

**School of Computer Science
University of Windsor
Master of Applied Computing**

**Course: COMP-8117 Applied Software Engineering
Instructor: Dr. Aznam Yacoub**



**University
of Windsor**

SPLASH - 3 Progress Report

ParkMate

“ParkMate - A smart parking application designed to simplify and enhance the parking experience by providing personalized solutions for locating, reserving, and managing parking spaces.”

Our Team:

Anshul Prajapati(110189295)

Nisthaben Patel(110184308)

Om Siddhapura(110187008)

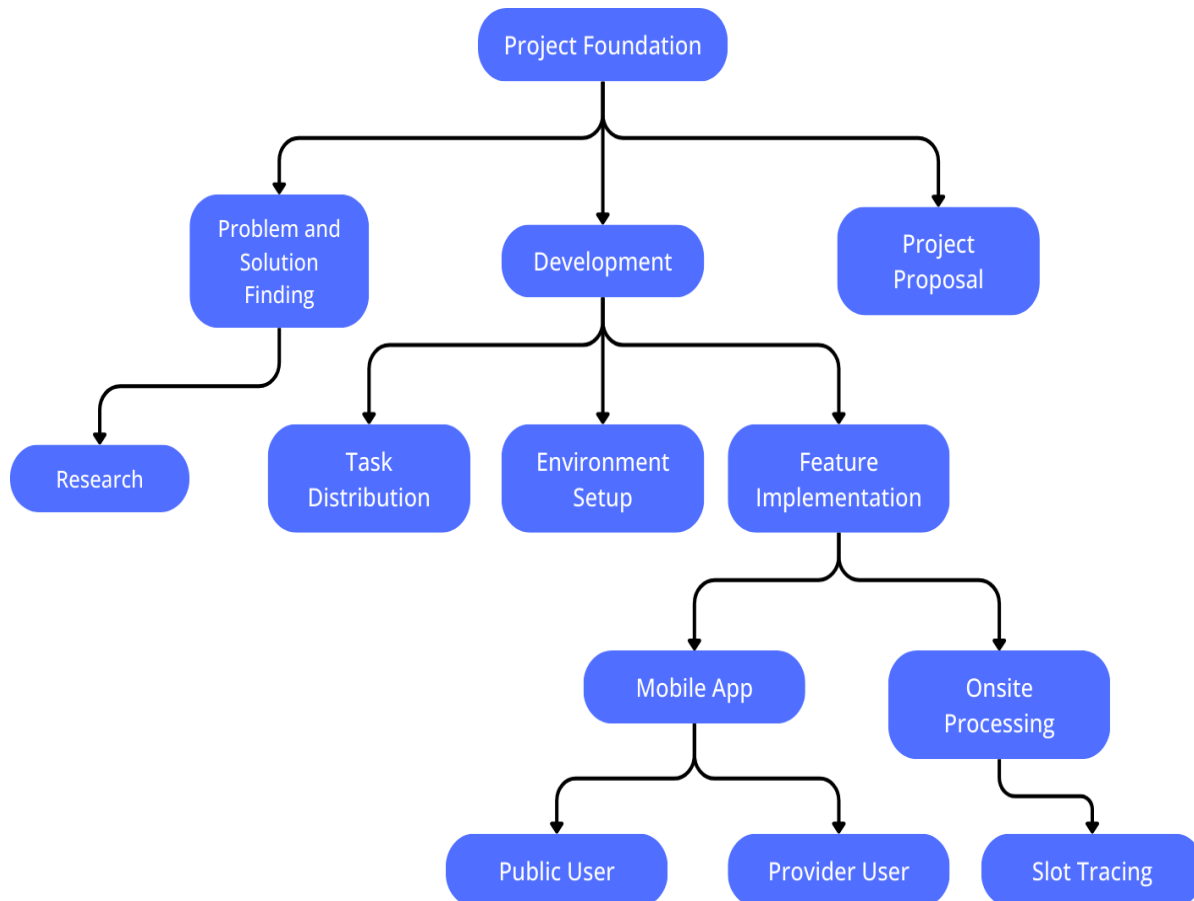
Ved Prajapati(110186761)

Table of Contents

1. Project Progress	1
1.1 Development Journey	1
1.2 Stability Metrics	2
2. Implementation Progress	3
2.1 Implemented Features	3
2.2 Mobile Application & Onsite processing	5
3. Resource Analysis	6
3.1 Budget Distribution	6
3.2 Development Metrics	7
4. Challenges and Solution	8
4.1 Technical Challenges	8
4.2 Follow up on issues	8
5. Future Development	10
5.1 Further Planned Development	10
5.2 Priority Areas	10
6. Learning Outcomes	12
6.1 Experiential Learning	12
6.2 Recognizing Software Engineering Scope	12
6.3 Key Insights into the Development Lifecycle	12
6.4 Valuable Lessons Learned	13
7. Conclusion	14
7.1 Technical Completion	14
7.2 Learning Outcomes	14
7.3 Future Growth	14
8. Project Artifacts	16
8.1 Core Documentation	16
8.2 Application Screenshots	16
8.3 System Design	16
8.4 Artifact Metrics	17
9. Task Distribution For Report	18

1. Project Progress:

1.1 Development Journey



The development of the project was started after the team formation and the impactful problems were discovered after that in the development phase first the tasks were divided according to the role and then according to the Framework that each individual needs to use was initialized via setting up the environment setup process. Also the actual feature implementation was started. Under the feature implementation there are two segments: mobile app and the onsite processing module. For mobile applications, the public user related screen and provider user related screens were created while in the onsite processing the development of the slot tracing module using machine learning was performed. This is demonstrated in the above figure.

1.2 Stability Metrics

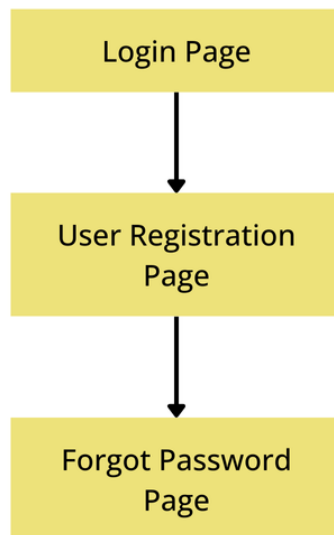
Stability Metrics

Metric	Value	Scope
Crash Free Session for Mobile App User	75.67	Require Fixing
Total Crashes	26	Require Fixing
Number of Users suffered the issue	5	Low

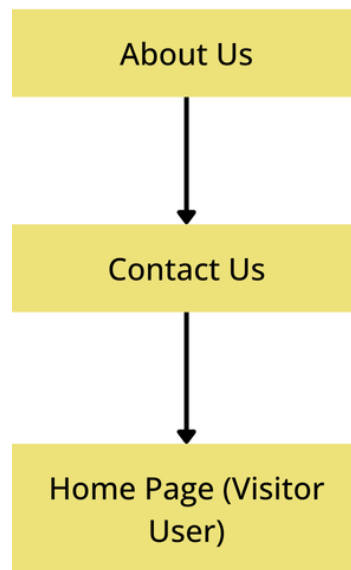
2. Implementation Progress

2.1 Implemented Features

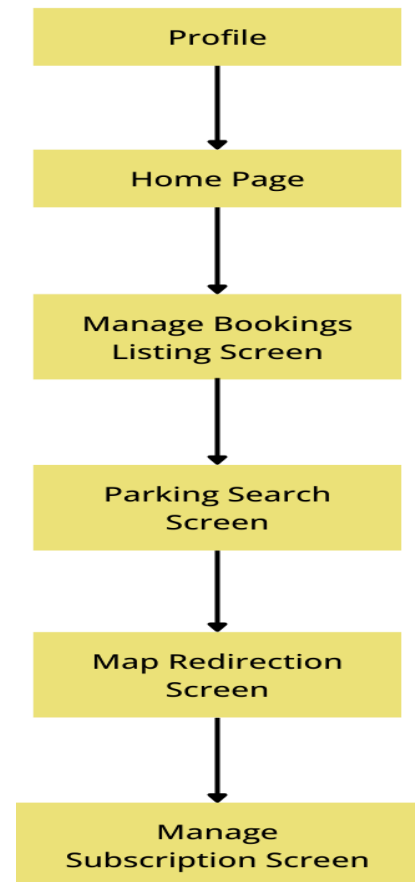
Authentication Module



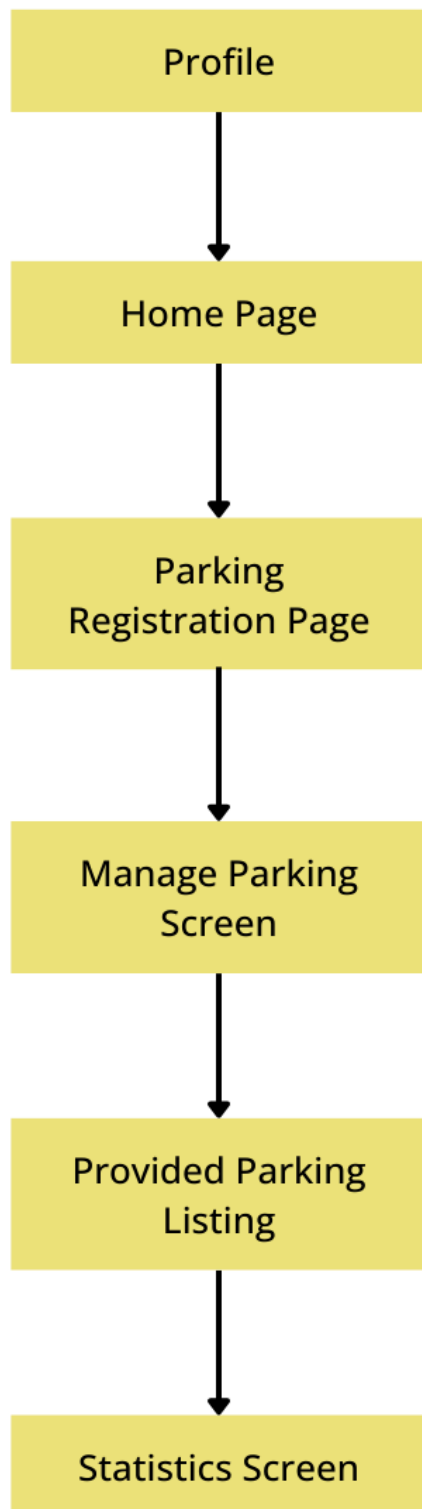
Basic Pages



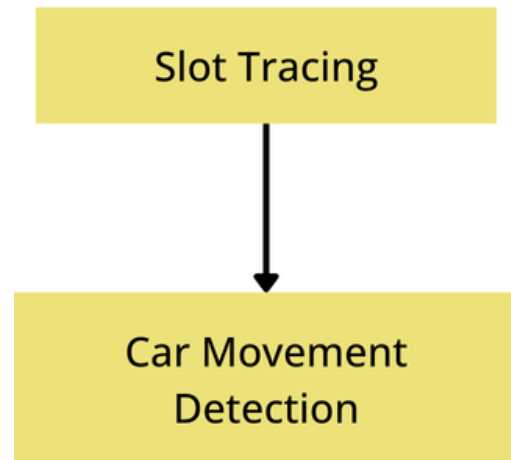
Public User Module



Provider User Module



Onsite Processing



Note: The flow shown in the above all the figures indicates the development till the point in the respective modules. In nutshell, The following implementation has been done in the project solution.

2.2 Mobile Application & Onsite Processing

Mobile Application

The application is designed to serve three types of users:

1. **Public Users:** They can search for available parking spaces, view details such as pricing and location, book slots, and manage their current and past reservations. Additionally, they can subscribe to parking services for extended periods and receive navigation assistance to their booked parking area.
2. **Providers:** Parking area owners can register their parking spots in the system, manage them, update pricing, and analyze revenue and occupancy statistics. They can also track parking slot usage and manage bookings efficiently.
3. **Visitors:** Guests can browse available parking spaces and search for locations, but they must register and log in to make bookings.

Onsite Processing

The system integrates real-time tracking and automation to enhance parking efficiency:

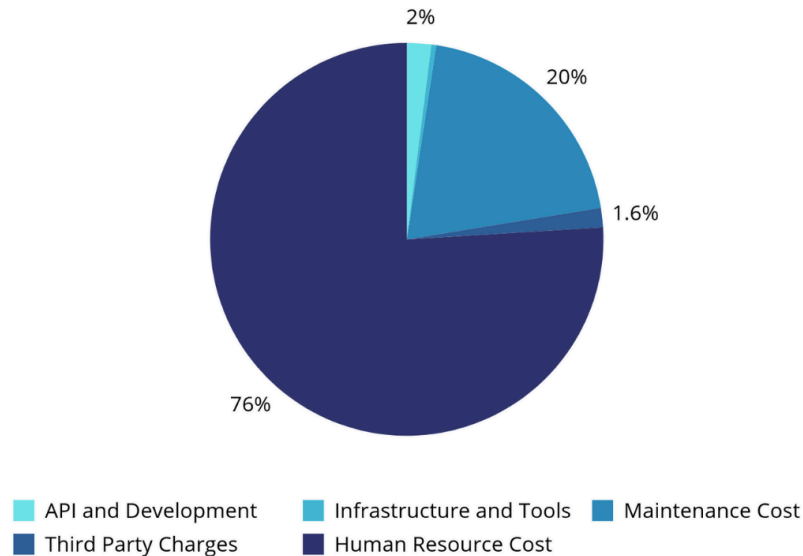
1. **Slot Detection & Tracking:** Using image processing and object detection techniques, the system accurately identifies individual parking slots. A top-mounted camera captures the parking area, and the system marks each slot with a unique identifier.
2. **Live Slot Availability:** The system continuously monitors the parking area and determines whether a slot is occupied or free based on real-time image analysis. This helps users find vacant spots without manual intervention.
3. **Number Plate Recognition & SMS Alerts:** When a car enters the parking lot, its number plate is detected using Optical Character Recognition (OCR) technology. If the vehicle has a prior booking, the system verifies the reservation and assigns a slot automatically. The user receives an SMS with details such as the parking slot number and duration of stay.

Note: Item 2 and 3 are yet to be implemented

3. Resource Analysis

3.1 Budget Distribution

Total Budget Distribution



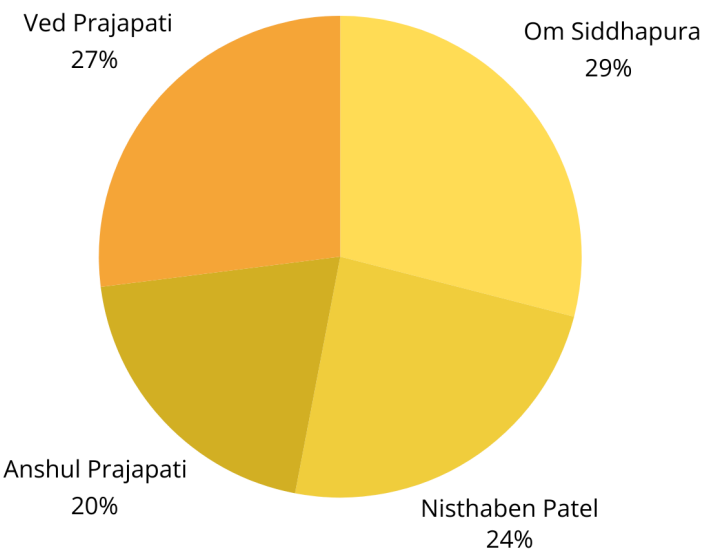
Note: the distribution of the cost is given till this point of time latest by report submission date.

Total budget for the application is 52,400 CAD now up to this point of time total budget used is 30000 CAD. Hence total remaining balance for the further project is 22400 CAD.

The largest portion, **76% (22,800 CAD)**, is allocated to Human Resource Cost, covering external services and integrations. Third Party Charges take **1.6% (480 CAD)**, ensuring the necessary hardware, software, and resources for smooth operations. Maintenance costs account for **20% (6,000 CAD)**, which is used for system upkeep and updates. Lastly, **API and development** receive **2% (600 CAD)**, supporting the creation and enhancement of application functionalities. This distribution ensures efficient resource allocation for the project's success. In conclusion, we are still left with 22400 CAD for our future costs. But we need to use the balance effectively to fit our project in the budget for the future events.

3.2 Development Metrics

Developer Task Distribution



Note: the distribution of the cost is given till this point of time latest by report submission date.

Task Type Wise Distribution

Task Type	Main Responsible Developer
API Integration and Frontend	Om Siddhapura
Architecture and Backend	Ved Prajapati
UI/UX and testing	Nisthaben Patel
Database Design	Anshul Prajapati

4. Challenges and Solution

4.1 Technical Challenges

During the project development a few technical issues were figured out. At the end the issues were resolved using a research based approach.

UI Component Responsiveness:

- **Issue:** Few listing screens such as current/previous booking listing screen and calendar component integration for the subscription module had issues related to the responsiveness where the components were not being fitted in the screen of different smartphones.
- **Solution:** Used React native supported library called React native material kit for generating reusable and responsive components.
- **Final Result:** The screen having trouble is now responsive for all the devices.

Deprecated Libraries

- **Issue:** Some of the libraries like react native map or open cv for python were causing compile time error due to incompatibility.
- **Solution:** Downgraded the development framework such as NodeJs and Python for compatibility.
- **Final Result:** The development went smoothly without any library support error.

Third party API Integration

- **Issue:** Multiple third party support available for implementing particular functionality. Which one to choose for long term support was difficult.
- **Solution:** Researched on the web related to the support and figured out the support that we want for our feature implementation.
- **Final Result:** All the features implemented easily through the use of the appropriate API.

Performance Optimization:

- **Issue:** Content loading performance issues In few screens were loading too much slowly.
- **Solution:** The reason was due to unnecessary data being fetched by default which will not be used immediately instead we implemented a way where only necessary data will be fetched for instantaneous screen.
- **Final Result:** 60% raised improvement in the loading time.

4.2 Follow up on issues

The issues were taken off by means of different support as follows:

- Detailed explanation on task allocation
- Frequent communication in teams

- Difficulty solving session
- Information learning from different sources

Meetings:

We organized meeting on several occasions in order to have update on the project along with the further task distribution

Meeting 1**Purpose:**

We organize a meeting on January 26 to have a view on the project proposal report and identify issues and solve it

Agenda:

- Identify issues in the project proposal.
- Have an insight on the problem and solution effectively.

Meeting 2**Purpose:**

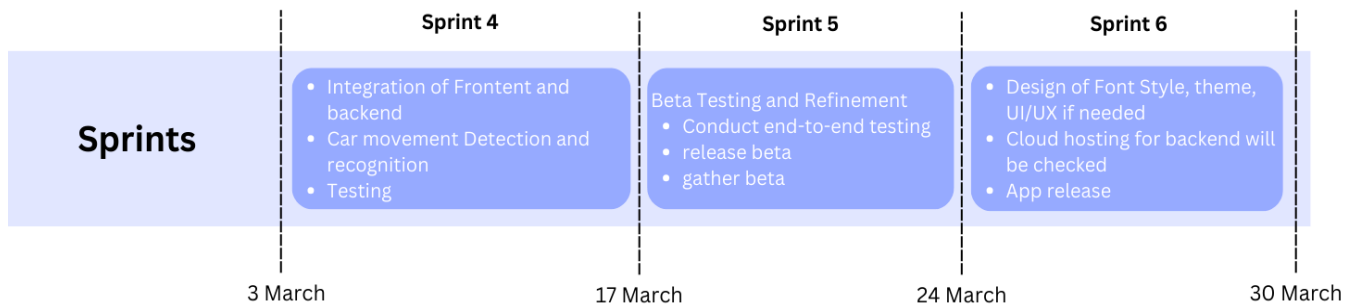
We organized a meeting on February 22 to review the tasks and also solve the problems being raised in the project.

Agenda:

- Review of work completed.
- Discussing challenges we are facing and finding solutions.
- Sharing useful tips and tricks from our experiences during working on these projects.
- Listing out tasks for the Next Week.

5. Future Development

5.1 Further planned development



Our further development plan consists of remaining sprint completion.

Sprint 4

Timeline: 03 Mar – 16 Mar

Description:

- Integration for Front End, backend and Database should be done where all the possible scenarios related testing will be performed for the integration.
- Car movement detection in the slot through the camera will also be done and number plate recognition should be completed using OCR technology.
- Testing of the number plate recognition, slot empty-ness detection and app integration need to be checked.

Sprint 5

Timeline: 17 Mar – 23 Mar

Description:

- Beta Testing and Refinement: Conduct end-to-end testing, release beta and gather feedback.

Sprint 6 (Final Release)

Timeline: 24 Mar – 30 Mar

Description:

- Design of font styles, Position UI/UX if needed, theme option exploration etc.
- Cloud hosting for database and backend will be checked and finally the app will be released.

5.2 Priority Areas

Our main focus would be more on the following aspects of the project.

1. Functionality Completion

- Ensure all the core features are fully developed and working.
- Addressing any missing functionalities that are essential.
- Optimizing the functionality for good performance and user experience.

2. Perform Integration

- Integrating various modules and components to provide a complete working app.

- Implementing APIs along with database and UI effectively.

3. Testing

- Performing individual testing for the developed item.
- Give an explicit session to the QA for the testing.
- Rapidly fixing bugs to enhance reliability and completeness.

6. Learning Outcomes

6.1 Experiential Learning

Creating a ParkMate application taught us a lot about who to build a software or any application. We learned more than just how to code. We discovered that making good software needs both strong coding skills and other important skills, like teamwork and problem-solving.

6.2 Recognizing Software Engineering Scope

Our experience has shown that software engineering covers much more than just coding and development, including different areas such as:

- Ensuring software scalability and performance optimization
- Designing system architecture and understanding its limitations.
- Integrating knowledge from various fields.
- Managing project timelines and resource allocation.

6.3 Key Insights into the Development Lifecycle

Through this process, we uncovered key insights into effective software creation:

1. Continuous Learning and Improvement

The team followed a progressive model of development and improvement.

Development Cycle:

- Practical implementation phases.
- Frequent evaluation sessions.
- Continuous growth and adaptation.
- Knowledge recording and sharing.

2. Technical Advancement

Our technical expertise expanded in many areas, including:

- Implications across the system
- Evaluating scalability factors
- Making architectural decisions
- Enhancing performance efficiency

3. Team Collaboration

All team members collaborate within the group when anyone is facing any difficulties or making important decisions.

Collaboration Highlights:

- The value of pair programming.
- Effective role allocation between team members.
- Exchanging new knowledge with team members.
- Defined communication standards.

6.4 Valuable Lessons Learned

Our journey provided valuable lessons across various aspects, including:

1. The diverse and complex nature of software engineering:

- Integrating various systems and platforms.
- Addressing scalability and performance challenges.
- Creating a valuable budget for the project.
- Balancing multiple technologies and tools

2. The Importance of continuous learning:

- Growth through tackling challenges.
- Reflective learning practices.
- Documenting and sharing knowledge.

3. Team collaboration factors:

- Effective communication.
- Clear role assignments.
- Regular knowledge sharing.

7. Conclusion

The ParkMate project has been an exciting path for us which is the combination of technology and solving real-world problems to provide a seamless parking experience. From the basic planning of the solution development and collaboration, we have generated a solution that simplifies parking along with improving efficiency for users.

7.1 Technical Completion

- Successful implementation of essential features for real-time availability and reservations of parking slots.
- Ensured consistent performance.
- Optimized resource utilization to acquire smooth functionality.
- Established an efficient technical base for future enhancements.

7.2 Learning Outcomes

- Expanded valuable insights into user experience and software design principles.
- Learned new technical concepts.
- Strengthened teamwork and collaboration throughout the project development.
- Upgraded project management strategies, from planning to execution of the project.
- Gather knowledge on effective documentation as well as presentation skills with better graphical user interface.

7.3 Future Growth

We are looking for extensive potential for the ParkMate application for its accomplishment in the market. With a solid foundation, this project paves the stage for innovative solutions in the parking systems.

AI & Smart Predictions

- **AI-driven Parking Predictions:** Utilizing machine learning to predict parking availability dependent on peak hours, weather conditions, and traffic patterns.
- **Personalized Recommendations:** Offering parking recommendations based on preferences of users, frequent locations, and previous parking history.

Payment & Subscription Models

- **One-Tap Payments:** Supporting online payment methods, credit/debit cards, and digital wallets for an efficient payment experience.
- **Membership & Subscription Plans:** Initiating monthly and yearly plans for persistent users with varied benefits like priority booking and exclusive discounts.
- **Dynamic Pricing Models:** Implementing surge pricing based on demand and during peak hours, while ensuring efficient utilization of parking slots.

Multi-City & Global Expansion

- **Scalability for Different Cities:** Adapting the platform in order to accommodate various urban parking infrastructures, regulations, and demand patterns.
- **International Expansion:** Expanding the application to the urban part of the country which is facing parking challenges beyond local markets.
- **Smart City Integration:** In order to make ParkMate an application to be the part of smart transportation systems, it will be collaborating with municipal authorization.

8. Project Artifacts

8.1 Core Documentation

Document	Description	Status
Project Proposal	Comprehensive Project Documentation	✓ Complete
Interim Progress Report	Project progress up to the current time	✓ Complete

8.2 Application Screenshots

Screenshot	Purpose	Status
Public User	Complete	✓ Complete
Provider User	Complete	✓ Complete
Slot Tracing	Complete	✓ Complete

8.3 System Design

UML and use case diagrams

Diagram	Purpose
Activity Diagram	✓ Complete
Data Flow Diagram	✓ Complete
ER Diagram	✓ Complete
Sequence Diagram	✓ Complete
State Diagram	✓ Complete
Use Case Diagram	✓ Complete

8.4 Artifact Metrics

Category	Count	Status
Documentation Files	1	✓ Complete
UML Diagrams	16	✓ Complete
Screenshots	28	✓ Complete
Technical Report	1	✓ Complete

Total Artifacts: 46 files across 3 directories.

9. Task Distribution for the Report

Name	Tasks
Om Siddhapura	<ul style="list-style-type: none">• Project progress• Implementation progress• Performed overall review
Nisthaben Patel	<ul style="list-style-type: none">• Resource analysis• Challenges and solution• Documentation format
Ved Prajapati	<ul style="list-style-type: none">• Future development and learning outcome• Performed overall review
Anshul Prajapati	<ul style="list-style-type: none">• Conclusion and project artifacts• Contributed to document format and design