POLITECNICO
MILANO 1863

Software Engineering Project

Lorenzo Amici, Marina Ranghetti,
Marta Rossi, Yinyao Zhang

Design Document

Deliverable: DD

Title: Design Document

Authors: Lorenzo Amici, Marina Ranghetti, Marta Rossi, Yinyao Zhang

Version: 1.0

Date: 26 May 2019

Download page: https://github.com/MarinaZen/SoftwareEng.git

Content

1 Introduction

1.1 Purpose

The purpose of the design document is to guide the development and provide high level information on the structure of the software. This document described a set of design characteristic required for the implementation by introducing constraints. In the same time, this document realizes the design and analysis of the overall structure of shared bicycle. System structure and system function are the core parts of system design. The structure of the whole system can be clearly seen through the functional structure diagram of system management module. Moreover, it presents in detail the implementation plan and integration plan.
In details this document contains the list of:

- The structure of database
- The structure of database and software(purpose、internal structure、interact)
- Interfaces provided by the components

1.2 Scope

The project MI-Bike is a web-app based on managing bike-sharing's service information. The goal of this software is to make the shared bicycle management system management electronic, systematic and simple. Moreover, the software aims to service also many categories of customers of MI-Bike. In order to save unnecessary waste of resources in bike management. The end users of the management system are end users, managers and other related personnel. This system includes the general functions of Bike-sharing Run Management.

2 Structure Design

2.1 Overview

Hardware structure is distributed into 3 logic layers which based on python:
● Flask: A lightweight Web application framework written in Python. This is the main visualization function.
● PostgreSQL: here information is stored and retrieved from database of the system, and server updates the position into the map that are published during the run.

2.2 Database Design

The design of database largely determines the usability and robustness of the system. This system uses PostgreSQL database, a table corresponds to a functional module, if other modules are connected, foreign key association is used.

2.2.1 Conceptual structure design of database

The design of database is to change every requirement module in system design from abstract concept to data entity. There are several entities in the campus bicycle as follows:.

- Users (user ID, login password,)
- Station information (corresponding location, corresponding number of bicycles)

2.2.2 Design of logical structure of database

Table 2-1: Data Table List

Title	Main attributes
Users	Users Information (ID, login password)
Station	Station Information (Location, number of bikes)

2.2.3 Detailed design of data table

User table: The table stores users information. The structure of the table user is shown in Table 2-3.

Table 2-3: User Information List

Field	Type of field	Definition of null	Describe
		Not Null	name
		Null	sex

Station table: This table stores station information, and the structure table of station is shown in Table 2-4

Table 2-4: station Information List

Field	Type of field	Definition of null	Describe

2.3 System Management Module

The function of the system includes several functional modules, each of which has several sub-functions. The system management module mainly includes common user management, customer information management, GPS and map service. Specific module function design is as follows.

2.3.1 System Management Module

The system management module is mainly used for users. It can carry out the functions of system users, bicycle information, station information and information statistics system. The structure of the management module is shown in Figure 3-1.

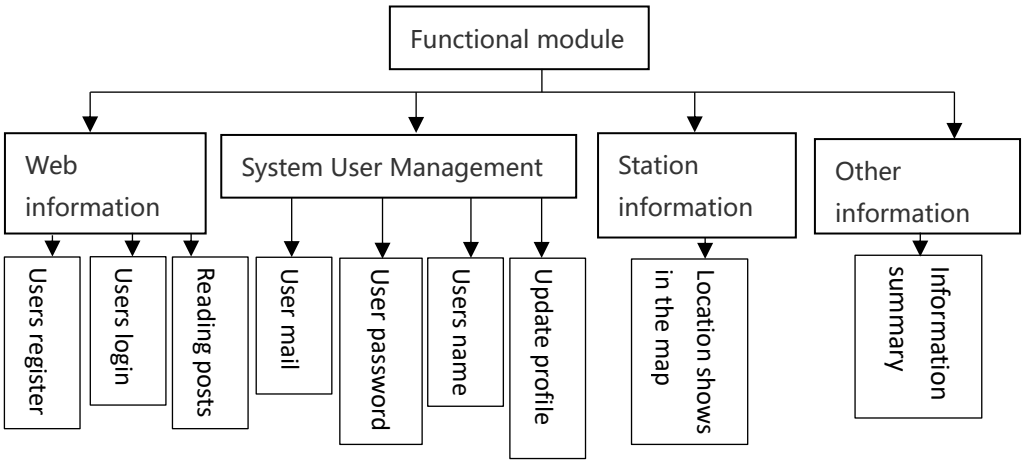

Figure 3-1 Functional Structure Diagram of System Management Module

The specific functions of the system management module are as follows:

● Web Information:

The system shall allow multiple users to access a same service at the same time, allow users to access the basic services without any registration needed, allow registered users to enter the privileged area, allow registered user to see statistics about bike movements.

● Users Information:

Users who use for the first time in the website need to register their personal information, name, email and other information to be clearly filled out, and ask account from Comune di Milano. And also, can update profile.

● Statiom Information:

Store information about the site, including the number and location of bikes, and display it on the map.

● Information Statistics Function. This module is used to count thetotal vehicle, in-storage vehicle and other data.

2.4 Requirements of function

2.4.1 Time requirement

Because this project is aimed at the city shared bicycle, the usage frequency is higher and the usability requirement is higher. It is required to have more reliable security performance. In short, stable, safe, convenient, easy to manage and operate.

● The system should be available 24h/7

● The system should be updated in real time
● The response time: the import and export of data should be completed within 1 to 2 seconds and within acceptable time.

2.4.2 Flexibility

This system has the independent server system and the database system. In order to meet the data requirements of internal and external institutions, and to exchange data with related systems of Comune di Milano, this system specially designed an extended interface for data exchange with these systems. This system uses the website standard interface, itself has the operation nimble characteristic.
● Supportability on mobile phones, which make user convenient operation and management.

2.4.3 Data management capability requirements

Data management is divided into insert, update, and delete. Users' posts information issued by the increase, modification, deletion. And the increase, modify, delete and audit control of Vehicle information.User access to information published by the increase, modification, deletion and audit control.

2.4.4 Fault handling requirements

No errors should be made during normal use, and if unrecoverable system errors are encountered during operation, the database must also be intact.

2.4.5 Operating environment requirements

bcrypt==3.1.4

Flask==1.0

Flask-Bcrypt==0.7.1

Flask-Login==0.4.1

Flask-Mail==0.9.1

Flask-SQLAlchemy==2.3.2

Flask-WTF==0.14.2

Pillow==5.3.0

SQLAlchemy==1.2.7

WTForms==2.1

Users can access the MI-Bike through the browser or mobile phone, and can add, delete and change their own information and attachments within the scope of authority.

2.4.6 Comment

The user report module or some useful module can be added at a later stage.

3 System Testing

4 Effort Spent