

UrbanFlow Project LogBook

Daily Logs:

June 24, 2024

Objectives for the Day:

- Research on my chosen field of study: Software Systems
- Choose a suitable project

Tasks Completed:

- Chose a project focused on optimizing traffic signals with priority for emergency vehicles

Challenges Encountered:

- Integration of real-time data for dynamic priority management
- Testing and simulation of my experiments

Solutions:

- Utilize a Backend API and an Android app and installed in emergency vehicle drivers' phones to gather real-time GPS data
- Implement an API server for integration with traffic signals
- Develop a central server for priority management enhanced by machine learning

Reflections:

- The initial brainstorming and project selection were crucial for setting a clear direction.
- Transitioning from research to practical application highlighted the importance of aligning objectives with feasible solutions.

Next Steps:

- Begin gathering training data for the project.

June 25, 2024

Objectives for the Day:

- Gather training data

Tasks Completed:

- Researched available datasets of vehicle GPS data

Challenges Encountered:

- Lack of publicly available datasets suitable for my project

Solutions:

- Create a data collection software (app) to gather self-collected and cleaned data

Reflections:

- The importance of custom data collection was realized, ensuring data relevance and quality.

Next Steps:

- Develop the data collection software.
-

June 27, 2024**Objectives:**

- Create the data collection software

Tasks Completed:

- Created a simple Python-powered Android app using Kivy and Buildozer

Challenges Encountered:

- Unintuitive documentation of frameworks
- Limited development freedom and features

Solutions:

- Shifted to Android Studio for app development for better capabilities

Reflections:

- Choosing the right development tools is critical for project success.

Next Steps:

- Familiarize with Java and Kotlin and begin development in Android Studio.
-

June 28, 2024

Objectives for the Day:

- Familiarize with Java and Kotlin
- Begin development of the Android app for data collection

Tasks Completed:

- Revisited Java and Kotlin basics
- Set up Android Studio development environment
- Started working on the user authentication and login process

Challenges Encountered:

- Initial struggle with switching back to Java and Kotlin after years of not using them
- Configuration issues in Android Studio and Firebase, causing delays in setup

Solutions:

- Spend additional time on small projects to refresh knowledge of Java and Kotlin
- Use FastAPI and PostgreSQL for creating API server

Reflections:

- The importance of continuous learning and practice in maintaining technical skills.

Next Steps:

- Continue working on the app development and resolve any remaining configuration issues.
-

June 29 - July 4, 2024

Objectives for the Period:

- Establish API system and begin foundational development

Tasks Completed:

- June 29: Initiated the ProjectIdFormScreen and began setting up the API system, defining dependencies
- June 30: Developed the initial version of the Info Screen and started the README documentation
- July 1-2: Focused on debugging issues related to dependencies and the initial API integration
- July 3-4: Completed API Release v0.6.0, achieving basic functionality without data upload

Challenges Encountered:

- Dependency Management: Encountered difficulties in maintaining consistent dependencies, causing build issues
- API Integration: Initial API interaction bugs required intensive debugging

Solutions:

- Refined build setup and dependencies
- Extensive testing and debugging stabilized API integration

Reflections:

- Early stage debugging and dependency management are crucial for smooth development progress.

Next Steps:

- Advance data upload functionality and refine documentation.
-

July 5 - 6, 2024**Objectives for the Period:**

- Advance data upload functionality and refine documentation

Tasks Completed:

- July 5: Enhanced README documentation, clarifying setup and usage guidelines
- July 6: Continued development on the data upload feature, focusing on smooth integration

Challenges Encountered:

- Data Handling: Ensured secure and accurate transmission of user data

Solutions:

- Implemented robust validation and encryption for data handling

Reflections:

- Ensuring data security and integrity is essential for user trust and system reliability.

Next Steps:

- Improve data persistence and user experience in the app.
-

July 14 - 16, 2024**Objectives for the Period:**

- Improve data persistence and user experience in the app

Tasks Completed:

- July 14: Made logins persistent, added notification features
- July 15: Addressed data upload and cleaning processes
- July 16: Optimized the app by transitioning from SQLite to PostgreSQL

Challenges Encountered:

- Persistent Logins and ViewModel Bugs: Faced issues maintaining persistent logins and managing ViewModels

Solutions:

- Studied Kotlin app development, optimizing the use of ViewModels and ensuring consistent user sessions

Reflections:

- Optimizing user experience and data management is key for application effectiveness.

Next Steps:

- Develop and refine the AI model for route prediction.
-

July 17 - 19, 2024

Objectives for the Period:

- Develop and refine the AI model for route prediction

Tasks Completed:

- July 17: Prototype development for AI training models
- July 18: Tested multiple models including Random Forest, XGBoost, GRU, and others, incorporating hyperparameter tuning
- July 19: Evaluated model accuracy and began preparing for further integration

Challenges Encountered:

- Route Prediction Model Accuracy: Initial accuracy was extremely low (around 30%)

Solutions:

- Incorporated more complex models and utilized Optuna for hyperparameter tuning, significantly improving model performance

Reflections:

- Iterative model testing and tuning are crucial for achieving reliable AI performance.

Next Steps:

- Finalize the data collection software and improve system architecture documentation.
-

July 22 - 25, 2024

Objectives for the Period:

- Finalize the data collection software and improve system architecture documentation

Tasks Completed:

- July 22: Implemented Git LFS for large file management
- July 23: Created detailed flowcharts and diagrams of system architecture
- July 24-25: Focused on refining server and database integration, without completing integration of predictions or Maps API

Challenges Encountered:

- Data Management and Redundancy: Managing and storing large data files efficiently

Solutions:

- Utilized JSON for data storage and implemented chunked data uploads

Reflections:

- Effective data management strategies are critical for system efficiency.

Next Steps:

- Strengthen security and optimize server performance.
-

July 26 - 28, 2024

Objectives for the Period:

- Strengthen security and optimize server performance

Tasks Completed:

- July 26: Set up UrbanFlow Server with asynchronous database handling
- July 27: Enhanced authentication mechanisms, including restricting registration to localhost and implementing a two-fold auth system with tokens and API keys
- July 28: Implemented caching mechanisms to reduce latency and improve response times

Challenges Encountered:

- Secure Authentication: Required robust security measures to prevent unauthorized access

Solutions:

- Implemented a combination of token-based and API key authentication, alongside restricted registration protocols

Reflections:

- Robust security measures are fundamental for protecting user data and system integrity.

Next Steps:

- Finalize legal compliance and data handling policies.

July 29 - 30, 2024

Objectives for the Period:

- Finalize legal compliance and data handling policies

Tasks Completed:

- July 29: Developed comprehensive Privacy Policy and Terms & Conditions
- July 30: Finalized handling of GPS status and ensured compliance mechanisms were in place

Challenges Encountered:

- Legal Compliance: Ensuring adherence to data protection and privacy laws

Solutions:

- Conducted thorough research and created detailed legal documents to ensure compliance with relevant regulations

Reflections:

- Ensuring legal compliance is crucial for the legitimacy and sustainability of the project.

Next Steps:

- Prepare for further testing and deployment of the system.

Milestones and Reflections:

Milestone 1: Project Inception and Objective Setting

- **Date:** June 24, 2024
- **Achievements:** Chose the project focused on optimizing traffic signals with priority for emergency vehicles.
- **Reflection:** The initial brainstorming helped narrow down a specific problem area, setting a clear project direction.
- **Impact:** Established a solid foundation for subsequent work on data collection and system architecture.

Milestone 2: Development of Data Collection App

- **Date:** June 27 - July 6, 2024
- **Achievements:** Created a Python-powered Android app, later shifted to Android Studio for better development capabilities.
- **Reflection:** Transitioning to Android Studio allowed for more robust and feature-rich development, crucial for the project's success.
- **Impact:** Improved app performance and user experience, facilitating more reliable data collection.

Milestone 3: API System Establishment and Initial Release

- **Date:** June 29 - July 4, 2024
- **Achievements:** Established the API system, completed API Release v0.6.0.
- **Reflection:** Developing the API was challenging due to dependency management but was vital for integrating various system components.
- **Impact:** Enabled seamless communication between the app and server, laying the groundwork for further features.

Milestone 4: Implementation of Secure Authentication

- **Date:** July 26 - 27, 2024
- **Achievements:** Implemented a two-fold authentication system with tokens and API keys, restricted registration to localhost.
- **Reflection:** Addressing security concerns was critical for protecting user data and ensuring system integrity.
- **Impact:** Enhanced the security and trustworthiness of the system, making it more robust against potential threats.

Milestone 5: AI Model Development and Tuning

- **Date:** July 17 - 19, 2024
- **Achievements:** Improved route prediction model accuracy from 30% to a significantly higher rate using advanced models and hyperparameter tuning.
- **Data and Metrics:** Initial accuracy: 30%, Post-tuning accuracy: [Specific percentage, if available].

- **Reflection:** Experimentation with various models and tuning techniques was necessary to achieve reliable predictions.
 - **Impact:** Provided a more accurate and reliable system for predicting emergency vehicle routes, critical for optimizing traffic signals.
-

Major Challenges and Solutions:

1. **Extreme Lack of Accuracy in Route Prediction AI Model (30% Accuracy)**
 - **Solution:** Added more complex models (Random Forest, XGBoost, GRU) and utilized Optuna for hyperparameter tuning, significantly improving accuracy.
2. **Persistent Android Logins and ViewModel Errors**
 - **Solution:** Enhanced understanding of Kotlin app development and optimized ViewModel usage to resolve session persistence issues.
3. **Secure Authentication in UrbanFlow Server**
 - **Solution:** Restricted registration to localhost, implemented a two-fold authentication system with tokens and API keys.
4. **Async Database Use and Caching to Reduce Latency**
 - **Solution:** Implemented asynchronous database interactions and caching after extensive debugging.
5. **Data Redundancy and Saving in App**
 - **Solution:** Utilized JSON for data storage, with chunked data uploads to manage redundancy.
6. **Legal Compliance and Policy Research**
 - **Solution:** Developed Privacy Policy and Terms & Conditions after thorough research, ensuring all aspects were legally sound.