

BalanceBoard - A JavaFX Application

Jack Parry-Wingfield

January 2025

Abstract

For as long as education has existed, students have struggled with time management, often overestimating their available time and missing critical academic deadlines. Balancing a busy schedule while knowing when to start each task is challenging, leading to unnecessary setbacks and, for many, an overwhelming cycle of falling behind. However, it doesn't have to be this way. Inspired by my younger sibling's academic hardships in their very first university semester, this application is designed to put an end to time management failures and procrastination, ensuring that completed work gets submitted on time, and further helping students achieve a sustainable work-life balance.

The application provides an intuitive agenda system, allowing users to manage tasks in multiple views, including week, day, and calendar-month modes. Tasks are structured as objects with key attributes like due dates and completion status, enabling efficient organization. To further motivate students, a reward system featuring virtual trophies recognizes task completion and consistency. Additionally, built-in analytics track submission patterns, time taken to start tasks, and other productivity metrics. These insights contribute to a personalized productivity score, providing students with a clear overview of their work habits over daily, weekly, monthly, and yearly periods. Though not initially implemented, AI and machine learning will be used to better cater to each student and their unique work habits, as no two students are exactly the same.

The application is developed using JavaFX for the user interface, offering a responsive and visually intuitive experience for students managing their tasks. The core logic is implemented in Java, utilizing its strong object-oriented paradigm to structure task management, scheduling, and analytics. Data persistence will be handled using SQLite, providing a lightweight yet efficient database solution for storing user tasks, progress, and productivity metrics. In future iterations, machine learning and AI will be integrated to personalize user experiences. This may involve TensorFlow for Java or Weka to analyze study habits and optimize task recommendations based on individual productivity trends. Additional libraries and frameworks, such as JFoenix for enhanced UI components and Quartz Scheduler for task reminders, may also be incorporated to improve functionality and user engagement.

This application is designed for students ranging from high school to university/college, helping them effectively manage their academic workload and deadlines. Recognizing the differences in workflow and academic structure between these levels, a possible feature under consideration is the implementation of two distinct modes: High School Mode and University Mode. This would tailor task organization, reminders, and productivity tracking to better suit the unique challenges faced by students at each stage. While this feature is not yet guaranteed, future iterations may explore its feasibility based on user needs and feedback.

This documentation provides a comprehensive overview of the application, detailing its purpose, design, and technical aspects. It begins with an introduction outlining the app's objectives and the motivation behind its development. The technology stack section describes the key technologies used and their respective roles in building the application. App functionality is then explored, highlighting core features and their significance in enhancing student productivity. Following this, the user interface design section presents the app's visual structure and explains how functionality is balanced with simplicity. The user guide offers detailed instructions on maximizing the application's potential, ensuring users can efficiently navigate and utilize its features. Additionally, data management is discussed, explaining how task-related data is stored, retrieved, and maintained for persistence. Finally, the documentation assesses potential future improvements, including the integration of artificial intelligence and machine learning to further personalize and optimize the user experience.

1 Introduction

The transition from high school to university—and even from elementary or middle school to high school—is a significant leap in both academic workload and expectations. Across all levels of education, falling behind in coursework can feel overwhelming, and once a student is struggling to catch up, it becomes all too easy to miss crucial deadlines and face unnecessary penalties.

This issue is personal to me. My younger brother, who had already faced motivational struggles in high school, encountered a particularly difficult first semester in university. Despite having completed most of his work, a simple lapse in due-date awareness led to multiple missed deadlines across all his courses, resulting in a failing CGPA. However, I knew this outcome did not reflect his abilities. Like many students, he simply lost track of deadlines—a challenge we all face at some point.

After a deep discussion on motivation and organization, he went on to ace his next semester, proving that success is not a matter of ability, but of structure and awareness. This experience reinforced the importance of maintaining an organized schedule and keeping track of tasks effectively—hence, the motivation behind this application.

This application is a comprehensive productivity tool designed to help students effectively manage their academic responsibilities and minimize the impact of procrastination and missed deadlines. At its core, it features an intuitive agenda system that allows users to organize tasks across multiple views, including weekly, daily, and calendar-month modes, catering to different planning preferences. Each task is treated as an object with key attributes, such as a due date and completion status, ensuring structured task management and efficient tracking of upcoming deadlines.

To further engage and motivate students, the app includes a reward system that recognizes consistency and task completion. By earning virtual trophies or achievement badges, students receive positive reinforcement, helping to build better work habits over time. Additionally, the app offers built-in analytics that track key productivity patterns, such as submission trends, time taken to initiate tasks, and overall work consistency. These data points contribute to a personalized productivity score, giving students a clear overview of their performance across various timeframes (daily, weekly, monthly, and yearly).

Future versions of the application will explore the integration of AI-driven insights, which will analyze individual work habits and suggest optimal study schedules based on past behavior. This will allow for personalized recommendations, ensuring that students receive tailored guidance on when and how to structure their workload most effectively. By combining structured task management, motivation-driven rewards, and analytical insights, this application serves as a smart academic companion, empowering students to take control of their productivity and achieve greater success in their studies.

2 Technology Stack

3 App Functionality

3.1 The Core Feature

The core functionality of this application revolves around the agenda system, an enhanced to-do list that allows students to efficiently track assignments, deadlines, and study plans. Tasks are displayed along with their respective due dates and time remaining, helping students visualize upcoming responsibilities. Users can add, remove, and complete tasks, with completed tasks disappearing in a ‘successful completion’ manner. Additionally, tasks can be set as recurring—ideal for assignments like weekly quizzes or discussion posts, ensuring that repeated deadlines are never overlooked.

The agenda will provide multiple viewing options to accommodate different planning styles. Users can switch between a daily view, a weekly view, and a calendar-style month view. To avoid overwhelming the interface, the monthly calendar view will display only hard deadlines and release dates, keeping the display clean and focused.

3.2 Analytics and Productivity Tracking

Beyond basic task management, the application continuously runs analytics in the background to provide meaningful insights into a student’s workflow. While more analytics may be added in the future, the following key metrics will be included in the initial implementation:

- Time elapsed between the release date of a task and when the student starts working on it.

- Time taken between task completion and final submission.
- Weekly and monthly distribution of workload—analyzing whether work is balanced daily or left to accumulate before deadlines.

These insights will contribute to a personalized productivity score, providing students with a clear and measurable depiction of their work habits over weekly, monthly, and potentially even yearly periods. This score aims to encourage self-improvement by helping students recognize tendencies such as procrastination or last-minute cramming.

3.3 Task Prioritization and Smart Scheduling

A dynamic task prioritization system will assist students in structuring their workflow more effectively. By considering assignment weight and due dates, the system will generate a personalized study plan that ensures high-impact assignments receive appropriate attention. Rather than simply listing tasks in chronological order, this feature helps students allocate time based on importance, preventing last-minute rushes and optimizing productivity.

3.4 Gamified Reward System

To further motivate students, a reward system featuring virtual trophies and achievements will recognize task completion and consistency. By gamifying the study process, this system fosters positive reinforcement, keeping students engaged and on track.

3.5 Built-In Grade Calculator

A feature that could prove to be very useful is a built-in grade calculator system, that is automated, pending the user's input grades and grading schemes. I personally have a routine that I execute come finals season where I make a list of all attainable final grades in the class and the corresponding required final exam score, giving me a foolproof look at how I need to perform - it is useful in cases where you need to simply pass, and also useful in cases where you need that A. In any case, it allows for strategic studying rather than relying on guesswork of how you need to perform, which has surely caused some preventable failures in the past.

The built-in grade calculator provides a seamless way for students to track their academic standing and determine required scores for their desired final grade. While not mandatory, this feature is available for users who wish to take advantage of it.

Students can customize their grading setup by adding exams, assignments, quizzes, papers, and other coursework, each with its respective weight. The system also supports grading rule variations, such as:

- **Dropped grades** - many courses offer schemes such that the lowest assignment, quiz, or other item is dropped.
- **Multiple grading schemes** - most courses offer multiple schemes and accept the highest grade.

Behind the scenes, the grade calculator is structured using object-oriented programming (OOP) principles. Each coursework entry (exam, assignment, etc.) is treated as an object, storing its weight and grade as fields. These objects are then processed through a calculation array, efficiently computing weighted averages and final grade projections.

This approach promises that the system remains scalable, efficient, and easy to update, providing students with a clear, stress-free way to stay on top of their grades.

3.6 Future AI/ML Enhancements

While the initial version of this application focuses on structured task management and analytics, future iterations will explore AI and machine learning to further optimize student productivity. Potential AI-driven enhancements include:

- **Personalized task suggestions** – Recommends optimal start times based on known work habits.
- **Smart reminders** – Adaptive notifications that trigger at moments when procrastination is most likely.
- **Predictive analytics** – Forecasts productivity trends and suggests improvements.

These features would allow the application to adapt to individual student behaviors, making it an even more powerful tool for academic success.

4 User Interface Design

4.1 Launching the app for the first time (Setup Screen)

What do we need to know about the user? Certainly not very much - there is no personal information required. We merely seek academic details, such as the number of courses, number of credits per course, grading scheme. When a new user launches the application for the first time, they will be prompted to answer a few short questions, including:

-
-
-

4.2 Launching the app regularly (Welcome Screen)

5 User Guide

6 Data Management

7 Future Improvements & AI Integration

8 Conclusion