Cases: Invalid inputs, typos, unknown locations

**SAMPLE TEST RESULTS:**

* Query: “How to go from Main Gate to Admin Block?”
* Response: “From Main Gate to Admin Block: Walk 260 meters. Path: Main Gate → Admin Block. Estimated time: 3 minutes 7 seconds.”
* Query: “Where is the Food Court?”
* Response: “Food Court: Cafeteria, Multiple food outlets, Open 7 AM – 10 PM”

**NAVIGATION PATH VALIDATION**

* ALGORITHM PERFORMANCE TESTING
* - Uniform Cost Search (UCS): Consistently found optimal paths
* - Path Validation: Cross-verified against manual distance calculations
* - Route Optimization: All paths confirmed to be shortest possible routes
* KEY TEST ROUTES
* - Main Gate → Admin Block: 260m
* - Hostel → Cricket Ground: 500m
* - Engineering Block → Food Court: 220m

**Summary:**  
The BotBrain Campus Navigator project now supports accurate campus navigation via Uniform Cost Search, provides real-time results in both web interface and chatbot formats, and is ready for final submission as per assignment Week 4 deliverables. All objectives have been met, and the system is robustly documented and tested for deployment.

