**User Requirements Document**

**Bike Rental Web Application**

### 1. Overview

This document defines the user requirements for the bike rental web application. The platform aims to deliver a seamless and efficient bike rental experience in Dublin, catering to various user groups, including casual riders, subscribers, business owners, and maintenance staff.

### 2. Goals of the Project

The primary goal of this project is to create a user-friendly and reliable bike rental system that enhances urban mobility while ensuring operational efficiency and scalability. The system should:

* Provide an intuitive interface for users to rent and return bikes with minimal effort.
* Offer secure and flexible payment options.
* Support different user roles, ensuring personalized experiences for casual riders, subscribers, business administrators, and maintenance staff.
* Enable business owners to manage rentals, monitor bike usage, and optimize fleet distribution.
* Integrate real-time tracking and maintenance features to improve service reliability and bike availability.
* Ensure a scalable and extendable architecture that can support future enhancements.

**3. User Roles & Needs**

**3.1 Casual Riders (Tourists & Occasional Users)**

* **Sign up/Login** via email, Google, or social media.
* **View available bikes** on an interactive map.
* **Reserve & unlock** bikes using QR codes or a PIN.
* **Pay easily** via credit card, PayPal, or digital wallets.
* **Track ride details** (distance, cost, duration).
* **Access customer support** for troubleshooting.

**3.2 Subscribers (Regular Users)**

* All features available to casual riders.
* **Subscription plans** for monthly/yearly rentals.
* **Discounted rates** for long-term users.
* **Ride history & usage statistics** dashboard.
* **Auto-renewal & easy cancellation** options.

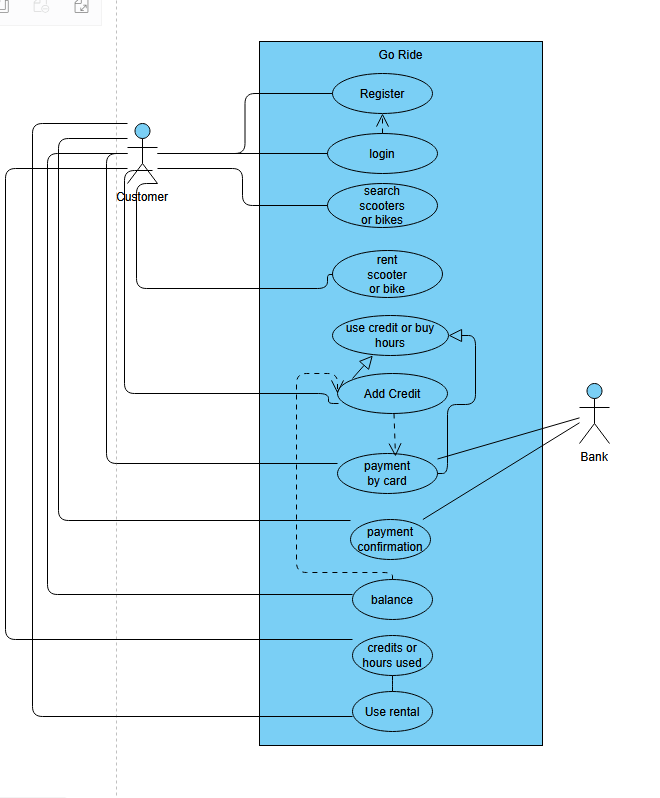
**3.3 Business Owners (Administrators)**

* **Manage bike inventory** (availability, condition, location).
* **Set rental pricing & subscription plans.**
* **Monitor real-time bike usage** with GPS tracking.
* **Analyze user trends** to optimize bike distribution.
* **Handle user support requests** and process refunds.

**3.4 Maintenance Staff**

* **View maintenance alerts** for damaged/missing bikes.
* **Update bike status** (available, under repair, lost).
* **Optimize bike redistribution** to high-demand areas.
* **Access location-based task assignments.**

**3.5** **Use Case Diagram for Customers Using The Website**



**3.6 Use Case Descriptions**

|  |  |
| --- | --- |
| **Use Case Id** | 1 |
| **Use Case Name** | Adding credit |
| **Created by** | Yones Ananzeh |
| **description** | The user will try to add credit to their account the user will have to access the system |
| **primary Actors** | Customer |
| **Secondary Actors** |  |
| **Pre-Conditons** | 1. The Customer must be on the website. 2. The Customer must have an account. 3. The Customer must have a debit or credit card. |
| **Post Conditions** | 1. The Customer recieves the credits. |
| **Main Path** | 1. The Customer accesses the website 2. The Customer logs in to the website 3. The Customer goes to their balance and adds credits 4. They Customer their payment card to pay for the credits 5. The bank confirms the payment 6. The Customer has added credits to their account |
| **Alternate Path** | 1. The Customer accesses the website 2. The Customer has no account 3. The Customer registers 4. The Customer goes to their balance and adds credits 5. They Customer their payment card to pay for the credits 6. The bank denies the payment 7. The Customer has no credits. |

|  |  |
| --- | --- |
| **Use Case Id** | 2 |
| **Use Case Name** | Renting a scoooter |
| **Created by** | Yones Ananzeh |
| **description** | The Customer will try to rent a scooter from the website. |
| **primary Actors** | Customer |
| **Secondary Actors** | Bank |
| **Pre-Conditons** | 1. The cutomer has access to the website 2. The customer must login 3. The customer must have goride credit or a payment card |
| **Post Conditions** | 1. The customer is able to use rental |
| **Main Path** | 1. The Customer accesses the website 2. The Customer logs in to the website 3. The customer browses scooters 4. The customer rents a scooter. 5. The customer selects their scooter to rent. 6. The customer uses credit to rent the scooter. 7. The customer can now use the scooter. |
| **Altenate path** | 1. Steps 1-5 of the main path. 2. The Customer selects how much time they will use the scooter for 3. The customer uses payment card. 4. The bank confirms the payment. 5. The User can now use the scooter. |

**4. Functional Requirements**

**4.1 User Authentication & Profiles**

* Secure **sign-up/login system** with multiple authentication methods.
* **User dashboard** displaying rental history, payments, and active bookings.
* **Profile management** (update payment details, preferences, etc.).

**4.2 Bike Rental & Availability**

* **Live map** displaying available bikes & rental stations.
* **One-click bike reservation & QR-based unlocking.**
* **Flexible rental plans** (pay-per-use, daily, weekly, subscription).
* **Automated ride tracking** with GPS and trip summaries.

**4.3 Payment & Billing System**

* **Multiple payment options** (credit card, PayPal, Apple Pay, Google Pay).
* **Automatic billing** for subscriptions.
* **Transparent pricing** with estimated trip costs.
* **Refund & cancellation policies** for flexibility.

**4.4 Safety & Security Features**

* **GPS tracking** for lost/stolen bike recovery.
* **Emergency button** for quick support in case of issues.
* **Geofencing** to enforce no-parking zones.
* **Bike locking system** to prevent unauthorized use.

**4.5 Admin & Maintenance Dashboard**

* **Real-time fleet management** with bike status updates.
* **Automated maintenance alerts** for damaged bikes.
* **Usage analytics** to track demand & optimize bike placement.
* **User feedback & issue reporting** for service improvements.

**5. Non-Functional Requirements**

**5.1 Performance & Scalability**

* The system must handle **high user traffic** and support future expansion.
* Optimized for **fast loading times** and **real-time updates.**

**5.2 Security & Compliance**

* **End-to-end encryption** for user data & transactions.
* **GDPR compliance** for data protection.
* **Regular security audits** to prevent vulnerabilities.

**5.3 User Experience (UX) & Accessibility**

* **Mobile-first design** for seamless access across devices.
* **Intuitive user interface** for effortless navigation.
* **Accessibility features** to accommodate users with disabilities.

**6. Project Feasibility**

**6.1 Timeline & Development Plan**

* **Three-month development period (Jan 27 – Apr 28)** with a **prototype demo every 3 weeks**.
* **Prioritize core features** (authentication, rentals, payments, mapping) before enhancements.
* **Development milestones:**
  + **Week 3:** Basic booking system.
  + **Week 6:** Payment integration.
  + **Week 9:** Admin dashboard & analytics.

**6.2 Potential Risks & Contingency Plans**

1. **Limited Team Capacity**
   * **Risk:** High workload could lead to burnout.
   * **Solution:** Focus on MVP features & ensure knowledge-sharing within the team
   * .
2. **Technical Challenges**
   * **Risk:** Potential issues with database, APIs, or security.
   * **Solution:** Allocate debugging time & maintain thorough documentation.
3. **Strict Deadlines & Unexpected Delays**
   * **Risk:** Bugs or unforeseen problems affecting timelines.
   * **Solution:** Include buffer time in final weeks to resolve last-minute issues.
4. **User Testing & Feedback Implementation**
   * **Risk:** Major usability issues discovered late in development.
   * **Solution:** Conduct early-stage testing with simulated users.

**7. Future Enhancements**

* **Loyalty rewards** for frequent riders.
* **Referral system** to encourage user growth.
* **Integration with public transport schedules.**
* **E-bike & scooter support** for broader mobility options.
* **AI-driven demand prediction** to optimize bike distribution.