CSYE 6200 FINAL PROJECT

EduPal

Team Name: Nova Squad

Team Members:
Vikas Meneni
Sai Shravan Neelamsetty
Narasimha Reddy Valam
Sreeja Pulaparty

Edupal

Student Grade & Mental Health Tracker

Project Overview:

This tracker is designed to help students manage their grades and mental health in an integrated way.

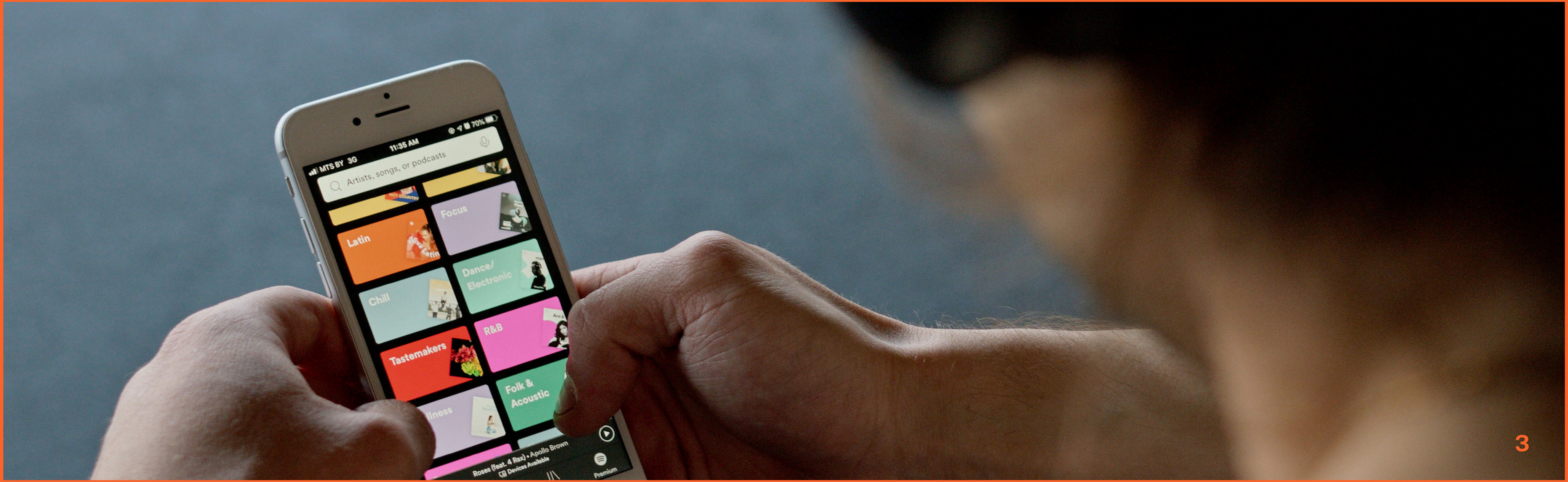
Purpose:

To empower students to track their academic performance and mental health in one place, fostering both success and well-being.

Target Audience:

College and university students seeking to manage academic stress and stay on top of their studies.

A Holistic Approach to Student Well-being and Academic Success



Problem We Plan to Solve:

01

Lack of Clarity on Academic Performance:
Students often don’t have a clear view of where they stand with their grades until it's too late.

02

Difficulty Finding Help:
With so many academic resources available, it can be tough to identify the right ones, leading to wasted time and frustration.

03

Mental Health Struggles:
The stress from exams and assignments can take a toll on students’ mental health, and many don’t reach out for help due to lack of awareness about available resources.

04

Lack of Work-Life Balance:
Juggling school, personal time, and social activities is difficult, and students often miss out on the chance to relax and recharge.

Our Timeline and Progress

Week 1 (Nov 4 - 10):

- Setup Git repository, create main branch, and organise tasks.
- Begin the layout design using Scene Builder.

Week 2 (Nov 11 - 17):

- Develop the rough version of the app with basic features: grade tracking, notifications, and initial UI components.
- Test all functionalities to ensure basic interaction.

Week 3 (Nov 18 - 24):

- Implement advanced features like student information, event listings, and personalized support.
- Refine application functionality and begin user testing.

Week 4
(Nov 25 - Dec 1):

- Focus on finalizing the UI to make it visually appealing and user-friendly.
- Conduct thorough testing for any bugs or functionality issues.
- Start preparing the presentation slides and video.

We are here!

Week 5
(Dec 2 - 7):

- Submit the video presentation (Dec 4).
- Finalize the report, source files, and slides for the project submission.

Our Solution: EduPal

EduPal is an innovative solution that serves as a personal academic guide and wellness buddy for students, combining both grade tracking and mental health management.

Grade Tracking:

- Students can track their grades, set academic goals, and receive alerts when they fall behind or exceed their expectations.

Personalized Support:

- EduPal offers customizable resources, including extra study materials, lecture recordings, and other academic support tools when needed.

Mental Health Focus:

- The platform offers mental health support. It also promotes social activities by highlighting fun campus events.

Work-Life Balance:

- EduPal reminds students to take breaks, prioritize self-care, and maintain a balanced lifestyle by suggesting healthy activities outside of academics.

Topics of CSYE 6200 Covered:

- **JavaFX**: Used for creating the user interface.
- **Class Definitions:** For organizing code logically.
- **Inheritance & Polymorphism:** To build flexible and reusable code structures.
- **Abstract Classes & Interfaces:** To define templates for more specific classes.
- **Set and Maps:** For handling relationships between different data points.
- **Lists and Stacks**: Key data structures for managing student data.

Where and how did we use the topics

Lists:

Used to store the list of subjects for each student.

- **Where Used:**
  - **List<Subject> is used to store the subjects associated with a student.**
  - **Example: private Map<String, List<Subject>> studentSubjects.**

- **How Used:**
  - **Stores multiple subjects for each student.**
  - **Dynamically adds or removes subjects when a student adds or removes them.**
  - **Used in loops to display all subjects for a student (e.g., in ViewAllStudentsWindow).**

Stacks:

Used for the undo functionality, tracking the last actions for each student.

- Where Used:
  - Stack<Subject> is used in studentActions to track actions like adding subjects for undo functionality.
  - Example: private Map<String, Stack<Subject>> studentActions.

- How Used:
  - Tracks the most recent action performed by a student (LIFO structure).
  - Supports undo functionality by popping the last action when the “Undo” button is clicked.

Maps :

Used to store and retrieve the relationship between students and their subjects or actions.

• How Used: • Enables flexibility and extensibility. For instance, the provideMentalHealthResources method could be overridden for specific implementations.

• Where Used: • Map<String, List<Subject>> studentSubjects stores subjects for each student. • Map<String, Stack<Subject>> studentActions stores undo actions for each student.

Used to store and retrieve the relationship between students and their subjects or actions.

Classes :

Classes are the building blocks of this program, representing entities such as students, subjects, and windows.

• Where Used:

• Main classes: Main, Student, Subject, and various UI-related classes (ViewSubjectsWindow, ViewGraphWindow).

• How Used:

• Encapsulate behavior and data, such as student details, subject grades, and UI components.

• Support modularization by separating concerns (e.g., subject management, grade checks, UI display).

Abstract Classes :

Abstract classes are base classes that cannot be instantiated and often contain abstract methods to be implemented by derived classes.

- **Where Used:**
- **Though not explicitly shown in the shared code, the use of interfaces or base classes like Subject suggests potential use of abstract classes.**
- **Abstract classes could be used for generic entities like Person, which Student could extend.**
- **How Used:**
- **Provides a base structure for extending entities in the application.**

Inheritance :

Inheritance allows a class to derive properties and methods from another class.

- **Where Used:**
  - **The Subject class can potentially be extended for specific subject types.**
  - **UI classes (ViewGraphWindow, ViewPieChartWindow, etc.) may share a common base class for reusable functionality.**

- **How Used:**
  - **Supports reusability and code organization by allowing derived classes to inherit functionality from base classes.**

Polymorphism :

Polymorphism allows one interface or method to behave differently based on the context.

- **Where Used:**

  - **Polymorphic behavior is seen in the use of interfaces and inheritance.**

  - **Example: MentalHealthResources uses an interface to provide resources in a general way that can be implemented differently if needed.**

- **How Used:**

  - **Enables flexibility and extensibility. For instance, the provideMentalHealthResources method could be overridden for specific implementations.**

Interfaces :

Interfaces define a contract that implementing classes must fulfill, providing method declarations without implementations.

- **Where Used:**
- **Polymorphic behavior is seen in the use of interfaces and inheritance.**
- **Example: MentalHealthResources uses an interface to provide resources in a general way that can be implemented differently if needed.**

- **How Used:**
- **Defines a contract that ensures implementing classes provide specific methods like provideResources().**
- **Enables polymorphism, allowing different implementations of the same interface to be used interchangeably.**

Conclusion:

EduPal is an innovative and holistic solution that bridges the gap between academic performance and mental wellbeing. By integrating grade tracking, personalized support, mental health resources, and work-life balance suggestions, EduPal empowers students to achieve their academic goals while fostering a healthy lifestyle.

Key Achievements:

- **Successfully implemented user-friendly UI with JavaFX.**
- **Utilized advanced programming concepts like lists, stacks, maps, inheritance, polymorphism, and interfaces to ensure efficient functionality.**
- **Provided a scalable solution for tracking grades and mental health through dynamic data structures and modular design.**

Future Scope:

- Leverage AI to provide personalized recommendations for improving academic performance.

- Integrate guided mindfulness and stress management programs into the platform.

- Enhance accessibility by developing mobile and web versions.

Thank you!