HabitForge

(React + Spring Boot + MySql)

Reyansh Goswami

UID: 23BCS11136

Aug 2025-Nov2025

Table of Contents:

Chapter 1: Introduction

HabitForge is an innovative productivity application designed to help users develop and sustain positive habits through the power of gamification. The project aims to transform routine habit tracking into a rewarding and motivating process by leveraging streaks, badges, and progress stats. With interactive visual feedback and engaging mechanics, HabitForge provides a fun and effective way to promote consistency and personal growth.

Chapter 2: Objectives

- To design and implement a user-friendly platform for tracking daily habits.

- To integrate gamification elements such as streak counters, level-based badges, and progress visualization.

- To provide users with weekly breakdowns and analytics for better self-assessment.

- To ensure scalability and maintainability using modern full-stack technologies.

- To create a project that is both academically valuable and portfolio-ready.

Chapter 3: Features

1. **Habit Calendar** – Visual representation of user's habit completions and streaks.

2. **Badges & Rewards** – Motivational rewards for achieving milestones such as 7-day streaks or 30-day consistency.

3. **Progress Stats** – Analytical dashboard to view performance trends with weekly/monthly breakdowns.

4. **Gamification Engine** – Automatic assignment of badges and streak tracking logic.

5. **User Management** – Secure authentication and personalized dashboards.

Chapter 4: Technologies Used

- **Frontend:** React.js for building a responsive and interactive UI.

- **Backend:** Spring Boot for creating RESTful APIs and handling business logic.

- **Database:** MySql for storing flexible user and habit data.

- **Tools:** VS Code

Chapter 5: System Design

5.1 High-Level Design (HLD)

The system follows a three-layer architecture:

- **Frontend (React)**: Habit calendar, stats dashboard, and badges UI.

- **Backend (Spring Boot)**: REST APIs, authentication, and gamification logic.

- **Database (MySql)**: Storage of users, habits, completions, and rewards.

5.2 Low-Level Design (LLD)

- **Users Collection**: Stores user profiles, login credentials, and earned badges.

- **Habits Collection**: Stores user habits with streak counts and descriptions.

- **Completions Collection**: Logs daily habit completions with timestamps.

- **Badges Collection**: Stores badge metadata and criteria.

Chapter 6: Project Phases

Phase 1: Requirement Analysis & Planning

This phase involved understanding the need for a gamified habit tracker and defining the project scope. The objective was to create a platform that motivates users to stay consistent by offering streaks, badges, and progress analytics. During this stage, system requirements such as user authentication, habit logging, and gamification rules were identified. The database schema (collections in MySql) was also outlined to store users, habits, completions, and rewards efficiently.

Phase 2: Frontend Development (React)

In this phase, the **user interface** of HabitForge was developed using React. The key modules included:

- **Habit Calendar**: Displays habits and allows users to mark them as complete for the day. Visual indicators such as colored blocks highlight streaks.

- **Badges Module**: A dedicated section where users can view earned badges and rewards for consistency milestones.

- **Progress Dashboard**: Provides weekly and monthly analytics through visual charts and graphs.

This phase focused on creating an interactive and responsive interface that motivates users through engaging visual design.

Phase 3: Backend Development (Spring Boot)

The backend was built using **Spring Boot**, providing a set of RESTful APIs to handle business logic. Major services included:

- **User Service**: Handles registration, login, and profile management.

- **Habit Service**: Provides APIs for creating, updating, and deleting habits, as well as marking daily completions.

- **Gamification Service**: Implements rules for streak calculation, badge allocation, and progress scoring.

- **Analytics Service**: Aggregates habit completion data and returns performance statistics to the frontend.
  This phase ensured that all business rules and gamification logic were securely and efficiently executed.

Phase 4: Database Integration (MySql)

This phase involved designing and integrating the MySql collections with the backend APIs. Collections included:

- **Users Collection**: Stores user details and badges.

- **Habits Collection**: Stores habit details, streak counts, and frequency.

- **Completions Collection**: Stores daily logs of habit completions.

- **Badges Collection**: Defines badge metadata and criteria.
  The use of MySql allowed for flexible data storage and easy retrieval of user-specific statistics. This phase also ensured proper indexing and query optimization for efficient API performance.

Phase 5: Testing & Deployment

Once development was complete, the system was tested for correctness and reliability. This included:

- **Unit Testing** for individual modules (React components, API endpoints).

- **Integration Testing** to validate smooth communication between frontend, backend, and database.

- **User Testing** with sample data to evaluate streak calculations, badge awards, and statistics accuracy.

Finally, the project was packaged and deployed on a local server, with future scope to host it on cloud platforms for wider accessibility.

Chapter 7: Future Scope

- Integration of **push notifications** for daily reminders.

- **Social features** like group challenges and leaderboards.

- **Advanced analytics** with long-term performance tracking.

- Development of a **mobile app** for Android/iOS platforms.