Lebanese University
Faculty of Engineering III
Electrical and Electronic Department

SMART STUDY HABIT TRACKER

COMMUNICATION AND MINI-PROJECT

by

Malak Srour (6142)

Aya Hassan (6338)

Spring 2024-2025

Presented for : Dr. Mohammad Aoudi

TABLE OF CONTENTS

LIST OF FIGURES

ABSTRACT

The Smart Study Habit Tracker (SSHT) is a full-stack web application designed to help students monitor, analyze, and enhance their study patterns. By collecting daily study data, the system evaluates potential burnout risks using both rule-based logic and a machine learning model. Furthermore, it integrates ChatGPT-4 to provide intelligent and personalized feedback, offering actionable advice based on user behavior. This project aims to foster better academic habits through data-driven insights and AI assistance.

CHAPTER 1
INTRODUCTION

This chapter is the introduction of this report. It introduce the idea of the the project. It involves a general introduction, problem statement and the objective of the project.

1.1 General introduction

In the modern academic environment, students are increasingly facing challenges related to managing their study habits effectively. Poor study routines, insufficient breaks, irregular sleep patterns, and low concentration levels can lead to burnout, negatively impacting both academic performance and mental well-being. Recognizing the need for tools that help students monitor and improve their study behaviors, this project proposes the development of a Smart Study Habit Tracker (SSHT)**.**

The SSHT is a web-based application designed to help users track their daily study patterns, sleep hours, break frequency, and concentration levels. Leveraging machine learning and rule-based analysis, the system provides personalized feedback and risk assessments related to burnout. The application aims to empower students with insights and actionable recommendations to optimize their study habits and maintain a healthy balance between work and rest.

1.2 Problem Statement

Many students struggle to identify the causes of their academic stress and burnout due to a lack of objective data about their study routines. Without consistent tracking and analysis, it becomes difficult to recognize harmful patterns such as prolonged study hours without breaks, poor sleep, and low concentration levels. Existing productivity tools often lack specialized support for study-related behaviors and personalized advice tailored to individual needs.

Thus, there is a pressing need for a tool that can:

Collect reliable data on study habits and related factors.

Analyze this data to identify burnout risks.

Provide clear, personalized advice to help students improve.

Maintain a history of study sessions for progress monitoring.

1.3 Objective

The main objectives of the Smart Study Habit Tracker project are:

- **Data Collection:** Enable users to input daily study-related data easily via an intuitive web form.
- **Risk Assessment:** Use machine learning and rule-based algorithms to calculate burnout risk scores based on user inputs.
- **Personalized Feedback:** Provide tailored warnings, recommendations, and coaching advice to improve study habits.
- **Historical Analysis:** Allow users to view and manage their past study sessions over different time ranges.
- **User-friendly Interface:** Develop a responsive and accessible React frontend to facilitate smooth user experience.
- **Backend Integration:** Implement a Flask API backend for data processing, storage, and serving analytics.
- **Data Persistence:** Store user data securely in a database with features to clear or filter historical records.

1.4 Thesis Outline

The report contains four chapters organized as follows:

Chapter 1 introduce the idea of the project, problem statement and the objective

Chapter 2 states the development part. It present detailed description of the system overview, technologies used , system design and implementation details.

Chapter 3 gives the result after the implementation, the conclusion and the future work.

CHAPTER 2
DEVELOPMENT

2.1 Introduction

To complete this project, many tools were used in a certain way. The work between both frontend and backend was organized in a way that ensured compatibility between them . In this chapter, we will present detailed description of the system overview, technologies used , system design and implementation details.

2.2 System overview

The Smart Study Habit Tracker is structured as a client-server web application comprising two main components:

- **Frontend:** Built with React.js, it presents users with forms to input their daily study metrics, visualizes analysis results, and displays study history. The UI ensures validations and smooth interaction.
- **Backend:** A Flask-based REST API service receives data from the frontend, processes it using ML models and rule engines, stores session records in a database, and serves requests for history or usernames.

The communication between frontend and backend is facilitated through well-defined API endpoints, with cross-origin resource sharing (CORS) configured to allow safe data exchange.

2.3 Technologies Used

- **Frontend:** React.js, Axios for HTTP requests, CSS for styling.
- **Backend:** Python Flask framework, Flask-CORS for cross-origin support.
- **Database:** SQLite for data persistence.
- **Machine Learning:** A custom lightweight model (could be a rule-based classifier or ML model) for burnout risk prediction.
- **Version Control:** Git and GitHub for source code management.

2.4 System Design and Architecture

2.4.1 Overall Architecture

The system follows a three-tier architecture:

- Presentation Layer (Frontend): Handles user interaction and data input.
- Application Layer (Backend): Manages business logic, data validation, risk analysis, and API services.
- Data Layer (Database): Stores user study sessions and metadata.

2.4.2 API Endpoints

- POST /analyze: Accepts study habit data, computes burnout risk and advice, returns analysis.
- GET /history: Retrieves historical study sessions filtered by username and time range.
- DELETE /clear-history: Deletes historical records for a user within a specified range.
- GET /usernames: Returns a list of all stored usernames for search suggestions.

2.4.3 Data Flow

1. User inputs data in the form.
2. Frontend validates data and sends it to /analyze.
3. Backend computes risk and stores the record.
4. Frontend displays analysis results and recommendations.
5. User can view history via /history.
6. User can clear records using /clear-history.

2.5 Implementation Details

2.5.1 Frontend

- **FormInput Component:** Handles form input, validation, local storage of username, and API call for analysis.
- **ResultCard Component:** Displays risk score with color-coded indicators, warnings, recommendations, and ChatGPT advice.

- **HistoryList Component:** Enables viewing, filtering, and deletion of study session records with username search support.
- **Header Component:** Presents the application title and tagline.

2.5.2 Backend

- **Flask app.py:** Defines API routes with CORS, handles POST and GET requests, interacts with SQLite.
- **Rule Engine:** Implements simple classification logic to assign burnout risk based on thresholds.
- **ML Model:** Optionally integrates a machine learning model for more nuanced risk prediction.
- **Database:** Stores session data with fields such as username, study hours, sleep hours, break frequency, concentration level, risk score, and timestamp.

2.5.3 Machine Learning & Rule-based Analysis

- The system combines heuristic rules (e.g., study hours > 8, sleep hours < 6) and ML outputs to generate a comprehensive risk score.
- Warnings and recommendations are dynamically generated based on the risk score and specific input values.

CHAPTER 3
RESULTS

3.1 Introduction

This chapter includes the result of the previous implementations, the conclusion of the report and the future work which contains ideas to expand the work to make the project more developed and to be used in real live.

3.2 results

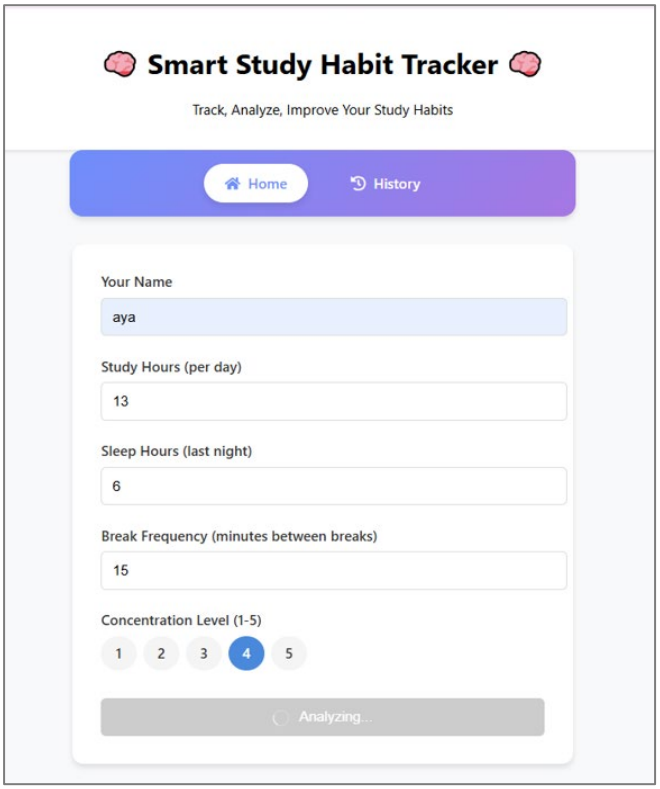

Figure 2 Home screen

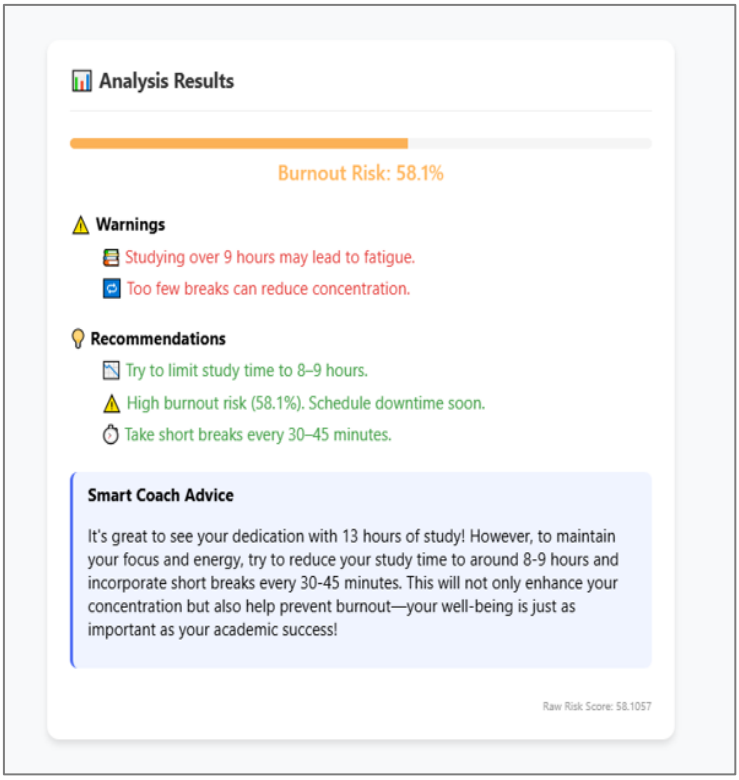

Figure 1 Analysis Result, with AI advice

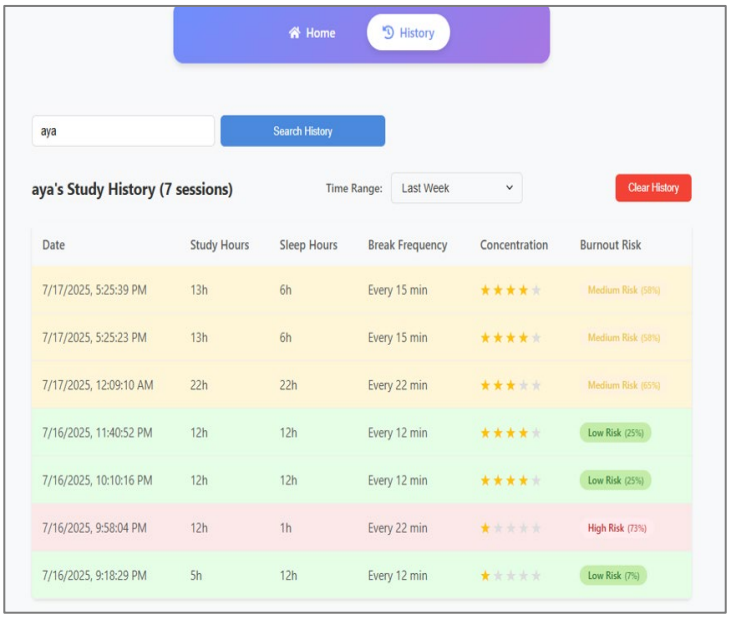

Figure 3 History list of user "Aya"

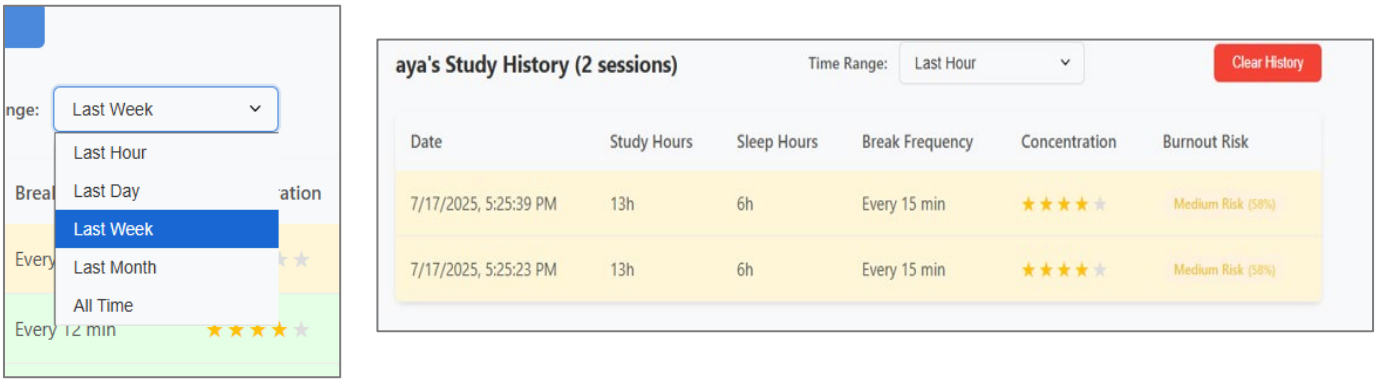

Figure 4 filtering history list by time

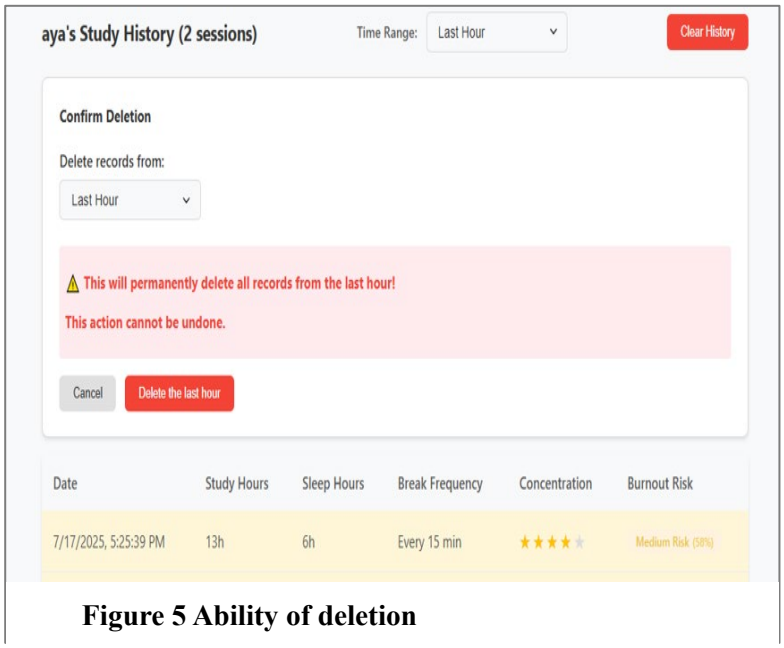

Figure 5 Ability of deletion

3.3 Conclusion

The Smart Study Habit Tracker provides a practical solution for students to monitor and improve their study habits. By combining data-driven analysis with user-friendly interfaces, the application aids in early detection of burnout risks and promotes healthier study practices. The project demonstrates how technology can support academic success and mental well-being effectively.

3.4 Future work

- Add login system and secure user authentication
- Use charts (e.g. Recharts) for historical trends
- Train a more accurate burnout detection ML model with custom data
- Make it suitable for mobile