Group2 Product Backlog

Application Discussion: [User Story Definition]

- **As a:** [Developer]
- **I want to:** [define our user stories clearly]
- **So that I can:** [build the best application meet our expectation]

Priority: [High]
Estimated Points: [8]
Acceptance Criteria:

- [Well define user stories]
- [User stories should align with project timeline and goals]

User Story: [Google MAP Interface]

- **As a:** [User]
- **I want to:** [see google map interface on the web]
- **So that I can:** [interact with]

Priority: [High]
Estimated Points: [5]
Acceptance Criteria:

- [A web page with the Google Map feature]
- [When the user navigates to the page, a functional and interactive Google Map should be displayed]

User Story: [Bike Occupancy data]

- **As a:** [User]
- **I want to:** [access the latest bike occupancy data]

- **So that I can:** [decide which bike to choose]

Priority: [High]
Estimated Points: [15]
Acceptance Criteria:

- [When the user access the web, the bikes' location should be displayed on the map]
- [bikes' location should be accurate and up-to-date]
- [Collects info every 5 mins]

User Story: [Weather Information]

- **As a:** [User]
- **I want to:** [see weather information on the map]
- **So that I can:** [better choose trip options]

Priority: [Medium]
Estimated Points: [15]
Acceptance Criteria:

- [When the user access the web, weather information should be displayed on the map in suitable place]
- [weather information should contain relevant details such as temperature, humidity, and wind speed]
- [weather information should be up-to-date]
- [collect data about weather from OpenWeather every hour]

Software Necessity: [Software Necessities Installation]

- **As a:** [Developer]
- **I want to:** [have all necessary software installed]
- **So that I can:** [develop efficiently]

Priority: [High]
Estimated Points: [8]
Acceptance Criteria:

- [All softwares are correctly installed]
- [Have front-end development tools installed]
- [Have back-end development tools installed]

Software Necessity: [Preliminary data scraping]

- **As a:** [Developer]
- **I want to:** [scrape data from websites or APIs efficiently]
- **So that I can:** [extract and use the data for application development]

Priority: [High]
Estimated Points: [8]
Acceptance Criteria:

- [The scraper should be able to retrieve data from a specified website or API]
- [The scraper should be fully automatic and easy to run]
- [Extracted data should be stored in database automatically]

User Story: [Visualize current information]

- **As a:** [User]
- **I want to:** [access the latest data]
- **So that I can:** [make informed decisions]

Priority: [High]
Estimated Points: [12]
Acceptance Criteria:

- [Be able to fetch current station and weather information]
- [Need to store current information into database]
- [Visualise the status of each station and weather information]
- [The current information should be able to show in 1 second]

User Story: [Visualize trends for station]

- **As a:** [User]
- **I want to:** [see the trends for each station]
- **So that I can:** [make informed decisions]

Priority: [High]
Estimated Points: [15]
Acceptance Criteria:

- [visualise daily trends for each station]
- [the chart should be in suitable position]

Development Task: [Modularize the js files]

- **As a:** [Developer]
- **I want to:** [modular the javascript files]
- **So that I can:** [reuse functions and cooperates effectively]

Priority: [High]
Estimated Points: [8]
Acceptance Criteria:

- [modularize javascript files into different modules]
- [modules should be divided by functionalities]
- [modules should be clear and easy to follow]
- [modules should be called in suitable place, so that the website works correctly]

Development Task: [Running on EC2]

- **As a:** [Developer]
- **I want to:** [run the website on the EC2 server]
- **So that I can:** [allow users to access from remote server]

Priority: [High]
Estimated Points: [5]
Acceptance Criteria:

- [The website should function identically on EC2 as it does on local environments]

User Story: [Get User Precise Location]

- **As a:** [User]
- **I want to:** [share my precise location]
- **So that I can:** [get more precise weather details]

Priority: [Low]
Estimated Points: [16]
Acceptance Criteria:

- [The user should get the weather data in the nearest district based on the location he/she shared]

Software Necessity: [Readme Document]

- **As a:** [developer]
- **I want to:** [have a clear and consistent readme document]
- **So that I can:** [have a clear guidance across all team members]

Priority: [High]
Estimated Points: [8]
Acceptance Criteria:

- [The README document is located in the root directory of the project.]
- [The README document is written in clear and concise language.]
- [The README document includes the following sections: project overview, installation instructions, and usage instructions]
- [The README document is consistent in formatting and style throughout.]
- [The README document is up-to-date and reflects the current state of the project.]
- [The README document is easily readable and understandable by all team members.]

User Story: [Different Color For Different Bikes number in a Station]

- **As a:** [User]
- **I want to:** [see different colors for different number of bikes in a station]
- **So that I can:** [better choose whether to go]

Priority: [Medium]
Estimated Points: [13]
Acceptance Criteria:

- [When displaying the station on the map, each station should have a color representing the number of available bikes]
- [A color legend is displayed on the page]
- [When users view the bikes with different colors, they can easily understand the meaning of each color and the corresponding bike numbers]

User Story: [Route For current location to chosen station]

- **As a:** [User]
- **I want to:** [get the route for my current location to chosen station]
- **So that I can:** [organize my trip easily]

Priority: [Medium]
Estimated Points: [15]
Acceptance Criteria:

- [The user can select a station on the map]
- [The web can show the most efficient route from the user's current location to the selected station]
- [The system displays the calculated route on a map, including the starting point (current location), destination (selected station), and any intermediate stops or directions]
- [The system provides additional route details, such as distance, estimated travel time, mode of transportation, and any relevant traffic or road conditions]
- [The system provides a user-friendly interface for selecting the station and viewing the route, including clear and concise instructions, intuitive navigation, and accessible design]

User Story: [Beautiful Web Interface]

- **As a:** [User]
- **I want to:** [see a beautiful clear web page]
- **So that I can:** [interact with the web easily and enjoy the experience]

Priority: [High]
Estimated Points: [12]
Acceptance Criteria:

- [The web page has a clear and consistent layout.]
- [The font is readable and visually appealing.]
- [The color scheme is aesthetically pleasing and accessible (e.g. sufficient contrast for users with visual impairments).]
- [All interactive elements (e.g. buttons, forms, links) are easily identifiable and usable.]
- [The page is optimized for various screen sizes and devices (e.g. desktop, tablet, mobile).]
- [The page loads quickly (e.g. within 3 seconds) and is responsive.]
- [The page is free of clutter and unnecessary elements.]

Software Necessity: [Annotation For Each Function]

- **As a:** [developer]
- **I want to:** [have a clear and consistent annotations for each function in the program]
- **So that I can:** [understand the functionality of each function easily and maintain the program easily]

Priority: [Medium]
Estimated Points: [8]
Acceptance Criteria:

- [Each function has a clear and concise annotation that describes its purpose and behavior.]
- [Annotations are written in a consistent format throughout the program.]
- [Annotations include information about the function's parameters, return values, and any exceptions it may throw.]
- [Annotations are accurate and up-to-date, reflecting the current implementation of the function.]
- [Annotations are written in a way that is easy for other developers to understand.]
- [Annotations include any relevant details about the function's dependencies or interactions with other parts of the program.]

- [Annotations are properly formatted and follow the established coding standards.]

Software Necessity: [ML model]

- **As a:** [Developer]
- **I want to:** [develop a ML model]
- **So that I can:** [predict occupancy based on collected data]

Priority: [High]
Estimated Points: [8]
Acceptance Criteria:

- [predictions are regularly updated as new occupancy data is collected]
- [predictions are reliable and provide actionable information]

User Story: [Weather Alerts]

- **As a:** [User]
- **I want to:** [be warned if there are weather alerts]
- **So that I can:** [prepare before the trip]

Priority: [Medium]
Estimated Points: [10]
Acceptance Criteria:

- [The system checks for weather alerts for the user's current location and destination]
- [The system updates the weather alerts in real-time, so that the user has the most up-to-date information before the trip]
- [The system allows the user to dismiss or acknowledge the weather alerts, so that they are not repeatedly notified of the same alert]
- [The system displays weather alerts in a clear and concise manner, using visual indicators such as icons or colors to draw attention to the alert]

Notes

- [User Stories could be added, modified or frozen based on future development progress]