**Requirements Sheet for Sustainable Living Assistant App**

**1. Initial Setup Requirements**

**Objective:** Establish the foundation for development by defining project goals, resources, and preliminary tools.

**1.1 Project Goals**

* Define clear objectives for the app (e.g., raising environmental awareness, reducing user’s carbon footprint).
* Outline measurable success metrics (e.g., user retention, number of completed challenges).

**1.2 Team Formation**

* Assemble a development team: frontend developers, backend developers, UI/UX designers, and a project manager.
* Assign a sustainability advisor to ensure content accuracy.

**1.3 Tools and Technologies**

* Select tech stack:
  + **Frontend:** Flutter or React Native.
  + **Backend:** Node.js or Django.
  + **Database:** PostgreSQL or MongoDB.
* Identify required APIs:
  + Carbon footprint calculations.
  + Banking data integration (e.g., Plaid).
  + Smart device integrations.
* Set up version control (e.g., GitHub, GitLab).
* Choose project management tools (e.g., Jira, Trello).

**2. User Research and Design Requirements**

**Objective:** Understand user needs and design an intuitive interface.

**2.1 Market Research**

* Analyze existing sustainable living apps to identify gaps and opportunities.
* Conduct surveys or interviews with potential users to gather insights.
* Create user personas to represent target audiences (e.g., urban millennials, eco-conscious families).

**2.2 User Experience Design**

* Develop wireframes and user journey maps.
* Create a clickable prototype for testing core functionalities.
* Collect feedback on prototype usability from a sample audience.

**2.3 Visual Design**

* Design a clean, eco-friendly UI theme with intuitive navigation.
* Ensure accessibility compliance (e.g., color contrast, screen reader compatibility).

**3. Core Feature Development Requirements**

**Objective:** Build and integrate primary app functionalities.

**3.1 Carbon Footprint Tracker**

* Develop backend logic to calculate carbon footprint based on user data.
* Integrate APIs for banking and shopping data analysis.
* Design a dashboard for users to view detailed carbon footprint breakdowns.

**3.2 Personalized Sustainability Tips**

* Implement an AI/ML module to generate customized suggestions based on user habits.
* Set up a notification system for daily/weekly tips.

**3.3 Sustainability Challenges**

* Create gamification logic to reward users for completing challenges.
* Build a leaderboard to foster community competition.
* Integrate a badge/reward system for motivation.

**4. Advanced Feature Development Requirements**

**Objective:** Add features to enhance user engagement and functionality.

**4.1 Eco-Impact Planner**

* Enable users to set sustainability goals and track progress.
* Design visual reports for impact assessment (e.g., graphs, charts).

**4.2 Community Engagement**

* Develop forums or chatrooms for local sustainability discussions.
* Add features for organizing events (e.g., cleanups, carpooling).

**4.3 Smart Device Integration**

* Sync with IoT devices like thermostats and fitness trackers.
* Provide alerts and suggestions based on device usage patterns.

**4.4 Offset Marketplace**

* Partner with verified environmental organizations.
* Implement payment gateway for users to contribute to projects.

**5. Testing and Quality Assurance Requirements**

**Objective:** Ensure the app functions as intended and provides a seamless experience.

**5.1 Functional Testing**

* Test each feature independently for proper functionality.
* Verify API integrations and data accuracy.

**5.2 Usability Testing**

* Conduct user testing sessions to identify pain points.
* Iterate on design and functionality based on feedback.

**5.3 Performance Testing**

* Test app performance under different conditions (e.g., high traffic, low network connectivity).

**5.4 Security Testing**

* Ensure user data is encrypted and complies with GDPR/CCPA regulations.
* Test for vulnerabilities in financial and IoT integrations.

**6. Deployment and Launch Requirements**

**Objective:** Release the app to the public and establish support channels.

**6.1 Deployment**

* Set up cloud hosting for backend services (e.g., AWS, Google Cloud).
* Publish the app on app stores (Apple App Store, Google Play Store).

**6.2 Marketing**

* Launch a marketing campaign targeting eco-conscious audiences.
* Partner with influencers and organizations in the sustainability space.

**6.3 User Support**

* Establish a helpdesk for user inquiries and troubleshooting.
* Build an FAQ section within the app.

**7. Post-Launch Requirements**

**Objective:** Maintain and improve the app over time.

**7.1 User Feedback**

* Continuously collect user feedback through surveys and reviews.
* Prioritize feature requests for future updates.

**7.2 Regular Updates**

* Release updates to fix bugs, improve performance, and add features.
* Monitor app analytics to identify areas for improvement.

**7.3 Scaling**

* Optimize backend infrastructure to handle growing user bases.
* Expand partnerships for challenges and the offset marketplace.

**Chronological Implementation Plan**

1. Initial Setup (1 month)
2. User Research and Design (2 months)
3. Core Feature Development (3-4 months)
4. Advanced Feature Development (3 months)
5. Testing and Quality Assurance (1-2 months)
6. Deployment and Launch (1 month)
7. Post-Launch Maintenance (Ongoing)