

Smart Bike Sharing and Rental Management Service

Sanjay Gundu¹, Sarveshwaran AP², and Dr.Vivek Jstus³

¹Student, Department of computing technologies, SRM Institution of Science and Technologies, Kattankalathur, chennai, INDIA.

²Student, Department of computing technologies, SRM Institution of Science and Technologies, Kattankalathur, chennai, INDIA

³Assistant professor, Department of computing technologies, SRM Institution of Science and Technologies, Kattankalathur, chennai, INDIA

Abstract—Convenient, adaptable transportation solutions are in greater demand as a result of the sharing economy and the growth of smart cities. Using Flask, HTML, CSS, and JavaScript, this project offers a Smart Bike Sharing and Rental Management System. Through the website, people (providers) can rent out their own bikes to other users or nearby rental shops. It provides two distinct sign-ups: one for users and one for suppliers. While consumers can explore and hire bikes based on their location, providers can display their bikes with information such as model name, registration number, rental fees, and more. By sharing unused resources, this approach not only encourages environmentally friendly commuting but also enables bike owners to make extra money. The application has secure login portals, accessible user interfaces, and basic functions like dynamic bike selection and bike listing management. This system seeks to satisfy the increasing need for short-term, reasonably priced, and environmentally friendly mobility options in cities by providing a smooth renting experience.

Keywords—flask, HTML, CSS, JavaScript

I. INTRODUCTION

Bike-sharing programs are now a common choice in cities all over the world due to the fast increase in urbanization and the demand for environmentally friendly transportation options. Existing platforms, however, frequently depend on big businesses and bicycle fleets, leaving a gap for people who own their own bikes but don't know how to rent them out. By offering a platform that links local personal bike owners (providers) with potential renters (users), the Smart Bike Sharing and Rental Management System fills this gap. The Flask framework was used to create this project, which uses HTML, CSS, and JavaScript for the front end and Python for the back end. The system has two sign-up portals: one for customers who want to rent bikes and another for providers who want to post their bikes for hire. In order to guarantee a clear and easy rental process, providers can provide important information about their bikes, such as the

model name, registration number, daily fees, and uploaded photos. Renters can use the user interface to look for bikes by choosing their city and looking through ads nearby. In addition to providing a more sustainable and localized option to corporate bike-sharing programs, our project enables people to make money off of unused assets and offers a user-friendly, safe platform for both providers and users.

II. EASE OF USE

The rise of shared mobility options, especially bike-sharing programs, has changed the urban transportation landscape globally in recent years. Critical urban problems including air pollution, traffic congestion, and the demand for more environmentally friendly modes of transportation are all intended to be addressed by these services. To improve RFID-enabled bike rental systems, a web services-based framework was suggested. In order to ensure effective bike tracking, inventory management, and safe transactions, their research concentrated on using RFID technology to expedite the rental process. The framework showed promise for increasing the operational effectiveness and scalability of bike-sharing services by combining web services with RFID, establishing a precursor to the technological developments found in today's intelligent bike-sharing systems [1]. applying genetic algorithms to maximize system efficiency and resource utilization. Research demonstrates how algorithmic models can improve the overall performance of stationless bike-sharing services by addressing operational issues like bike redistribution and user demand prediction [2]. A comprehensive method for bringing into practice an Internet of Things network-based smart bicycle sharing system. Their work demonstrates how sensors, GPS units, and cloud-based infrastructure may be integrated to offer real-time bicycle monitoring, improving system efficiency and user experience [3].

Exploring the relationship between environmental sustainability, social media participation, and bike sharing programs. They emphasized the critical role social media platforms play in promoting bike-sharing programs, shaping user behavior, and encouraging environmentally friendly urban mobility strategies [4]. Using machine learning techniques to improve bike-sharing programs' effectiveness. In order to maximize bike availability and maintain demand-supply balance across several locations, their research made use of predictive models. This data-driven strategy demonstrated how machine learning may enhance urban bike-sharing networks' operational management and user experience [5]. Employing a mechanism design approach to offer important insights on how supply and demand are coordinated in mobility service networks. The authors suggest a dynamic pricing and incentive framework to maximize vehicle allocation, highlighting the significance of striking a balance between user demand and available resources in shared mobility networks. This strategy is especially pertinent to smart bike-sharing systems, because user pleasure and system efficiency depend on real-time coordination of bike availability and needs [6]. The sharing economy's perspective on the planning and execution of a campus bike rental program. The solution uses mobile apps and cloud-based platforms to make bike rentals easy while providing location monitoring and real-time availability updates. Their research lays the groundwork for comparable applications in urban contexts by highlighting the value of resource efficiency and user experience optimization in boosting the uptake of bike-sharing programs on campuses [7].

Improvements to mobile applications for car rental services with an emphasis on enhancing operational effectiveness and user experience. Their study emphasizes the value of incorporating real-time data and intuitive user interfaces to streamline the rental process [8]. Data-driven balancing solution intended to improve system maintenance and quality of service (QoS) in bike-sharing operations. The strategy efficiently distributes bikes across stations by utilizing real-time data and predictive analytics, reducing service interruptions and streamlining maintenance plans. This combined emphasis on operational effectiveness and service quality establishes a standard for next smart bike-sharing systems that aim to strike a compromise between customer demand and system sustainability [9]. Introducing a public bicycle-sharing program that is controlled by smartphones, with a

focus on encouraging healthy living through cycling in addition to its environmental advantages. The system provides a more user-centric and sustainable urban transportation option by tracking user activity and optimizing bike distribution through the use of smartphone sensors and cloud-based services [10]. In order to maximize bike availability and rental experiences, smart bike-sharing systems examined user behavior. They improved operational efficiency and service satisfaction for public bicycle rental systems by offering insights into customer preferences and rental trends through the use of real-time data and network-based solutions [11].

III. METHODOLOGY

The Smart Bike Sharing and Rental Management System was developed utilizing an approach that prioritizes user and provider experiences to guarantee usability, security, and functionality. The following crucial steps comprise the methodology:

1. Planning and Requirements Collection:

Determining the needs for the user and provider modules is the first step in the project. While users must browse and rent bikes depending on location, providers must register, list their bikes, and maintain listings. Additionally, the platform must use captcha verification and login authentication to guarantee data protection.

2. Python and Flask back-end development:

The back-end is built with Flask, a lightweight Python web framework. User and provider functionalities have their own routes. Routes handle bike listing, updating, deletion, login, and registration for providers. Routes manage user registration, login, city selection, and bike availability viewing. Bike information, rental transactions, and user and supplier data are stored in a SQLite database or a comparable one.

3. Front-end development using JavaScript, HTML, and CSS:

JavaScript provides dynamic functionality, CSS provides styling, and HTML provides structure for the front-end. Dynamic content presentation is made possible by integrating back-end data into HTML views using Jinja2 templating. To make managing

bike listings and renting bikes easier, the supplier and user sites have different user interfaces.

4. Registration of Users and Providers:

Users and providers each have their own sign-up forms created, where providers submit extra information such as their city, mobile number, and captcha. Encrypted passwords are stored securely. The purpose of captcha is to prevent spam registrations.

5. Management and Listing (Provider Portal):

Providers can add bike listings after logging in by entering information such as the bike model, registration number, daily fees, date of manufacture, and providing a picture of the bike. When necessary, the system enables providers to add or remove listings.

6. User Portal for Bike Search and Rental:

After logging in, users can choose their city, which starts a search for bikes that local providers have advertised. Depending on availability, users can browse the postings, check the bike's specifications, and then hire one.

7. Deployment and Testing:

The system is put through a rigorous testing process to guarantee a seamless user experience and consistent functionality. While integration tests make sure all modules function together, unit tests are carried out for individual components. For accessibility, the platform is subsequently set up on a cloud hosting provider.

This approach guarantees a strong, intuitive bike-sharing system that successfully satisfies the requirements of both bike owners and renters.

IV. IMPLEMENTATION

To provide a smooth user experience for both bike providers and renters, the Smart Bike Sharing and Rental Management System integrates front-end, back-end, and database components. The project is constructed with SQLite for data storage, HTML, CSS, and JavaScript for the front end, and the Flask framework for back-end operations. The steps listed below describe the implementation process in detail:

1. Setup of the Environment:

Setting up the development environment is the first step in the project. The required Python libraries, including Flask-WTF for form handling, Flask-Login

for authentication, and SQLite for the database, are installed with Flask. Dependencies are managed in a virtual environment.

2. Design of the Database:

Bike listings, rental transactions, and user and provider data are all stored in SQLite. Three important tables are made:

users: Holds user information, such as password, email address, and mobile number.

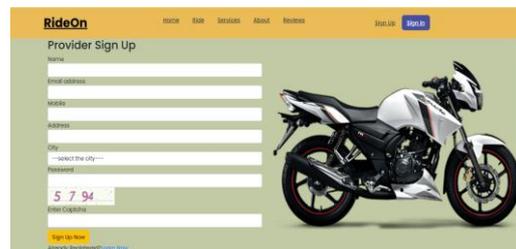
providers: Holds provider information, including name, password, city, email, and cellphone number.

Model name, registration number, daily charge, production date, description, bike image, and provider ID (foreign key) are all stored in the `bikes` section.

3. Authentication of Users and Providers:

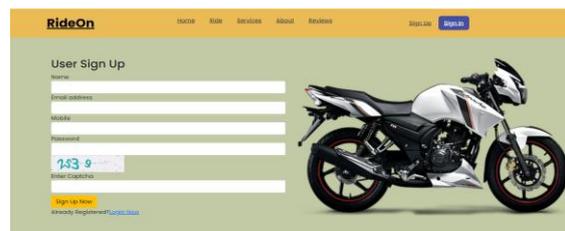
There are two distinct systems in place for users and providers to register and log in.

Provider Registration: Name, email address, city, mobile number, password, and captcha are among the information that providers enter. Following a successful registration process, providers can access the provider dashboard by logging in.



User Registration: Users enter their name, password, email address, mobile number, and captcha. The hashed password is saved in the `users` table when the system verifies the input. Users can access the bike rental portal by logging in after registering.

Session management is handled by Flask-Login, guaranteeing safe login and logout processes for providers and users alike.



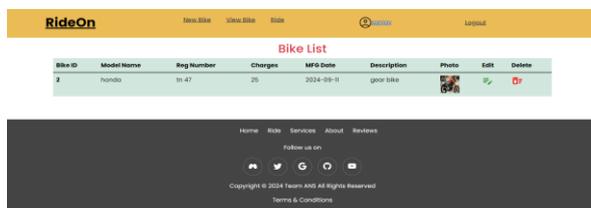
4. Bike Listing and Provider Dashboard:

Providers are taken to their dashboard after logging in, where they can control bike listings. The following features are put into practice:

Include Bike: By providing the bike's model name, registration number, daily charge, manufacturing date, description, and a picture file, providers can add a new bike. The path to the image is kept in the database, and the image itself is saved in a directory.

Edit Bike: A current listing's details can be changed by providers. The current bike information is pre-populated on the form, enabling suppliers to change specifics.

Delete Bike: A bike listing can be deleted by providers, which will take it out of the database and storage.



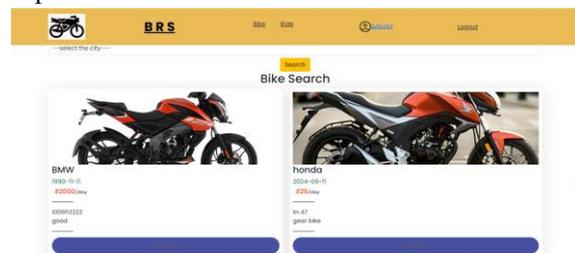
5. Bike Search and User Dashboard:

Users are taken to a dashboard where they can browse bikes according to their location after logging in. The following are included in the implementation:

City Selection: From a drop-down list, users choose their city. After selection, the server receives a Http request to retrieve the bikes that are accessible in that city.

Bike Listings: Bikes that fit the chosen city are shown in an easy-to-use format with pictures, descriptions, and rental information.

Hire Bike: By choosing a bike from the list, users can hire one. Payment gateways and email booking confirmations are examples of potential future improvements.



6. Integration of Captcha:

To stop spam and bot registrations, captcha is incorporated into the user and provider registration forms. It makes use of its reCAPTCHA service, which

offers a straightforward but efficient security precaution.

7. Development from the Front End:

JavaScript is used for interactivity, CSS is used for styling, and HTML is used for structure in front-end design. Data on the sites is dynamically updated using jinja templating. In order to make the website accessible on a variety of devices, responsive design principles were used in the creation of the provider dashboard, bike listing forms, and user bike search interface.

8. Image Upload and File Management:

Providers can attach pictures of their bikes to the bike listing form. File uploads are handled safely via Flask's `request.files` module. The paths of uploaded photographs are saved in the database, and they are kept in a special directory. The routes retrieved from the database are used to display the photos on the front-end.

9. Debugging and Testing:

Following the deployment of essential features, the system is put through an extensive testing procedure to make sure everything functions as it should. Each module, such as the bike listing, login, registration, and search features, has its own set of unit tests. Flask's built-in debugger is utilized for debugging, and Tester and other tools are used to test Http routes.

IV. RESULT

The Smart Bike Sharing and Rental Management Services were successfully tested and set up for operation, offering a simplified platform for users to locate and hire bikes in their area as well as for bike owners to rent out their bikes. The platform successfully supported the listing, administration, and rental of bikes while meeting its main goals of providing a safe, user-friendly experience with distinct portals for providers and users. The project's main outcomes include:

1. User satisfaction and operational efficiency:

Bicycle listings may be easily added and managed by providers, who could also update and remove listings as necessary. According to user feedback during testing, the easy search and rental process—made

possible by city-based bike filtering—offered a convenient and effective experience.

2. Protection of Spam and Data Security:

With password hashing and captcha verification, the dual registration system reduced the possibility of spam registrations and offered a high degree of data protection. Both users and suppliers were guaranteed safe and private access to the platform through the use of Flask-Login and hashed passwords.

3. Using Real-time Data Display to Provide a Smooth Navigation Experience:

The site gave customers the most recent possibilities by constantly displaying bike listings based on real-time inputs, like the city they choose, using Jinja2 templating. This method reduced loading times and enhanced the system's overall usability.

4. Sustainability and User Engagement:

The website encouraged sustainable transportation options and a sense of community by enabling people to rent their own bikes. While users valued the accessibility and cost of localized bike rentals, providers saw the platform as a useful means of making money off of unused assets. Overall, the project effectively met the needs of both bike providers and renters in a single, user-friendly platform, showcasing a workable, user-centric solution for bike sharing and rental.

REFERENCES

- [1] Kuo-shien Huang and Shun-ming Tang, "A web services based framework for RFID bike rental business solutions," 2008 IEEE International Conference on Industrial Engineering and Engineering Management, Singapore, 2008, pp. 1475-1479.
- [2] H. Li, C. Wu, Z. Zhang and G. Yang, "Analysis on Operating Model of Station-Less Shared Traffic System Based on Genetic Algorithm," 2016 13th International Conference on Embedded Software and Systems (ICESSE), Chengdu, China, 2016, pp. 114-118.
- [3] J. L. M. Puyol and V. M. Baeza, "Bicycle Sharing System Using an IoT Network," 2021 Global Congress on Electrical Engineering (GC-ElecEng), Valencia, Spain, 2021, pp. 131-135.
- [4] O. B. Piramuthu and W. Zhou, "Bicycle Sharing, Social Media, and Environmental Sustainability," 2016 49th Hawaii International Conference on System Sciences (HICSS), Koloa, HI, USA, 2016, pp. 2078-2083.
- [5] J. Qian, M. Comin and L. Pianura, "Data-Driven Smart Bike-Sharing System by Implementing Machine Learning Algorithms," 2018 Sixth International Conference on Enterprise Systems (ES), Limassol, Cyprus, 2018, pp. 50-55.
- [6] C. Lisson, N. Roedder, P. Stroehle and C. Weinhardt, "Decisions in Mobility Service Networks -- Coordinating Demand and Supply Using a Mechanism Design Approach," 2016 49th Hawaii International Conference on System Sciences (HICSS), Koloa, HI, USA, 2016, pp. 1606-1613.
- [7] Y. Lin, J. Xu, J. He, X. Meng, Y. Wang and S. Wang, "Design and implementation of campus bike rental system from the perspective of sharing economy," 2022 International Conference on Computers, Information Processing and Advanced Education (CIPAE), Ottawa, ON, Canada, 2022, pp. 131-135.
- [8] F. Y. H. Ahmed, E. B. Hazlan and M. I. Abdulla, "Enhancement of Mobile-Based Application for Vehicle Rental," 2021 IEEE 11th IEEE Symposium on Computer Applications & Industrial Electronics (ISCAIE), Penang, Malaysia, 2021, pp. 163-168.
- [9] S. Wang, X. Zhu, G. Wang, Y. Liu, T. He and D. Zhang, "eShare+: A Data-Driven Balancing Mechanism for Bike Sharing Systems Considering Both Quality of Service and Maintenance," in IEEE Transactions on Knowledge and Data Engineering, vol. 35, no. 10, pp. 10497-10513.
- [10] S. Wang, X. Zhu, G. Wang, Y. Liu, T. He and D. Zhang, "Y. Zhao, L. Chen, C. Teng, S. Li and G. Pan, "GreenBicycling: A Smartphone-Based Public Bicycle Sharing System for Healthy Life," 2013 IEEE International Conference on Green Computing and Communications and IEEE Internet of Things and IEEE Cyber, Physical and Social Computing, Beijing, China, 2013, pp. 1335-1340.

- [11] H. -E. Chueh, C. -Y. Kao and M. -L. Wu, "Usage Behavior Analysis of Intelligent Network based Public Bicycle Rental System," 2020 International Computer Symposium (ICS), Tainan, Taiwan, 2020, pp. 300-303.