StudyStreak App

1.1 System Description

A high-level description of the system from a business and user perspective. Make it very clear what the purpose of the system is and any important high-level aspects of how it will accomplish this task. **(5 Marks)**

StudyStreak is a mobile application designed to promote and reward consistent studying habits while building an engaged learning community. The system combines proven productivity techniques, gamification elements, and social features to help students maintain their study momentum and achieve their academic goals.

From a business perspective, StudyStreak addresses the growing demand for digital learning tools that make studying more engaging and rewarding. The app employs a comprehensive reward system, progress tracking, and community features to transform regular studying into an achievement-oriented activity.

From a user perspective, StudyStreak serves as a personal study companion that helps students maintain focus, track progress, and earn rewards for consistent studying. The app provides tools for time management, subject organization, and progress visualization, while fostering a supportive community of learners.

1.2 Business Case

What is the motivation behind the project? What problems will this solve for your target audience? What competitive advantages does it offer? How can it improve efficiency, effectiveness or workers? If it is for entertainment, then what will make it compelling for users? **(5 Marks)**

The motivation behind StudyStreak stems from several key observations in the educational technology space:

- ❖ Students often struggle with maintaining consistent study habits, leading to cramming and poor retention
- ❖ Traditional study methods lack immediate feedback and rewards that could reinforce positive behaviors
- ❖ Existing educational apps focus primarily on content delivery rather than habit formation
- ❖ There's a growing market for gamified educational tools, especially among younger demographics

Competitive advantages:

- ❖ Unique reward system that combines virtual and real-world benefits
- ❖ Integration with popular educational platforms and resources
- ❖ Focus on building long-term study habits rather than just content consumption
- ❖ Community-driven features that create accountability and peer support

The app improves efficiency by:

- ❖ Reducing procrastination through immediate reward feedback
- ❖ Promoting spaced repetition and consistent study schedules
- ❖ Providing data-driven insights into study patterns and effectiveness
- ❖ Facilitating peer support and collaborative learning opportunities

1.3 User-level goals for the system

State the main expectations and requirements the user will have for your system. **(5 Marks)**

1. **Study Session Management**
   - ❖ Easy-to-use timer interface for tracking study sessions
   - ❖ Flexible session scheduling and reminder system
   - ❖ Support for different study techniques (Pomodoro, flow state, etc.)

2. Progress Tracking and Rewards
- ❖ Clear visualization of daily/weekly/monthly study streaks
- ❖ Achievement system with meaningful milestones
- ❖ Point-based reward system redeemable for real benefits
- ❖ Detailed analytics of study patterns and productivity

3. Content Organization
- ❖ Subject-based organization of study materials
- ❖ Task prioritization and deadline management
- ❖ Integration with popular notetaking and learning platforms
- ❖ Cloud sync across devices

4. Community Features
- ❖ Study group formation and management
- ❖ Peer accountability partnerships - Knowledge sharing and discussion forums
- ❖ Virtual study rooms for group sessions

5. Customization and Accessibility
- ❖ Personalized study goals and rewards - Offline functionality for core features
- ❖ Cross-platform availability
- ❖ Accessibility options for different learning styles

1.4 Project Team
List all project team members here and ensure that the time they spend on the project is accounted for. You can explain their contribution (the time spent; the problem worked on) until the submission of milestone-1 for this project. The following table is only here to provide you with an example. Feel free to change the following table according to your need. **(2 Marks)**

Name	Responsibility	Availability	Task
Kayden Ions	Project Manager & UI/UX Lead	15 Hours / Week	- Manages team progress, conducts user research, creates wireframes, and designs UI components for study features - Proposed a Reading app to the team - Created a poll to figure out the type of program to develop. - Created a GitHub Repository - Created surveys for baseline research and sent out to targeted user groups. - Set up focus groups and individual interviews. - Conducted a group meeting to update team members on all needed features, prioritized features, and begin iterative development. - Worked on wireframes and basic UI for needed features.

Tuan Nguyen	Mobile Developer	14 Hours / Week	- Initially proposed the idea of a Home Automation System to the team. - Established mobile development environment and project structure. - Implemented native mobile components based on UI/UX lead's wireframes. - Created mobile-specific features including study timer and offline functionality. - Developed mobile data synchronization with backend systems. - Built mobile-optimized interfaces for community features. - Integrated cross-platform compatibility solutions. - Addressed QA feedback for mobile-specific issues. - Implemented mobile push notifications system.
Hong Xuan Quy Nguyen	Backend Developer	14 Hours/Week	- Proposed Food Donation web platform to the team. - Design and implement a scalable database schema. - Build RESTful API, integrated external APIs. - Set up secure authentication and implement user profile management and roles. - Manage data handling for virtual study rooms, group chats, and study group recommendations. - Ensure the backend is optimized for fast response times. - Build backend logic to support cloud synchronization across devices. - Implement load balancing and scaling strategies to handle high usage periods.
Alex Nickerson	Community Lead	12 Hours / Week	- Proposed a Study Streak application to the team - Created sample study materials for testing - Created engaging achievements and rewards - Established content moderation guidelines - Created documentation for study techniques and best practices - Developed content guidelines and

			templates
Shafiul Kalam	Frontend Developer	15 Hours / Week	- Proposed a Shopping Assistant app to the team - Helped design and implement the user Interface - Helped troubleshoot and debug UI bugs - Implemented accessibility features - Helped maintain code quality
Kevin Bui	QA & User Testing	12 Hours / Week	- Proposed a university campus event planner application - Tested usability and viability of various software processes - Checked compatibility across different platforms and devices - Identified potential bugs and usability issues - Gathered user feedback and suggestions - Assessed common vulnerabilities like SQL injection, ensuring greater security

Each team member is a Computer Science major in their third year, balancing this project with their regular course load. Team meetings are held twice weekly: Tuesdays at 3:30 PM and Thursdays at 4:00 PM, scheduled around class times. Additional collaboration occurs through Discord and GitHub.

1.5 User involvement plan
You will be involving users throughout your development process (consider the complete development of the project). Write a plan for how you intend to involve users, including which users you will involve, how much of their time you will need, when you will need it, and why. **(3 Marks)**

Target User Groups:
1. High School Students (Ages 14-18)
2. University Students (Ages 18-24)
3. Adult Learners (Ages 25+)
4. Education Professionals

Involvement Schedule:
- ❖ Initial Research Phase (Weeks 1-2)
  - o Purpose: To gather baseline user insights, understand study habits, and preferences.
  - o Activities:
    - ▪ Online surveys: 15-20 minutes per user, aiming for 50 – 100 participants per

group. This will help identify trends and any user needs.
- Focus groups: 1-hour sessions with 6-8 users. This will help to understand any potential challenges users face in their study routines.
- Individual interviews: 30 minutes per user. Allows for in-depth, personalized feedback about study habits and technology usage.

❖ Design Phase (Weeks 3-4)
  o Purpose: To ensure the app's design is aligned with user expectations and needs.
  o Activities:
    - UI/UX testing sessions: 45 minutes per user. A group of 8 users from each target group to test early wireframes and app prototypes to assess usability.
    - Feature priority workshops: 2 hours with mixed user groups. Two-hour sessions with mixed user groups to help prioritize which features to focus on and validate design choices.

❖ Development Phase (Weeks 5-12)
  o Purpose: To refine the app through iterative testing and gather feedback on specific features as they are developed.
  o Activities:
    - Weekly beta testing: 1 hour per week per user. Users will test new features and provide feedback to improve functionality.
    - Feature feedback sessions: 30 minutes bi-weekly. Focused on new and updated features. Participants will provide feedback on functionality, performance, and user experience of specific features.

❖ Launch Phase (Weeks 13-16)
  o Purpose: This is to ensure that the app meets full user expectations and is ready for release.
  o Activities:
    - Usability testing: 1 hour per user. Focuses on final design and feature set. This ensures that the app is intuitive and easy to navigate for all groups.
    - Performance monitoring interviews: 30 minutes per user. After the release, users will provide any feedback for any technical issues, bugs, or usability issues.

1.6 Use Cases
Develop five major use cases for the above system using the five parts components (Actors, Input, Output, Normal Description, Exception). **(10 Marks)**

1. **Starting a Study Session**
   ❖ Actors: Student
   ❖ Input: Subject selection, study duration, study technique preference
   ❖ Output: Active timer, progress tracking, reward points
   ❖ Normal Description: User selects subject, sets timer, starts session, receives points upon completion
   ❖ Exception:
     o Session interruption: If the session is interrupted through crashing or a phone call, the session timer pauses, and the app must notify the user.
     o Partial credits: If the session is incomplete, the app should calculate partial progress based on time studied and awards points.

2. **Joining a Study Group**
   ❖ Actors: Student, Group Admin
   ❖ Input: Group search criteria, join request
   ❖ Output: Group membership, access to shared resources
   ❖ Normal Description: User searches for groups, sends request, gets approved, joins group activities
   ❖ Exception:

- o Group Capacity Limits: If the group capacity is at its max limit, display a message to the user and show a list of other, similar groups.
- o Membership Requirements Not Met: If the user lacks membership requirements like study level or location, an error message must be displayed which prompts the user to select a similar group with different requirements.

3. Redeeming Rewards
- ❖ Actors: Student, Reward System
- ❖ Input: Point balance, reward selection
- ❖ Output: Reward voucher or benefit activation
- ❖ Normal Description: User views available rewards, selects desired item, confirms redemption
- ❖ Exception:
  - o Insufficient points: If the user has insufficient points for a selected reward, display a message that reads "Insufficient points, study more to earn more points!"
  - o Expired rewards: If the selected reward has expired, the app will display a message that reads "This reward has expired".
  - o Redemption limits: If there are restrictions on how many rewards can be redeemed per user (per day or week), the app should notify the user "You have reached you reward limit for the day/week".

4.Creating Study Goals
- ❖ Actors: Student
- ❖ Input: Goal parameters, timeframe, subject areas
- ❖ Output: Structured study plan, progress metrics
- ❖ Normal Description: User sets specific goals, receives planning suggestions, tracks progress
- ❖ Exception:
  - o Conflicting Goals: If the user sets conflicting goals (too many goals within a certain period) then the app should highlight these areas in read and display a message that reads "Your goals conflict with each other. Adjust the time frame or reduce the number of goals".
  - o Unrealistic Time Frames: If the time to study material is considered too unrealistic (Studying an entire semester worth in a week) the app should display a message that reads "This goals time frame is too short. Adjust your goal to make it more feasible".

5. Participating in Group Challenges
- ❖ Actors: Student, Challenge Participants
- ❖ Input: Challenge registration, study activity
- ❖ Output: Challenge progress, rankings, rewards
- ❖ Normal Description: User joins challenge, completes required activities, earns achievements
- ❖ Exception:
  - o Missing Deadlines: If the user misses the deadline for a challenge, the app should notify the user that they missed the deadline and should check upcoming challenges or join another.
  - o Incomplete Participation Requirements: If the user does not meet participation requirements the app should notify the user or display a prompt that reads "You did not meet the challenge criteria, please complete the required tasks to earn points".