

PROJECT PROPOSAL DOCUMENT

PROJECT TITLE

FOCUS – A Personalized Learning Platform

PROBLEM STATEMENT

Students often struggle with procrastination, poor study planning, and low motivation. Many existing tools are either too complicated or too basic, so they do not help students stay consistent. Also, managing assignments and daily study tasks becomes stressful when everything is not in one place.

OBJECTIVES OF THE PROJECT

- To develop a distraction-free study timer that tracks focus sessions by subject.
- To implement a visual 'Treemap' for assignments that highlights urgency intuitively.
- To track and display daily study streaks to encourage consistent learning habits.
- To provide a centralized dashboard for notes, tasks, and productivity analytics.

BRIEF DESCRIPTION OF THE PROPOSED SOLUTION

FOCUS is a web-based application designed to bridge the gap between task management and habit tracking. It features a 'Focus Mode' timer, an intelligent assignment tracker that visually scales tasks based on deadlines. The application utilizes a dark, Notion-inspired aesthetic to minimize eye strain and maintain a calm study environment.

TECHNOLOGY STACK

Framework: Django 6.0 (Python)

Frontend: HTML5, Vanilla CSS3, Vanilla JavaScript

Database: SQLite (Development)

Visualization: Chart.js

EXPECTED OUTCOMES

By the end of this project, we will have a fully functional web application that allows users to register, log in, track their study hours in real-time, visualize pending assignments by urgency, and review their historical study performance.

ER DIAGRAM

Entity-Relationship (ER) Diagram

The following section describes the entities, attributes, and relationships modeled in the system.

Entity: User (Django Auth)

Attributes: ID (PK), Username, Password, Email, Date_Joined.

Entity: StudySession

Represents a completed block of study time.

Attributes: ID (PK), Subject, Duration (int), Date, Created_At.

Relationships: Many-to-One with User.

Entity: Assignment

Represents a task with a deadline.

Attributes: ID (PK), Title, Subject, Deadline, Estimated_Hours, Status (Pending/Completed), Urgency (Low/Med/High).

Relationships: Many-to-One with User.

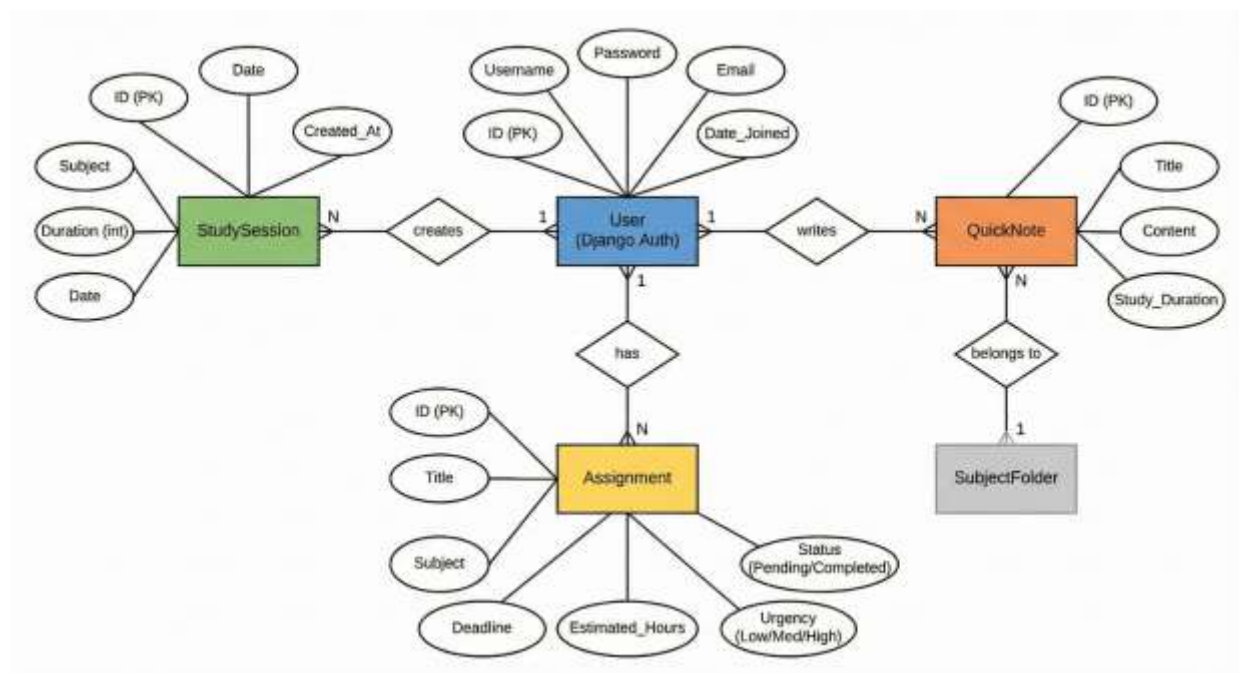
Entity: QuickNote

Represents short reflections added after study sessions.

Attributes: ID (PK), Title, Content, Study_Duration.

Relationships: Many-to-One with User, Many-to-One with SubjectFolder.

DIAGRAM



Requirements Specification Document

Functional Requirements

1. Authentication: Users must be able to sign up and log in securely.
2. Timer: Users can start, pause, and stop a study timer, selecting a subject.
3. Assignments: Users can add assignments with deadlines and mark them as complete.
4. Analytics: The system must display a weekly study chart and calculate streaks.

Non-Functional Requirements

1. Usability: The interface must be clean, 'distraction-free', and intuitive.
2. Performance: Dashboard and charts should load within 2 seconds.
3. Responsiveness: The application must be usable on mobile devices.
4. Reliability: Study data must be strictly persisted in the SQLite database.

Assumptions & Constraints

Assumption: Users verify their own study time (honesty system).

Constraint: Initial version uses SQLite; migration to PostgreSQL recommended for production.