

---

# **Software Requirements Specification**

**for**

**<COMP30830 Dublin Bike>**

**Version 1.0**

**Prepared by <Group 13>**

**Zhaohuan GUO, Suwen JIN, Peijin JIANG**

**03/2022**

## 1.Introduction

Dublin Bike software is a web-based application that collects and displays real-time information about bike sharing stations in Dublin, Ireland. The software recommends the nearest bike sharing station to the user based on their location and the availability of bikes at each station.

## 2.Purpose

The purpose of this Software Requirements Specification (SRS) document is to provide a detailed description of the requirements for the development of the Dublin Bike software. The SRS document will guide the development team in designing, implementing, and testing the software.

## 3.Scope

The Dublin Bike software will be a web-based application that allows users to view real-time information about bike sharing stations in Dublin, Ireland. The software will use an open source API to collect fixed data and real-time information about the location of each station, the number of available bikes, and weather data. Based on this data, the software will recommend the nearest bike sharing station to the user. The software will be implemented in HTML.

## 4.Overview

The Dublin Bike software will have the following main features:

- ①Real-time information about bike sharing stations: The software will display real-time information about the location of each station, the number of available bikes, and weather data.
- ②Station recommendation: The software will recommend the nearest bike sharing station to the user based on their location and the availability of bikes at each station.

## 5.Functional Requirements

The following are the functional requirements for the Dublin Bike software:

- ①The software shall be able to collect fixed data and real-time information about the location of each bike sharing station, the number of available bikes, and weather data from an open source API.
- ②The software shall be able to recommend the nearest bike sharing station to the user based on their location and the availability of bikes at each station. The software shall display real-time information about the location of each bike sharing station, the number of available bikes, and weather data. The software shall be implemented in HTML.

## 6.Non-functional Requirements

The following are the non-functional requirements for the Dublin Bike software:

- ①Performance: The software shall be able to collect and display real-time information about bike sharing stations in a timely manner.
- ②Usability: The software shall be easy to use and navigate.
- ③Compatibility: The software shall be compatible with all major web browsers.
- ④Security: The software shall protect user privacy and data security.

## 7.User Interface

The user interface of the Dublin Bike software will consist of an HTML page that displays real-time information about bike sharing stations and weather data. The page will also include a map that shows the location of each station and a search bar where the user can input their location.

## 8.Risks and Issues

The following risks and issues are associated with the development of the Dublin Bike software:

①API data source reliability: The Dublin Bike software relies on the reliability of the open source API data source. If the API data source is not available or experiences issues, the software will not be able to retrieve the latest station and weather data.

②Accuracy of user location data: The accuracy of the recommended station depends on the accuracy of the user's location data. If the user provides inaccurate location data, the software will not be able to recommend the correct station.

③Network connectivity issues: The Dublin Bike software requires network connectivity to access the API data source and user devices. If the network connectivity is unstable or interrupted, the software will not be able to retrieve or display station and weather data or recommend a station.

④Security issues: The Dublin Bike software needs to protect user privacy and data security. If the software has security vulnerabilities or is attacked, user privacy and data may be compromised or damaged.

## 9. Implementation Plan:

①Data source selection: Select a reliable open source API data source that provides accurate and up-to-date information about bike sharing stations in Dublin, Ireland.

②API integration: Integrate the selected API into the software to collect and display real-time information about the location of each station, the number of available bikes, and weather data.

③Front-end development: Use Flask to design and construct the front-end part of the software that displays real-time information about bike sharing stations and weather data.

④Station recommendation algorithm: Develop an algorithm that recommends the nearest bike sharing station to the user based on their location and the availability of bikes at each station.

⑤Testing and debugging: Test the software thoroughly to ensure that it meets all the functional and non-functional requirements, and debug any issues that are identified.

⑥Deployment: Deploy the software on a server and make it available to the public.