 

**Title: SRS for Eco-B Application.**

Author: Rohit Patil

Sayali Bhavsar

**Introduction**

This document is the SRS report for a Eco-B Application. This system has the following three main components:

* We can call a bicycle from another location if not available nearby.
* Select between varieties of locations.
* Take ECO-B for specific ride as per user’s convenience

**Purpose:**

This Eco-B Point System is developed to provide the following services:

Book Ride:

User selects the starting and destination location after selecting bicycle is allotted to the user. After ending ride release bicycle from user and the bicycle with another location.

Report Broken Bicycle:

If there’s any broken bicycle allotted to a user then there must be a provision to user to report it. Admin can inspect bicycles and make them available to serve users after repairing is done.

**Scope:**

This project is design for Book the bicycle ride. The area covers include:

* College Campus: This application is used in college campus
* Spring Boot Technology used for the development of the application.
* Student, all college staffs will be able to use the system effectively.
* Web-platform means that the system will be available for access 24/7 except when there is a temporary server issue which is expected to be minimal.

**Overview:**

Eco-B Point System is used for the book the bicycle ride for the college campus. In college one building to other building distance is long, walking is not possible any time that’s the reason for designing this application for student, staff, and all others members in college. All buildings locations are available in this system, user can go one building to other building by using the bicycle and bicycle book by using this application.

**Functional Requirements :**

* **Register**

1. New users can register through the register option.
2. Required details are entered through a form(after verification) and saved in the database.(Users Table)

* **Login**

1. Previous users can login through mobile number and password. After verifying from DB(Users table).
2. If a mobile number is not found in DataBase then send user to the front page.

* **Book** a ride(After successful login)

1. Give starting point
2. Enter destination
3. Allot a bicycle(From DB of all bicycles filter which are available at that particular point/stop)
4. Release bicycle from user and set that bicycle with another point/stop.

* Get **list** of available **point/stop** to travel.

1. While a user chooses to go for a ride, his boarding point is chosen from his location and closest point/stop to him.
2. Now for destination except his standing point/stop other locations are shown as destination to him.
3. User can choose any of destinations and start his ride with a system allocated bicycle.

* **Track** and **predict** time for travel.

1. Users must get estimated time for his ride.
2. We can make this possible with Google maps API.

* Report **broken** bicycle.

1. If there’s any broken bicycle allotted to a user then there must be a provision to user to report it.
2. If user reports any bicycle then that bicycle is temporarily removed from bicycles database and shifted to the maintenance database(Maintenance database/table).
3. Admin can inspect bicycles and make them available to serve users after repairing is done.

* Get a bicycle available if **not available** to the **nearby station**.

1. If user wants a bicycle for ride but there’s no bicycle available at the closest point/stop then it’s responsibility of system to prompt user if he wants to continue his ride.
2. If user wants to continue his ride then system should automatically select a bicycle from next closest point/stop. And deliver it to that user’s point/stop.(bicycle is shifted from one point to another point)
3. After this all ride starts for user

* Manage bicycles as per **rush hours**.(OPTIONAL)

1. Find on which times at which point/stop requirement of bicycle increases.
2. Now find nearest couple of points/stops to that point and send bicycles automatically to that point/stop where rush is expected.

* Save **user data**(getting reports,etc) - specifically for casual rides.(OPTIONAL)

1. Save user data for any/every ride
2. Data like distance travelled, total ride time.

* Save **previous** rides.(OPTIONAL)

1. Save previous rides(recent 2/5/10) for every user.

* Receive **personalised ride recommendations** according to users' frequent rides.(OPTIONAL)

1. On the basis of previous rides, give ride recommendations to the user.
2. Like if a particular user takes ride at 8:00AM everyday, then ask him to book ride on 7:50AM.

**Hardware Interface:**

Since the application is internet based, all the hardware shall require to connect internet.

**Software Interface:**

1.As the system is on server, so requires any scripting language thymeleaf, Spring boot, etc.

2.System will require capable Database like MySql etc.

3.System will communicate with database for storing user details, admin details, and all ride requests, etc.

**Non-Functional Requirements**

* Security:

In Online mode security is a core issue that must be considered. Viruses and hacking are threatening e-commerce. to avoid hackers and viruses attack we need to provide a security solution.

Security can be provided by encryption,firewall and secure Socket layers.

To prevent unregistered users to place orders in system, the use of session object can be made for verification.

Only system admins can change or update the data in the application and no access will be given to unathourized person.

* Usability:

System will be efficient to use, easy to use.

Information about best seller products will be given to customers. System will provide correct messages so that user can use system easily.

* Reliability:

Backup is useful in recovering your data in the case of an electronic disaster like hardware failure or server crashes. System will maintain a failover server with the replica of all data,so that data can be restore at disaster situation.

* Extensiblity:

System can easily extend the functionality can be flexibly added functional modules.

* Availability:

The System will be available all the time except at downtime. Downtime will be on every sunday at 1:00 am to 1:30 am.

Alternate page should be displayed to user at time of downtime.

* Maintainability:

System will not be shut down for more than once in 24 hours of period for maintainence.

* Portability:

This system can be used on Windows or Linux OS. System is easy to port on any other server.

* Performance:

The online shopping system has adequate performance requirements. it does not take more than 10 seconds to load new pages.

* Integrity:

Whenever the changes are made related to customer data and his orders, it shall be reflected in database as well.

**System Interactions:**

* User Interaction:

User will book the ride through the website. System will allow the user to book the ride if he/she is registered user. Firstly user will register and login then user will book the specific ride.

* Admin Interaction:

Admin will be able to add or remove the cycles. User will send the request and admin will accepts the request then admin allows to user for take a ride.