REDBACK OPERATIONS SMARTBIKE MOBILE APP

Table of Contents

Introduction

Welcome to the SMARTBIKE mobile app, your ultimate companion for tracking and enhancing your biking adventures. Whether you're a casual rider exploring new routes or a competitive cyclist aiming for personal bests, SMARTBIKE is designed to support all your biking needs with its robust and user-friendly features.

About SMARTBIKE

SMARTBIKE is more than just a tracking tool; it's a comprehensive platform that integrates advanced performance monitoring, social engagement, and personalized insights to help you make the most out of every ride. Developed by the Redback Operations Web and Mobile Development Team, SMARTBIKE combines cutting-edge technology with user-centric design to deliver a seamless biking experience.

Key Features

- **Activity Tracking:** Monitor your biking sessions with detailed metrics such as speed, RPM, power, distance, heart rate, and calories burned.
- **Performance Analysis:** Gain insights into your performance with daily, weekly, and monthly statistics, showcasing trends and improvements.
- **Social Engagement:** Connect with friends, participate in challenges, and compete on leaderboards to stay motivated and engaged.
- **Customization:** Tailor your experience with customizable profiles, themes, and goal settings.
- **Health Integration:** Sync your biking data with popular health platforms like Apple HealthKit and Google Fit for a holistic view of your fitness journey.
- **Advanced Security:** Protect your data with features like Multi-Factor Authentication (MFA) and secure communication protocols.

Purpose of This Document

This README document is your comprehensive guide to using SMARTBIKE. It provides detailed instructions on installing the app, setting up your profile, and making the most of its features. Whether you are a new user or an experienced cyclist, this guide will help you navigate the app with ease.

Who Should Use This Guide

- **New Users:** Get started with SMARTBIKE by following the installation and setup instructions.
- **Experienced Users:** Discover advanced features and troubleshooting tips to enhance your biking experience.
- **Developers:** Contribute to the app's development by following the guidelines provided for setup and version control.

Features

SMARTBIKE offers a robust suite of features designed to cater to both casual bikers and serious cyclists, enhancing your biking experience through advanced tracking, insightful analytics, and vibrant community engagement. Below are the key features that set SMARTBIKE apart:

Activity Tracking

- **Real-Time Monitoring**: Keep a close eye on your biking sessions with real-time tracking of essential metrics such as speed, RPM, power output, distance traveled, heart rate, and calories burned. The intuitive interface ensures that you have immediate access to your performance data at all times.
- **Detailed Session Summaries**: Post-session, receive comprehensive summaries that provide an in-depth analysis of your performance, including total time spent, average speed, peak performance metrics, and more. These summaries help you understand your progress and areas for improvement.

Performance Analysis

- **Comprehensive Reports**: Access detailed reports on your performance over various timeframes—daily, weekly, and monthly. These reports aggregate your data to show trends and improvements, offering a clear picture of your fitness journey.
- **Trend Analysis and Insights**: Utilize trend analysis tools to identify patterns in your biking habits. Understand which times of the day or days of the week you perform best, helping you optimize your training schedule.
- **Visual Data Representation**: Enjoy sophisticated data visualization with charts, graphs, and other visual aids that make it easier to interpret your performance data. These visual tools highlight key statistics and trends at a glance.

Social Engagement

- **Friends and Community Interaction**: Build a network of friends within the app, enabling you to share your achievements, participate in group activities, and encourage each other. The community features foster a sense of camaraderie and mutual motivation.
- **Challenges and Competitions**: Engage in various challenges designed to push your limits and keep you motivated. Compete with friends and other users in time-based or distance-based challenges, and strive to top the leaderboards.
- **Leaderboards and Rankings**: Compare your performance against friends and the broader SMARTBIKE community. Leaderboards offer a competitive edge, motivating you to improve your rankings and achieve new milestones.

Customization

- **Profile Management**: Personalize your profile by adding personal details, uploading a profile picture, and setting up a unique user ID. Secure your changes with password confirmation or Multi-Factor Authentication (MFA) to ensure your data remains protected.
- **Goal Setting and Tracking**: Set and track personal biking goals, such as specific distances or time targets. The app provides reminders and progress updates, helping you stay focused and motivated to achieve your fitness objectives.
- **Theme and Interface Customization**: Tailor the app's appearance to your preferences with customizable themes and interface settings. This feature ensures that your interaction with the app is both enjoyable and visually appealing.

Health Integration

- **Seamless Health Data Sync**: Integrate your biking data with popular health and fitness platforms like Apple HealthKit and Google Fit. This seamless synchronization allows you to have a comprehensive view of your overall fitness activities, combining biking data with other health metrics.
- **Holistic Fitness Tracking**: Leverage the integration with health platforms to track a wide array of fitness activities. This holistic approach provides a more complete picture of your fitness journey, aiding in better health management and goal setting.

Advanced Security

- **Enhanced Account Protection**: Secure your account with Multi-Factor Authentication (MFA), adding an extra layer of security to your personal data. MFA ensures that even if your password is compromised, your account remains protected.

- **Secure Data Communication**: Ensure that all data transmitted between your app and the backend servers is encrypted. This protects your personal information and performance data from potential breaches during transmission.

Additional Features

- **Diverse Workout Options**: Choose from a variety of predefined workout types or create custom workouts tailored to your fitness level and goals. This flexibility ensures that your training sessions remain effective and enjoyable.
- **Unique Session IDs and Activity Logs**: Each biking session is assigned a unique ID for easy tracking and reference. Detailed activity logs help you review past sessions, analyze performance, and plan future workouts.
- **User Feedback and Support**: Provide feedback and seek support through in-app forms or direct contact options. Continuous feedback helps improve the app, ensuring it meets your needs and expectations.
- **Localization and Internationalization**: Access the app in multiple languages, making it user-friendly for a global audience. Content localization ensures that users from different regions can use the app comfortably.
- **Social Sharing**: Share your biking achievements and performance stats on social media platforms. This feature not only helps you celebrate your milestones but also inspires others in your network to engage in fitness activities.

SMARTBIKE is meticulously designed to cater to all aspects of your biking experience, from detailed activity tracking and performance analysis to robust social engagement and personalized customization. Explore these features to maximize your biking potential and stay motivated on your fitness journey!

Installation

Setting up the SMARTBIKE mobile app involves several steps to ensure both the front-end and back-end environments are correctly configured. Follow these detailed instructions for a smooth installation process.

Prerequisites

Before you begin, ensure you have the following prerequisites installed on your system:

- **Operating System**: Windows 10 or later, macOS 10.15 or later, or a recent version of Linux.
- **Git**: Version control system for cloning repositories.
- **Python 3.8+**: For back-end development.

- **Android Studio**: IDE for Android app development.
- **Flutter SDK**: For building the mobile app.
- **Visual Studio Code**: Recommended code editor.

Step-by-Step Installation

1. Install Git

Visit the Git website and download the latest version for your operating system. Follow the installation instructions provided on the site.

3. Install Python

Go to the Python website and download Python 3.8 or later. Ensure you add Python to your system PATH during installation.

5. Install Android Studio

1. Visit the Android Studio download page.
2. Download the installer for your operating system.
3. Run the installer and follow the setup wizard to install Android Studio and necessary SDK components.
4. Launch Android Studio and complete the initial setup wizard, including installing additional SDKs and tools.

6. Install Flutter SDK

1. Go to the Flutter installation page.
2. Download the Flutter SDK for your operating system.
3. Extract the downloaded file to a suitable location on your system.
4. Add the Flutter binary to your system PATH.
5. Verify the installation by running *flutter --version* in your terminal or command prompt.

7. Clone the Front-End Repository

1. Open Android Studio.
2. Navigate to *File > New > Project from Version Control.*
3. Enter the repository URL: https://github.com/alexbaar/app.git
4. Choose a local directory to clone the repository.
5. Click Clone to download the project files.

8. Set Up the Front-End Environment

1. Create a .env file in the root directory of the cloned front-end project.
2. Add the following configuration to the .env file:
   API_URL_BASE=http://<your_machine_network_address>:8000
3. Replace <your_machine_network_address> with your machine's network address (e.g., http://192.168.1.100:8000).
4. Open pubspec.yaml in the root project directory.
5. Click the *Pub Get* button to install project dependencies.

9. Clone the Back-End Repository

Option 1: Using Terminal

1. Open a terminal on your machine.
2. Type: *git clone* https://github.com/alexbaar/app_backend.git
3. Navigate to the cloned repository: *cd app_backend*
4. Open the project in Visual Studio Code: *code .*

Option 2: Using Visual Studio Code

1. Open Visual Studio Code (with no project open).
2. Click on the third icon on the vertical panel on the left (Source Control).
3. Click on *Clone Repository.*
4. Paste the repository URL: https://github.com/alexbaar/app_backend.git
5. The project will open in Visual Studio Code.

10. Set Up the Back-End Environment

1. Open a terminal in Visual Studio Code (*View > Terminal*).
2. Navigate to the backend server directory: *cd backend_server*
3. Verify you are in the correct directory by listing the files: *ls*
4. You should see a list of files such as *__init__.py, admin.py*, and others.
5. Create a virtual environment: *python3 -m venv .venv*
6. If you are using python instead of python3, type: *python -m venv .venv*
7. Activate the virtual environment:
   - On Windows: *.venv\Scripts\activate*
   - On macOS/Linux: source *.venv/bin/activate*
8. Install the required dependencies: *pip install Django djangorestframework*

11. Run Database Migrations

1. Navigate one level up to the project root directory: *cd ..*
2. Verify the files in the directory: *ls*
3. You should see README.md, backend_server, and other files.
4. Run the following commands to make and apply migrations:
   ```
   python3 manage.py makemigrations
   python3 manage.py migrate
   ```
5. If you are using python instead of python3, type:
   ```
   python manage.py makemigrations
   python manage.py migrate
   ```

12. Start the Back-End Server

1. In the terminal, run the following command to start the server:
   ```
   python3 manage.py runserver 0.0.0.0:8000
   ```
2. If you are using python instead of python3, type:
   ```
   python manage.py runserver 0.0.0.0:8000
   ```

13. Run the Front-End App

1. Connect a physical Android device via USB or set up an Android emulator in Android Studio.
2. Open Android Studio and navigate to the project directory.
3. Select the run configuration for the app.
4. Click the "*Run*" button to start the app.

Additional Configuration

Environment Variables

1. Install python-dotenv to manage environment variables:
   ```
   pip install python-dotenv
   ```
2. At the project root, add a new file called .env.
3. Inside the .env file, specify the following values:
   ```
   SECRET_KEY=<your_secret_key>
   EMAIL_BACKEND=<your_email_backend>
   DEFAULT_FROM_EMAIL=<your_default_from_email>
   ```

Common Issues:

- **Pillow Installation Error**: If you encounter an error related to the ImageField in Django, it likely means the Pillow library is not installed. Resolve this by running: python -m pip install Pillow or python3 -m pip install Pillow

Usage

Once you have successfully installed and set up the SMARTBIKE mobile app, you can start exploring its features and functionalities. This section provides a detailed guide on how to use the app effectively, from logging in to leveraging advanced analytics and engaging with the community.

1. Initial Setup

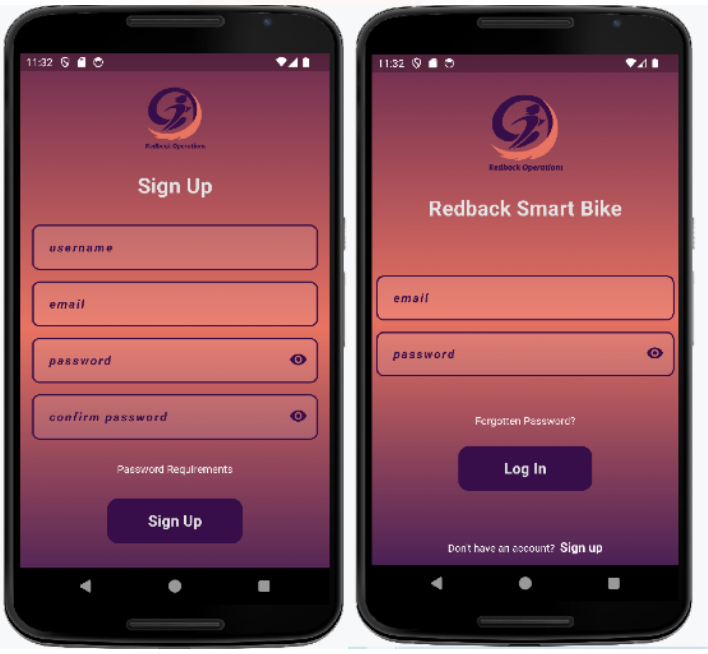
a. Launch the App

1. Open the SMARTBIKE app on your Android device or emulator.
2. You will be greeted with a welcome screen.

b. Registration and Login

2. Register:

- Tap on the "Sign Up" button.
- Fill in the required information such as username, email, password, and confirm password.
- Tap "Sign Up" to create your account.

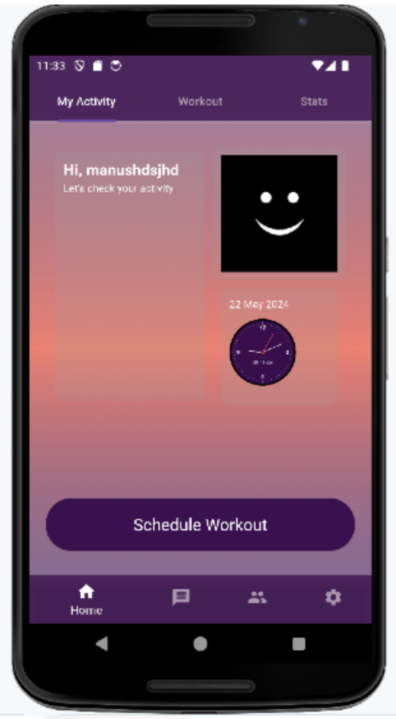


3. Login:

- Tap on the "Log In" button.
- Enter your registered email and password.
- Tap "Log In" to access your account.

4. Home Screen

Upon successful login, you will be directed to the Home Screen, which serves as the central hub for navigating the app's features.



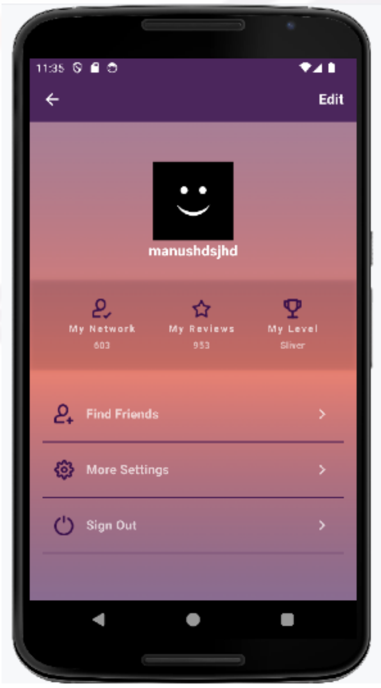
Navigation Bar

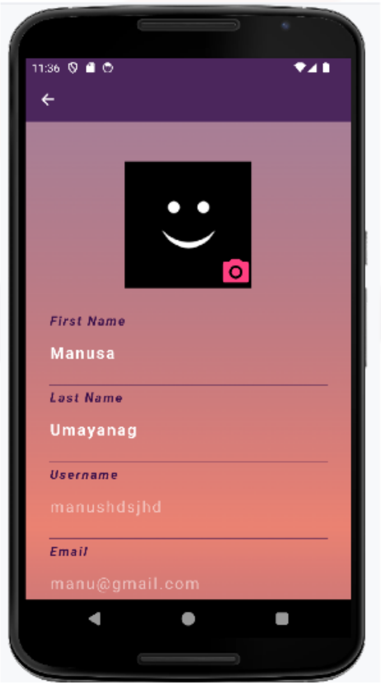
- **Home**: Returns you to the home screen from any section.
- **My Activity**: Access your past workouts and detailed statistics.
- **Workout**: Start new workouts and choose different workout types.
- **Stats**: View your performance trends and analysis.
- **Settings**: Access and modify app settings.

5. Profile Management

a. Editing Your Profile

1. Navigate to the "Profile" section.
2. Tap "Edit Profile".
3. Update your personal details such as name, profile picture, and bio.





First Name
Manusa

Last Name
Umayanag

Username
manushdsjhd

Email
manu@gmail.com

b. Viewing User Details

1. Navigate to the “Profile” section.
2. Tap on your profile picture to view your details.

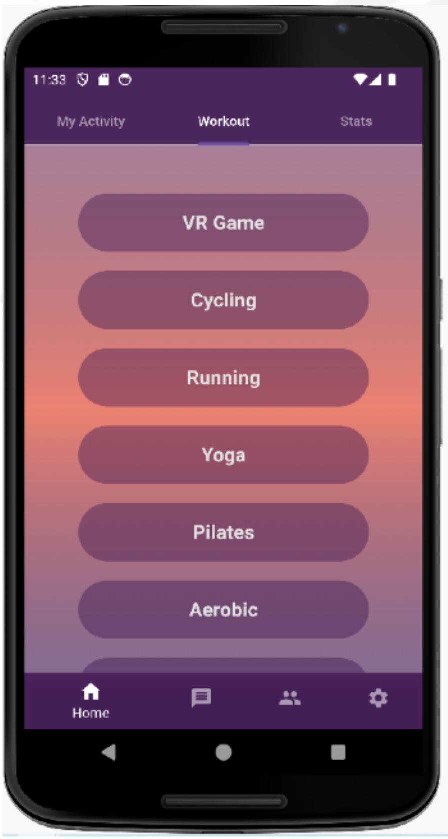
6. Activity Tracking

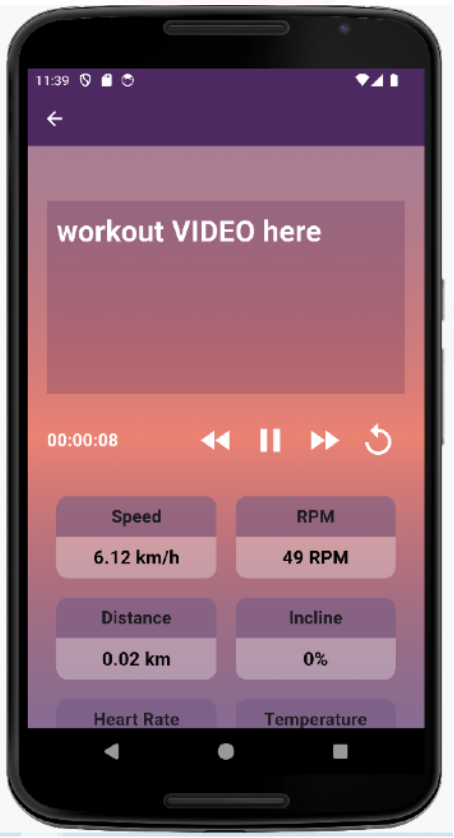
a. Starting a Workout

1. Navigate to the “Workout” section.
2. Choose a workout type (e.g., VR Game, Cycling, Running, Yoga, Pilates, Aerobic).
3. Set your goals and preferences (e.g., solo or group session).
4. Tap “Start” to begin tracking your workout in real-time.

b. Monitoring Your Session

1. During your workout, monitor key metrics such as speed, RPM, power, distance, heart rate, and calories burned.
2. The real-time dashboard displays these metrics, allowing you to adjust your performance on the go.

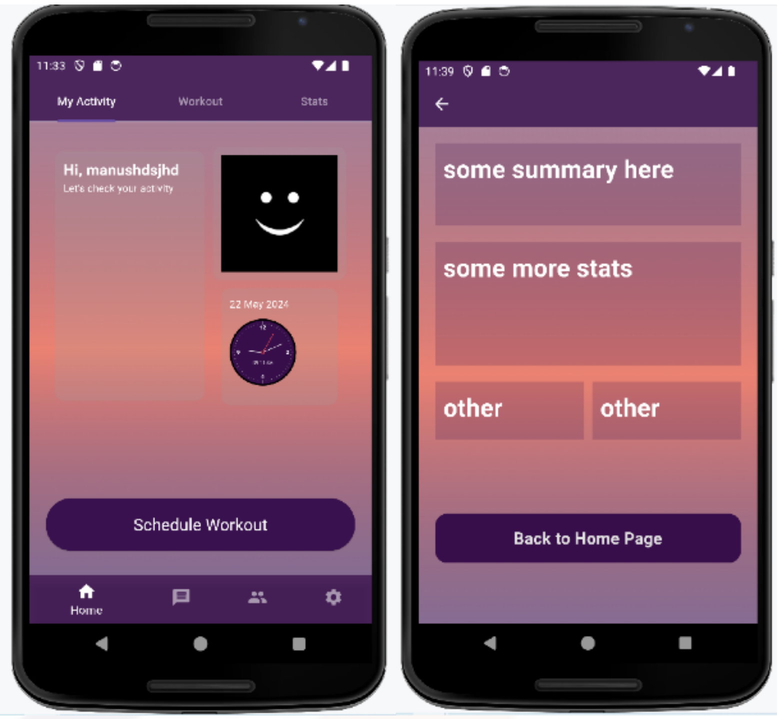




7. My Activity

a. Viewing Past Workouts

1. Navigate to the “My Activity” section.
2. Select a date to view detailed information about your past workouts.
3. For each session, view metrics such as average speed, RPM, power, distance, heart rate, and calories burned.



b. Generating Summaries

1. Choose a date range to generate a summary of your workouts.
2. The summary includes averages and totals for key metrics, helping you analyze your performance over time.

8. My Stats

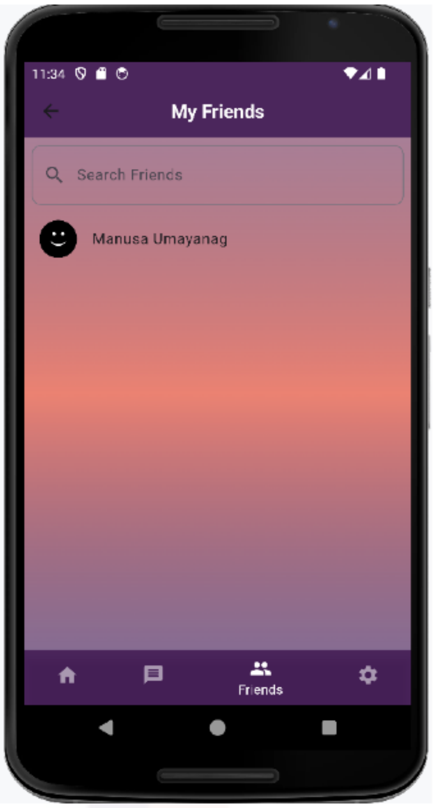
a. Daily, Weekly, and Monthly Analysis

1. Navigate to the “Stats” section.
2. View performance trends over different periods (daily, weekly, monthly).
3. Analyze your data to identify patterns and areas for improvement.

b. Data Visualization

1. Access visualizations such as charts and graphs that display your performance metrics.
2. These visual aids help you understand your progress and make informed decisions about your training.

9. Social Engagement



1. Navigate to the “Friends” section.
2. Use the search bar to find and connect with other users.
3. Send friend requests and view your friends’ activities.

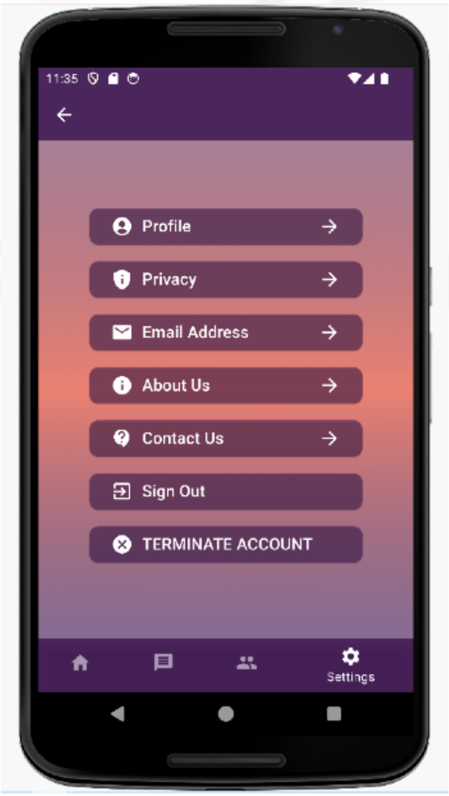
10. Security and Privacy

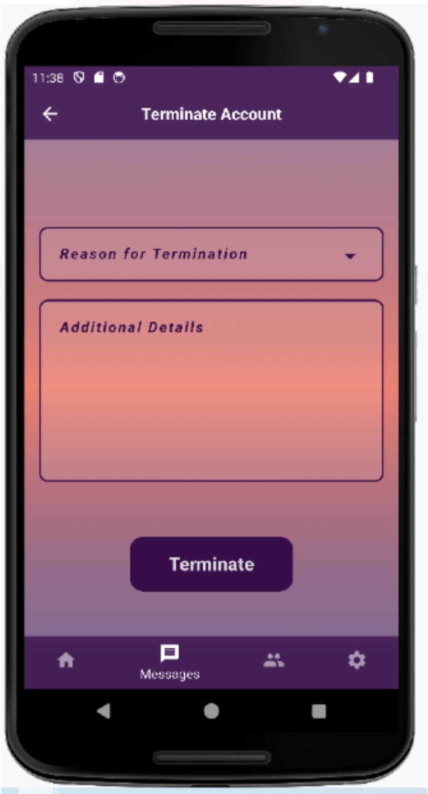
1. All data transmitted between your app and the backend servers is encrypted.
2. This ensures that your personal information and performance data are securely handled.

11. Terminating Account

If you wish to terminate your account:

1. Navigate to the “Settings” section.
2. Select “Terminate Account”.
3. Provide the reason for termination and any additional details.
4. Tap “Terminate” to complete the process.





12. Additional Settings

a. Managing Email Addresses

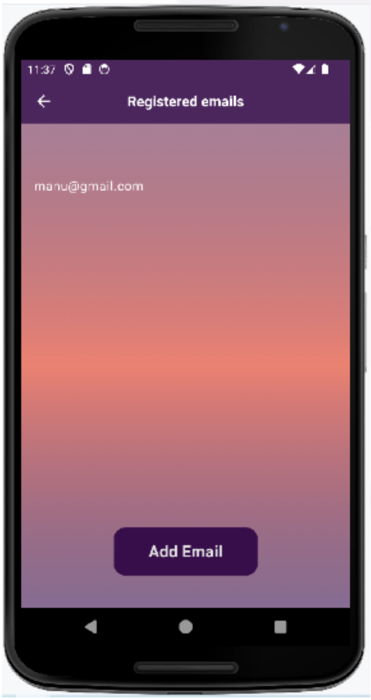
1. Navigate to the “Email Address” section under “Settings”.
2. Add or remove email addresses associated with your account.

b. About Us

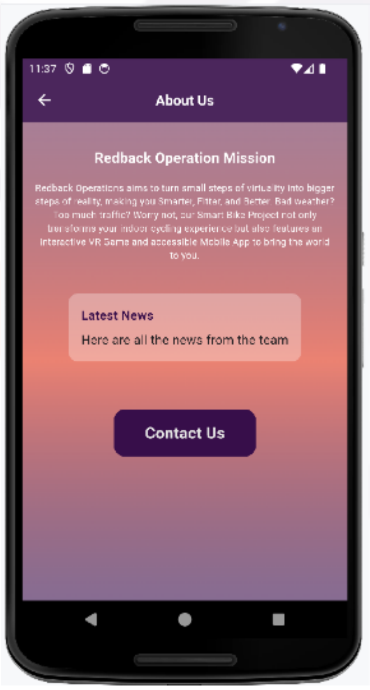
1. Navigate to the “About Us” section under “Settings” to learn more about Redback Operations and the SMARTBIKE project.

c. Contact Us

1. Navigate to the “Contact Us” section under “Settings” for support or inquiries.
2. Fill out the form with your subject, topic, and message.
3. Tap “Send” to submit your inquiry.



manu@gmail.com

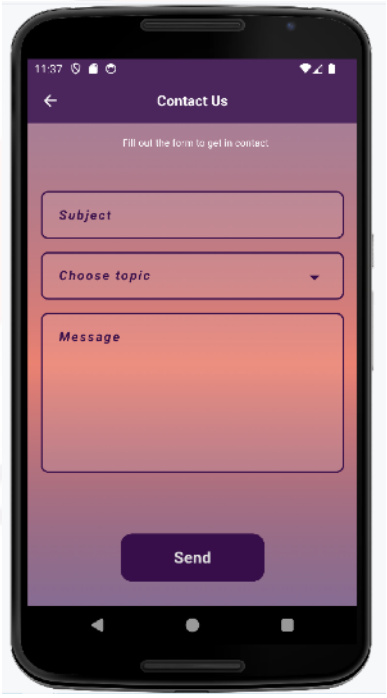


Redback Operation Mission

Redback Operations aims to turn small steps of virtuality into bigger steps of reality, making you Smarter, Fitter, and Better. Bad weather? Too much traffic? Worry not, our Smart Bike Project not only transforms your indoor cycling experience but also features an interactive VR Game and accessible Mobile App to bring the world to you.

Latest News

Here are all the news from the team



Fill out the form to get in contact

Requirements

Before you start using the SMARTBIKE mobile app, ensure that your system meets the following requirements. This section outlines the software, hardware, and environmental prerequisites for optimal performance and compatibility.

1. Operating System

Ensure that your device is running one of the following supported operating systems:

- **Android**: Version 6.0 (Marshmallow) or later.
- **iOS** (if applicable): Version 12.0 or later.

2. Hardware Requirements

To ensure smooth operation and optimal performance, your device should meet these hardware specifications:

- **Processor**: Quad-core 1.5 GHz or higher.
- **RAM**: Minimum 2 GB.
- **Storage**: At least 100 MB of free storage space for app installation, plus additional space for data storage.
- **Display**: A screen resolution of at least 720x1280 pixels.
- **Sensors**: GPS, Accelerometer, and Gyroscope for accurate activity tracking.

3. Network Requirements

A stable internet connection is required for various functionalities such as synchronization, data upload, and accessing community features:

- **Internet Connection**: Wi-Fi or mobile data connection with a minimum speed of 1 Mbps.
- **Bluetooth**: For connecting to external sensors and devices (optional).

4. Software Requirements

Ensure that the following software and tools are installed on your development environment for building and running the app:

a. For Android Development

- **Android Studio**: Latest stable version. Download from Android Studio.
- **Android SDK**: Included with Android Studio. Ensure you have installed the SDK tools and platforms for the target Android versions.

b. For Flutter Development

- **Flutter SDK**: Latest stable version. Follow the installation instructions at Flutter Install.
- **Dart SDK**: Included with Flutter SDK.

c. For Backend Development

- **Python**: Version 3.8 or later. Download from Python Downloads.
- **Django**: Version 3.1 or later. Install using pip install Django.
- **Django REST framework**: Version 3.12 or later. Install using pip install djangorestframework.

5. Development Environment

Set up your development environment with the following tools and configurations:

a. Version Control

Git: For cloning repositories and version control. Download from Git.

b. Integrated Development Environment (IDE)

Visual Studio Code: Recommended for both front-end and back-end development. Download from VS Code.

Extensions:

- Python extension for VS Code.
- Flutter and Dart extensions for VS Code.

6. Environment Configuration

Ensure the following configurations for seamless operation and development:

a. Environment Variables

Set up environment variables to store sensitive data and configuration details:

.env File: Create a .env file in the project root with the following variables:

```
SECRET_KEY=<your_secret_key>
EMAIL_BACKEND=<your_email_backend>
DEFAULT_FROM_EMAIL=<your_default_from_email>
API_URL_BASE=http://<your_machine_network_address>:8000
```

b. Backend Server Configuration

- **Virtual Environment**: Set up a virtual environment for the backend server:

```
python3 -m venv .venv
source .venv/bin/activate  # On macOS/Linux
.venv\Scripts\activate  # On Windows
```

- **Dependencies**: Install required dependencies:

```
pip install Django djangorestframework
```

7. External Devices

For a complete SMARTBIKE experience, the following external devices can be connected:

- **Heart Rate Monitors**: Compatible Bluetooth heart rate monitors.
- **Cadence Sensors**: For tracking cycling cadence.
- **Power Meters**: For measuring cycling power output.

9. User Account Requirements

To access all features of the SMARTBIKE app, users must:

- **Create an Account**: Provide a valid email address and set up a secure password.
- **Profile Information**: Optionally, users can add additional information such as name, profile picture, and bio for a personalized experience.

Common Errors and Solutions

While using the SMARTBIKE mobile app, you may encounter some common issues. Below are potential errors and their solutions to help you troubleshoot effectively.

1. App Crashes on Startup

Error: The app crashes immediately after launching.

Solution:

1. Ensure your app is up to date with the latest version available on the Google Play Store.
2. Restart your device to clear any temporary issues.
3. If the problem persists, uninstall and reinstall the app.

2. Unable to Log In

Error: Cannot log in to the app with correct credentials.

Solution:

1. Check your internet connection to ensure it is stable.
2. Verify your login credentials are correct.
3. Use the “Forgotten Password?” option to reset your password if necessary.
4. If the issue continues, contact support via the “Contact Us” section in the app.

3. Tasks Not Syncing Across Devices

Error: Biking data and stats are not syncing between devices.

Solution:

1. Ensure you are logged into the same account on all devices.
2. Check your internet connection.
3. Navigate to the settings and manually trigger synchronization.
4. If syncing issues persist, log out and log back in on all devices.

4. Notifications Not Working

Error: Notifications for reminders, messages, or friend activities are not being received.

Solution:

1. Ensure notifications are enabled in your device settings for the SMARTBIKE app.
2. Check the app's notification settings to make sure all relevant notifications are enabled.
3. Restart the app and your device.
4. If the issue continues, reinstall the app.

5. Error in Primary Color Configuration

Error: Error related to the primary color in your Android app.

Solution:

1. Look for the error in Android Studio.
2. Click on the red bulb icon next to the error.
3. Select "Show Configuration" to change the primary color to a different color.
4. Apply the changes and check if the error is resolved.

6. Pillow Installation Error

Error: backend_server.acc_details.image: (fields.E210) Cannot use ImageField because Pillow is not installed.

Solution:

Install the Pillow library by running one of the following commands in the VS Code terminal:

`python -m pip install Pillow` or `python3 -m pip install Pillow`

Once Pillow is installed, rerun the code that caused the error.

7. Database Migration Issues

Error: Errors during database migrations.

Solution:

1. Ensure your virtual environment is activated.
2. Run the following commands to make and apply migrations:

```
python3 manage.py makemigrations
python3 manage.py migrate
```

or

```
python manage.py makemigrations
python manage.py migrate
```

3. If issues persist, check for any conflicting migration files and resolve them.

8. API URL Base Configuration Error

Error: API calls failing due to incorrect base URL configuration.

Solution:

1. Ensure the .env file is correctly configured with the API_URL_BASE value.
2. Verify the network address is correct and the backend server is running.
3. Restart the app to apply the new configuration.

9. Dotenv File Issues

Error: Errors related to the dotenv file or environment setup.

Solution:

1. Uninstall the dotenv package if it is causing issues: pip uninstall python-dotenv
2. Delete the existing virtual environment:

```
rm -rf .venv  # On macOS/Linux
rmdir .venv /s /q  # On Windows
```

3. Create a new virtual environment:

```
python3 -m venv .venv
source .venv/bin/activate  # On macOS/Linux
.venv\Scripts\activate  # On Windows
```

4. Reinstall the dotenv package: pip install python-dotenv

License

This project is licensed under the MIT License - see the LICENSE file for details.

MIT License

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.