# ParkMe | MU Interim Report

Student: Redon Ferizi (21358591) ([Redon.ferizi.2022@mumail.ie](mailto:Redon.ferizi.2022@mumail.ie))

Supervisor: Dr Aidan Mooney

# Overall Project Objectives

The primary objective of this project is to develop “ParkMe | MU,” a comprehensive web application designed to streamline parking management at Maynooth University. The app aims to:

* Provide **real-time parking availability** across campus.
* Facilitate **easy check-ins and check-outs** for users.
* Offer **navigation assistance** to available parking spots.
* Deliver **analytics and insights** to help users plan their commute.
* Enhance the overall parking experience for **students**, **staff**, and **visitors**.
* Reduce parking congestion and improve the efficiency of campus parking

Table of Contents

[Overall Project Objectives 2](#_Toc182818220)

[Description of Work Completed 3](#_Toc182818221)

[Description of Work Completed 3](#_Toc182818222)

[Evidence of Work Completed and Issues Encountered 3](#_Toc182818223)

[Requirements Gathering: 3](#_Toc182818224)

[System Design: 3](#_Toc182818225)

[Frontend Development: 3](#_Toc182818226)

[Backend Development: 4](#_Toc182818227)

[Database Setup: 4](#_Toc182818228)

[Feature Implementation: 4](#_Toc182818229)

[Literature review 5](#_Toc182818230)

[Background and Motivation 5](#_Toc182818231)

[Prior Work in Web Development and Parking Systems 5](#_Toc182818232)

[Technical Resources and Inspirations 5](#_Toc182818233)

[Overview of Learning Gained Through GenAI 6](#_Toc182818235)

[Understanding Best Practices in Web Development 6](#_Toc182818236)

[Exploring and Integrating New Technologies 6](#_Toc182818237)

[Enhancing UI/UX Design Skills 7](#_Toc182818238)

[Outline of Future Work 7](#_Toc182818239)

[Complete Feature Implementation: 7](#_Toc182818240)

[Testing Phase: 7](#_Toc182818241)

[Deployment Preparation: 7](#_Toc182818242)

[Comprehensive Documentation: 7](#_Toc182818243)

[Final Presentation and Submission: 7](#_Toc182818244)

[Appendix - Additional Evidence 8](#_Toc182818245)

[Appendix A: Screenshots 8](#_Toc182818246)

[Appendix B: Wireframe 12](#_Toc182818247)

[Appendix C: Backend 13](#_Toc182818248)

[Appendix D: Gantt Chart 14](#_Toc182818249)

[Appendix E: UI Iterations 15](#_Toc182818250)

# Description of Work Completed

## Description of Work Completed

Substantial progress has been achieved during the initial project phases, following the set schedule and objectives. Key milestones include establishing the overall framework, setting up both the frontend and backend infrastructure, and developing core features that bring real-time functionality and user interactivity.

## Evidence of Work Completed and Issues Encountered

### Requirements Gathering:

* ***Completed:*** Created a requirements document detailing functional and non-functional requirements to guide development.
* ***Issues Encountered:*** Initially struggled to capture all necessary requirements, as the complexity of real-time parking data required input from potential users and stakeholders. This feedback took time to gather, and adjusting requirements to meet practical expectations while keeping the scope realistic was challenging.

### System Design:

* ***Completed:*** Developed system diagrams outlining the architecture of frontend and backend components and created user interface wireframes in Figma to illustrate layout and design (see Appendix B).
* ***Issues Encountered:*** Designing a seamless user interface that balanced functionality with simplicity proved challenging. Translating the wireframes into interactive features required numerous revisions to avoid clutter and keep the layout user-friendly. Additionally, coordinating frontend and backend integration was complex, requiring multiple design iterations to ensure data flowed smoothly across the system.

### Frontend Development:

* ***Completed****:* Built the user interface using HTML5, CSS3, and JavaScript. Implemented responsive design using Bootstrap 5, ensuring compatibility with various devices. This codebase contains approximately 4,500 lines across HTML, CSS, and JavaScript files (see Appendix A).
* ***Issues Encountered:*** Integrating Bootstrap effectively while maintaining unique design elements presented some styling challenges. Consistency across different screen sizes was difficult to maintain, and managing CSS conflicts (especially with the interactive map) required extensive debugging to keep the layout intact.

### Backend Development:

* ***Completed:*** Set up a server using Node.js and Express for efficient request handling and developed RESTful APIs to manage parking data retrieval, check-ins, check-outs, and real-time updates (see Appendix B).
* ***Issues Encountered:*** Initially struggled with configuring Express routing and managing API requests effectively, particularly when handling real-time data updates. MongoDB connection errors and port configuration issues also slowed development, as debugging these errors required familiarity with server environments and fine-tuning configurations for stability.

### Database Setup:

* ***Completed:*** Configured a MongoDB database to store user data, parking spot information, and log management. Designed schemas for users, parking spots, and transactions, which helped standardize data handling across the system (see Appendix C).
* ***Issues Encountered:*** Designing efficient schemas was initially a challenge, especially when balancing flexibility with database efficiency. Implementing relationships between data types (such as users and parking spots) created complexity in database structure, leading to performance issues when querying large datasets. Debugging these issues involved optimizing queries and fine-tuning schema configurations.

### Feature Implementation:

* Real-Time Availability: Integrated Chart.js for dynamic data visualization, providing real-time availability insights.
* Check-In/Check-Out System: Created a system to update parking availability based on user input.
* Interactive Map: Utilized Leaflet to allow users to view real-time parking availability on a campus map.
* Navigation Assistance: Integrated Google Maps API to guide users to available parking spots.
* User Feedback System: Added a feedback form to gather user feedback and drive improvements.

***Issues Encountered****:* Integrating each of these features came with unique challenges:

* *Real-Time Availability:* Achieving accurate real-time updates involved frequent data refreshes, which impacted performance and required adjustments to API calls and Chart.js configuration.
* *Check-In/Check-Out System:* Implementing check-in and check-out logic was complex, as user input had to update availability without data conflicts, leading to extensive debugging.
* *Interactive Map:* Leaflet integration faced compatibility issues with CSS, which disrupted the layout on smaller screens. Aligning Leaflet’s color-coded markers with accessibility requirements (e.g., colourblind-friendly colours) required multiple adjustments.

# Literature review

## Background and Motivation

Inspired by challenges in finding parking at Maynooth University, the project aims to improve parking management for students, staff, and visitors.

## Prior Work in Web Development and Parking Systems

* **Internship at Electric Ireland:** Gained expertise in CMS and web systems, directly applicable to this project.
* **Academic Studies:** Web development modules provided skills in HTML, CSS, JavaScript, and modern frameworks.

## Technical Resources and Inspirations

The project utilises several state-of-the-art technologies and practices. The following resources and prior work form the technical foundation of this project:

1. **Mapping Systems:** Leaflet and Google Maps API for dynamic maps and navigation.

2. **Real-Time Visualisation:** Chart.js for real-time data representation.

3. **Scalable Backend:** Node.js, Express, and MongoDB for real-time data management.

4. **Accessibility:** Designed for inclusivity with colourblind-friendly features and intuitive UI.

# Overview of Learning Gained Through GenAI

Generative AI tools like **ChatGPT** and **GitHub Copilot** have been invaluable resources throughout the development of the ParkMe | MU project.

### Understanding Best Practices in Web Development

* **Usage:** Leveraged GenAI to gain insights into industry-standard best practices for both frontend and backend development.

### Exploring and Integrating New Technologies

* **Usage:** Utilised GenAI to explore and understand new technologies and APIs relevant to the project’s requirements.

### Enhancing UI/UX Design Skills

* **Usage:** Employed GenAI to improve user interface and user experience design by receiving suggestions on layout, colour schemes, and responsive design techniques.

# Outline of Future Work

### Complete Feature Implementation:

Finish developing features like push notifications, user history, and advanced analytics.

### Testing Phase:

Conduct thorough unit, integration, and user acceptance testing.

Address any bugs or performance issues identified.

### Deployment Preparation:

Optimise the application for deployment on a scalable hosting platform (e.g., Heroku or AWS).

Ensure security best practices are in place.

### Comprehensive Documentation:

Develop user manuals, API documentation, and technical guides.

### Final Presentation and Submission:

Prepare presentation materials, including slides and a live demo

# Appendix - Additional Evidence

## **Appendix A: Screenshots**

**Figure 0:** *landing page*

A screenshot of a parking lot

Description automatically generated

**Figure 1:** *Features under landing*

A screenshot of a screen

Description automatically generated

**Figure 2:** *Occupancy Graph under features*

A screenshot of a parking permit

Description automatically generated

**Figure 3:** *Interactive App with Realtime data with legend*

A screenshot of a map

Description automatically generated

**Figure 4:** *Admin page*  A screenshot of a phone

Description automatically generated

**Figure 5:** *Google maps integration with different icons*

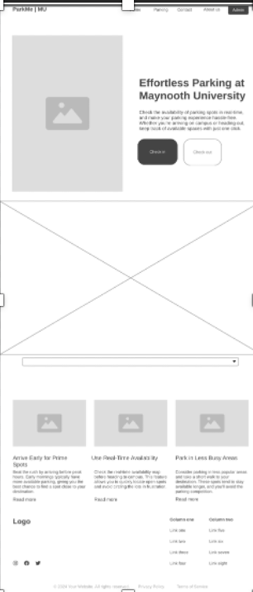
A screenshot of a map

Description automatically generatedA map with a blue line

Description automatically generated

## **Appendix B: Wireframe**

**Figure 6:** *Initial Wireframes*

A screenshot of a computer

Description automatically generated

## **Appendix C: Backend**

**Figure 7:** *MongoDB Dashboard*

A screenshot of a computer

Description automatically generated

**Figure 8:** *MongoDB Collection Sample*

A screenshot of a computer

Description automatically generated

**Figure 9:** *Server.js Config*

A screen shot of a computer program

Description automatically generated

**Figure 10:** *Seed.js with parking spot data for mongoDB*

A screenshot of a computer program

Description automatically generated

## **Appendix D: Gantt Chart**

**Figure 11:** *YouTrack Gantt Chart*

A screenshot of a computer

Description automatically generated

**Figure 12:** *You track gantt chart issues/items*

A screenshot of a computer

Description automatically generated

## **Appendix E: UI Iterations**

### 11th October

A screenshot of a parking map

Description automatically generated

### 25th October

A screen shot of a parking management

Description automatically generatedA map of a parking spot

Description automatically generated

### 6th November

A screenshot of a parking lot

Description automatically generatedA screenshot of a map

Description automatically generated

### 13th November

A screenshot of a parking lot

Description automatically generatedA screenshot of a graph

Description automatically generated

A screenshot of a parking permit information

Description automatically generatedA screenshot of a map

Description automatically generated