Project Documentation: Gamified Habit Tracker App

Project Overview

Project Name: HabitQuest

<u>Objective</u>: Help users build and maintain positive habits using gamification techniques such as experience points (XP), badges, levels, streaks, and social challenges.

Target Users:

- Individuals looking to improve daily routines
- Students, professionals, and wellness seekers
- Productivity and self-improvement communities

Functional Features

Core Habit Features

- Create and customize habits (name, frequency, reminders)
- Daily habit check-in
- Habit calendar and streak view
- Progress tracker per habit

Gamification Features

- XP (Experience Points) for completing habits
- Level system based on total XP
- Streak rewards (bonus XP, badges)
- Achievements/badges (e.g., "10 Days Streak")
- Daily & weekly challenges
- Avatar and theme unlocks

Social & Community (Optional for MVP)

- Compete with friends on leaderboards
- Share achievements
- Join group challenges

Gamification Logic (Example)

Action Reward

Daily habit check-in +10 XP

7-day streak +100 XP + badge

Completing challenge +250 XP

Level up New avatar/theme

Architecture Overview

Frontend:

• Mobile App: Flutter / React Native (recommended)

• Web App (optional): React.js / Next.js

Backend:

• Language: Node.js (Express) or Python (FastAPI)

• **Database**: PostgreSQL or MongoDB

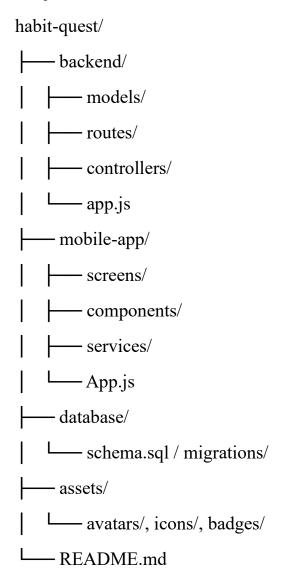
• Notifications: Firebase Cloud Messaging / OneSignal

Hosting:

• Frontend: Expo Go, Firebase Hosting, or Vercel

• Backend: Render, AWS, or Railway

Project Structure



Database Schema (Simplified)

Tables:

- Users: id, name, email, password, level, total_xp
- Habits: id, user id, name, description, frequency, is active
- HabitLogs: id, habit_id, date_completed
- Badges: id, name, description, icon
- UserBadges: id, user id, badge id, awarded at
- Challenges: id, name, description, start_date, end_date
- UserChallenges: id, user_id, challenge_id, progress

API Endpoints (Sample)

Method Endpoint	Description
-----------------	-------------

POST /api/auth/signup User registration

POST /api/habits Create new habit

GET /api/habits List user habits

POST /api/habits/:id/check-in Log habit completion

GET /api/user/xp Get XP, level, streaks

GET /api/user/badges Get unlocked badges

POST /api/challenges/join Join a new challenge

Authentication

- JWT-based auth system
- OAuth (Google or Apple) for easier login
- Secure API token validation

Development Timeline (10 Weeks MVP)

Phase Timeline

Requirement Planning Week 1

UI/UX Design Week 2–3

Backend Development Week 3–5

Mobile Frontend Week 4–7

Gamification Engine Week 6–7

Notifications & Reminders Week 7–8

Testing & Bug Fixes Week 9

Phase Timeline

Deployment Week 10

UI/UX Ideas

Screens:

- 1. **Dashboard** Today's habits, XP, streaks
- 2. Habit Tracker List, create/edit, calendar view
- 3. **Progress** XP progress, streaks, achievements
- 4. Challenges Join public or friends' challenges
- 5. **Profile** Avatar, level, settings

Design Tools:

• Figma, Adobe XD, or FlutterFlow for quick prototyping

Testing Strategy

- Unit Tests: Backend habit logic, XP, level-up
- Integration Tests: Habit check-in flows
- **UI Testing**: With Detox (React Native) or Flutter test
- Manual QA: Notifications, reminders, offline mode

Tech Stack Summary

Layer Tech

Mobile App Flutter / React Native

Backend API Node.js (Express) / FastAPI

Database PostgreSQL / MongoDB

Notifications Firebase / OneSignal

Auth JWT / OAuth

Layer Tech

Hosting Render / Railway / Firebase

Dev Tools GitHub, Figma, Postman

Future Enhancements

- Daily habit scoring (difficulty levels)
- AI habit suggestions based on goals
- Pomodoro timer integration
- Wearable or smartwatch support
- Offline sync capability