

De La Salle University

**Computer Engineering
2nd Sem AY 2023**

Subject/Course: LBYCPEI: Object Oriented Programming		Professor's Name: Mr. Ruiz
Team Hermes Members: DIMAPILIS, David Brian S. FERROLINO, Rudd Anthony C. Ong, Aidan Matthew G.	Section: EQ5	Date accomplished: 6/26/2023

TypeCraft: A Groundbreaking Adaptive Typing Program for Enhancing Digital Literacy and Individualized Learning in the Digital Age

I. Introduction

TypeCraft is a groundbreaking typing program designed to enhance the way students acquire typing skills. Through a blend of gamified learning, adaptive feedback, and tailored lessons, it aims to bolster digital literacy, individualize learning, and integrate seamlessly into existing educational frameworks. Unlike existing applications such as Typing.com, TypingClub, and KeyBlaze, which primarily focus on providing typing lessons and tracking progress, TypeCraft goes a step further. TypeCraft provides a dynamic typing journey, adapting to each learner's skill level and offering personalized feedback to improve accuracy and speed. Its unique feature of customizable lessons allows educators to align typing exercises with curriculum needs, making it a comprehensive tool for boosting computer literacy. This feature sets TypeCraft apart from its competitors, as it empowers educators to tailor the typing program according to their students' needs, enabling seamless integration into existing educational systems. Moreover, TypeCraft's engaging and gamified interface motivates students to continue practicing and progressing. It includes interactive elements, rewards, achievements, and progress tracking, making the learning experience enjoyable and immersive. This level of engagement is a

distinctive aspect of TypeCraft, as it transforms the often monotonous task of learning to type into an exciting and rewarding experience.

Additionally, TypeCraft is designed to optimize the learning process by utilizing efficient techniques that help students learn and improve typing skills in less time. It employs effective exercises, targeted practice, and adaptive learning algorithms to maximize the learning outcome within a shorter duration. This optimization of time is another differentiating factor that gives TypeCraft an edge over similar applications. Suitable for various educational settings, TypeCraft equips students with essential typing skills, paving the way for academic and professional success in the digital age. It's time to unlock the power of typing with TypeCraft.

II. Methodology

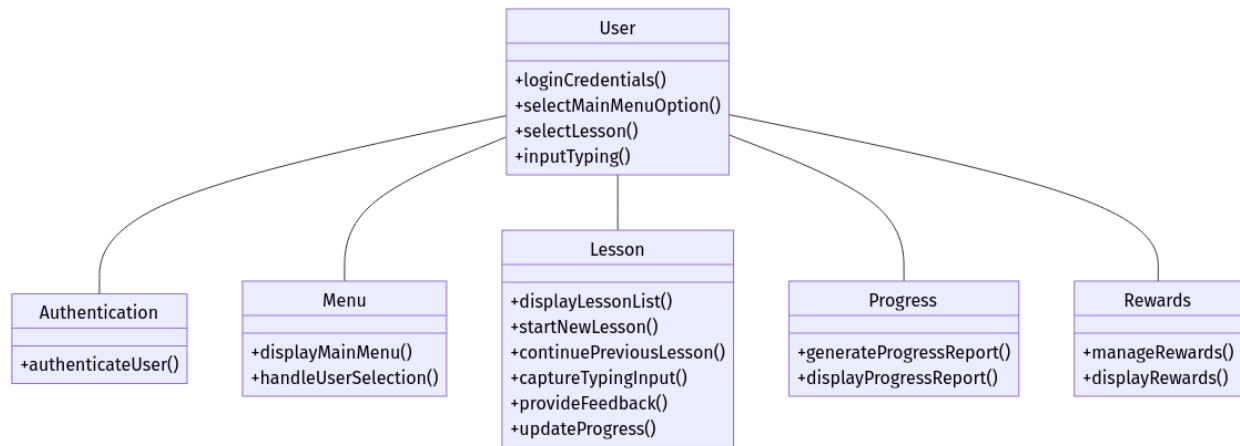
To develop TypeCraft, we aimed to employ a user-centric design approach, focusing on the needs and experiences of the end-users: which is our fellow students and educators. The software will be developed using IntelliJ Java, leveraging its robustness, accessibility, and platform independence. The four pillars of Java - abstraction, encapsulation, inheritance, and polymorphism - will be integral to our development process. Abstraction will be used to hide the complexity of certain operations, encapsulation to protect the data, inheritance to promote code reusability to the mainstream, and polymorphism to allow entities to take on many forms.

The project will be divided into a six-step process:

1. *Requirement Gathering and Analysis*: Understand the needs of the users and define the system requirements.
2. *Design*: Develop the system architecture, including the user interface and database design.
3. *Implementation*: Code the system functionalities using Java.
4. *Testing*: Test the system for bugs and fix them.
5. *Deployment*: Deploy the system for use in educational institutions once approved.
6. *Maintenance*: Provide ongoing support and updates.

III. Project Description

Class Diagram:



IPO: (Input-Process-Output)

Input:

- User login credentials
- User selection from the main menu (start a new lesson, continue a previous lesson, view progress)
- User selection of a specific lesson
- User's typing input during the lesson

Process:

- User authentication
- Display of main menu and handling of user selection
- If a new lesson is started, the program displays a list of lessons and starts the selected lesson
- If a previous lesson is continued, the program displays a list of incomplete lessons and resumes the selected lesson from where it was left off
- If progress is viewed, the program generates and displays a progress report
- During a lesson, the program captures the user's typing input, provides real-time feedback, and updates progress upon lesson completion
- The program also manages and displays rewards and achievements based on the user's progress

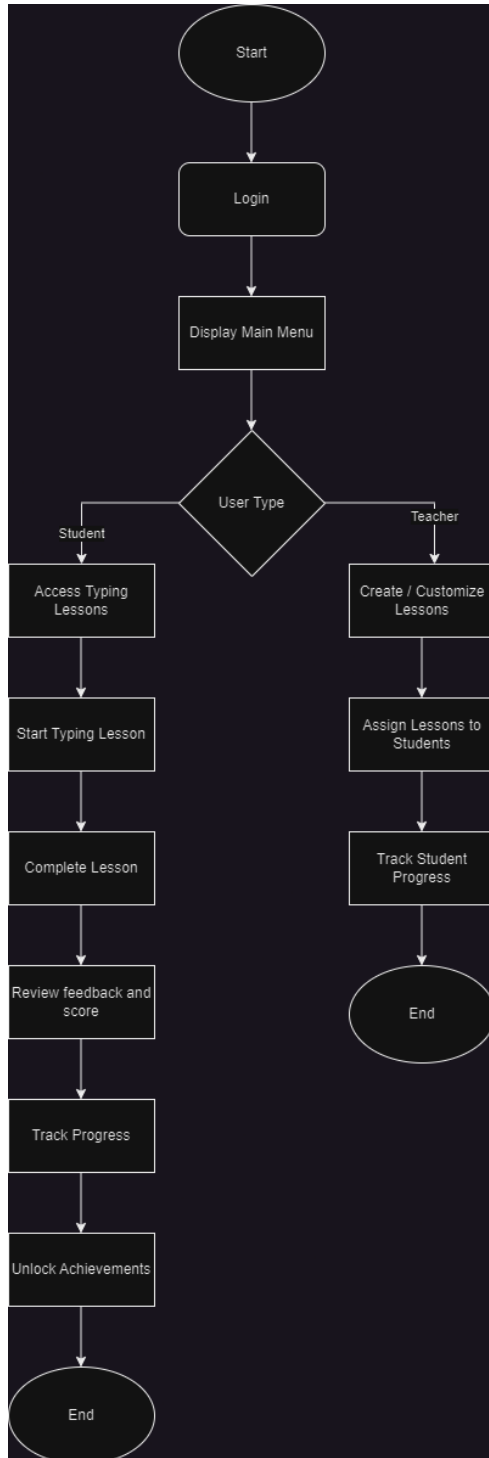
Output:

- Main menu interface
- List of lessons (new or incomplete)
- Typing interface for the selected lesson
- Real-time feedback during the lesson
- Updated progress, rewards, and achievements upon lesson completion
- Progress report when the user chooses to view progress

Description of IPO:

This current IPO is what Team Hermes expects the program to do, it also showcases the major components of the program and how they interact, but the actual implementation would involve more detailed design and coding, which will be done later on.

Flowchart;



Description of Flowchart:

The flowchart provides a comprehensive illustration of the user journey within the TypeCraft educational typing system. The journey commences with the user logging into the system, a crucial step that ensures security and personalization of the user experience. Upon successful login, the system presents the main menu to the user, serving as the central navigation hub for the entire program. At this step, the system identifies the user type, bifurcating the journey into two distinct paths: one for students and another for teachers. This decision point is pivotal as it tailors the subsequent steps to the specific needs and roles of the user.

For students, the journey is primarily focused on learning and skill development. They first access the typing lessons available to them. Once a lesson is selected, they embark on the learning experience, starting and completing the typing lesson. Upon completion, the system provides immediate feedback and a score, offering insights into their performance. This real-time feedback mechanism is instrumental in fostering a continuous learning environment. The students then proceed to track their progress, a feature that allows them to monitor their improvement over time and set personal goals. The journey for the student culminates with the unlocking of achievements, a gamified element that adds an element of fun and motivation to the learning process.

On the other hand, the teacher's journey is centered around lesson management and student progress tracking. Teachers have the ability to create or customize lessons, enabling them to tailor the learning content according to their curriculum or specific requirements. Once the lessons are prepared, they assign these to their students, initiating the learning process. A significant part of the teacher's journey is tracking the progress of their students, a feature that allows them to monitor student performance, identify areas of improvement, and provide targeted support.

PUML:

@startuml

actor User

participant "Typing Game" as Game

participant "Performance Evaluator" as Evaluator

User -> Game: Start Game
activate Game

loop Game Session

 User -> Game: Type Characters

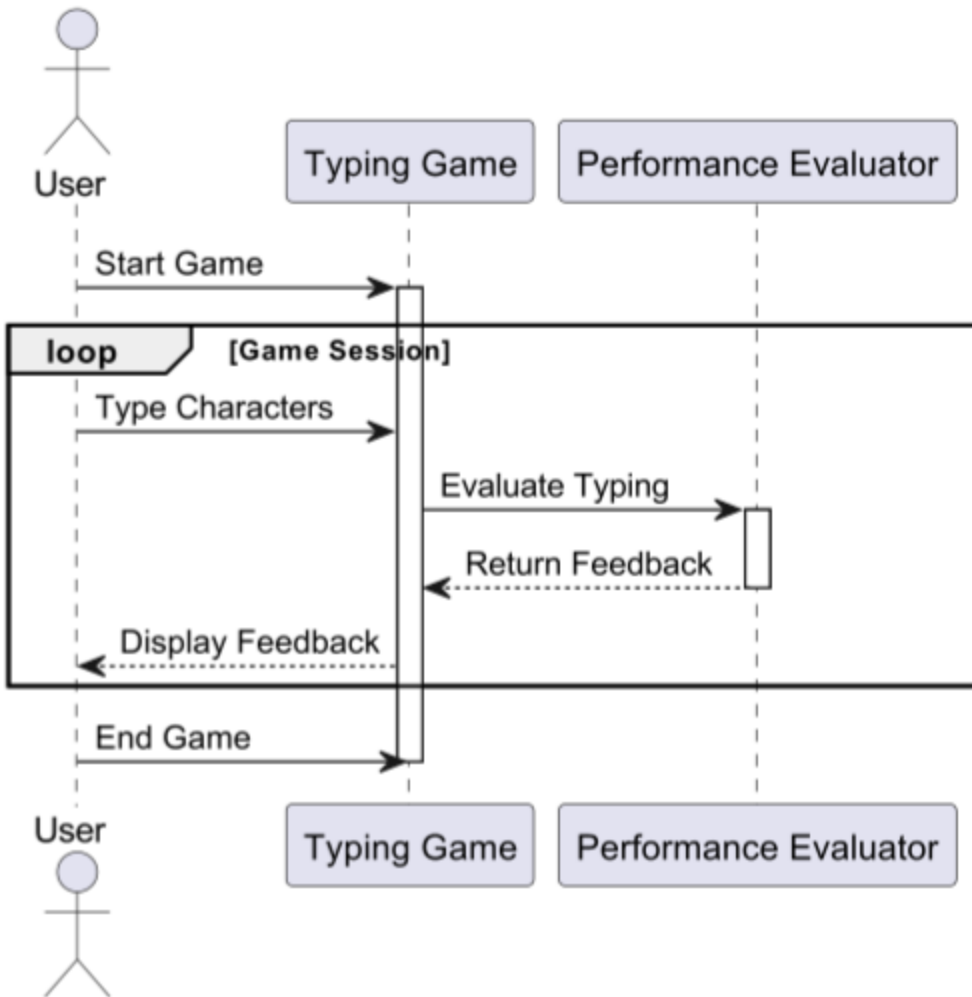
 Game -> Evaluator: Evaluate Typing
 activate Evaluator

 Evaluator --> Game: Return Feedback
 deactivate Evaluator

 Game --> User: Display Feedback
end

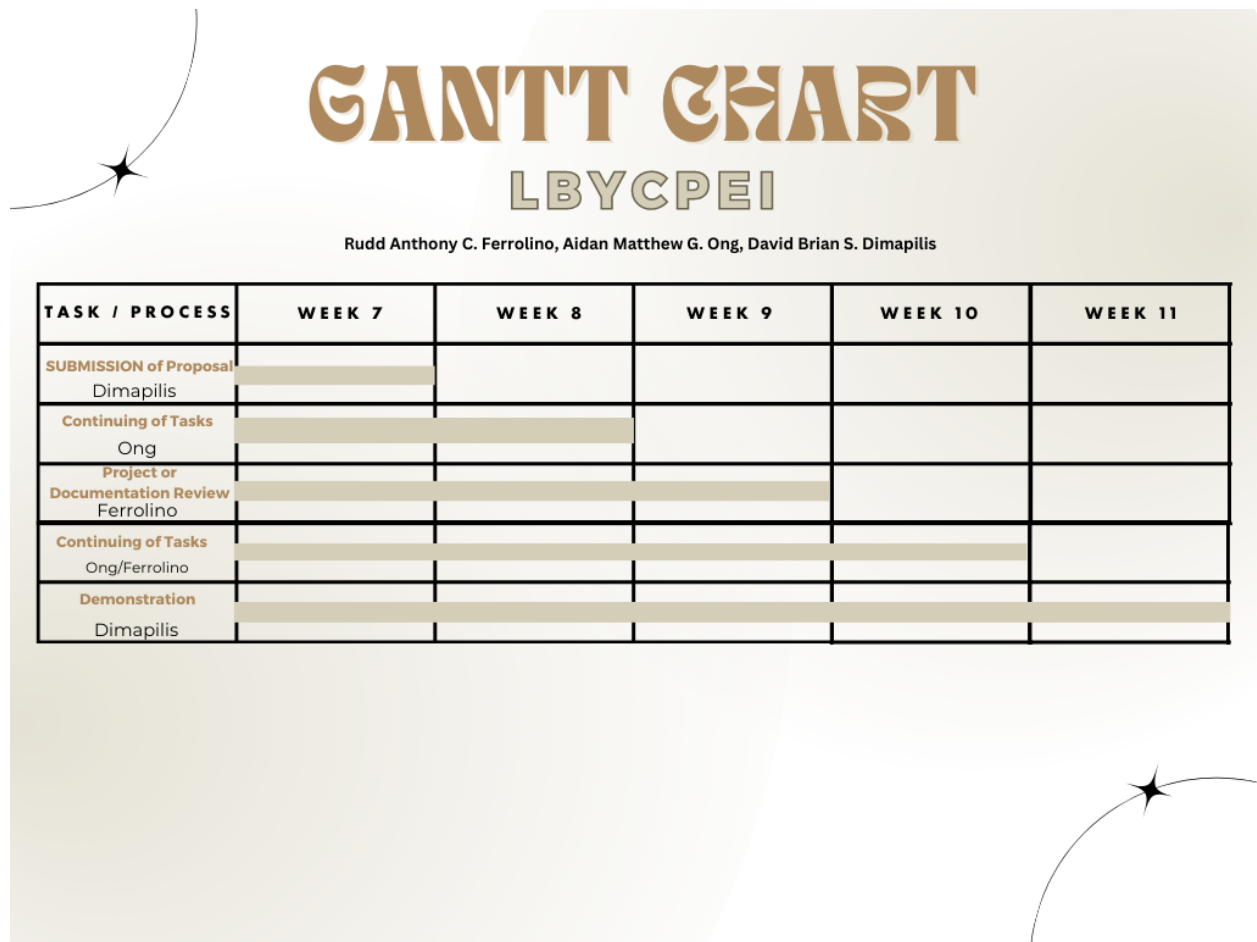
User -> Game: End Game
deactivate Game

@enduml



IV. Deliverables

GANTT CHART:



V. Evaluation

User Interface and Experience: The design of the interface should be intuitive and easy to navigate, even for users with little to no experience with similar software. It should also be engaging to maintain user interest.

Adaptability and Customizability: The program should have the ability to adapt to a user's skill level and provide customizable lessons. It should be evaluated on how well it caters to different user needs and learning styles.

Learning Effectiveness: This is one of the most important criteria. It measures how effective the typing program is in improving the typing skills of the students. Metrics like improvement in typing speed and accuracy over time can be used.

Feedback and Performance Tracking: The program should provide clear, specific, and constructive feedback on the user's performance. It should also track user performance over time to show progress.

Integration with Existing Systems: If the typing program is being used in an educational institution, it needs to be compatible with the existing systems in the institution.

Reliability and Stability: Since it's a local program, the stability of the software is important. It should not crash frequently, and should be able to recover gracefully in case of any failures.

VI. Conclusion

The TypeCraft project is a significant initiative to enhance digital literacy by creating an engaging and adaptive educational typing program. Using gamification and personalized feedback, it facilitates individualized learning and fosters typing mastery, filling a gap in many educational institutions that lack an engaging and effective platform for teaching these essential skills. Notably, its local, non-online nature broadens its accessibility, making it an invaluable tool for schools or individuals with limited internet access. Overall, TypeCraft offers a comprehensive solution for computer literacy that supports academic success and prepares students for future professional endeavors in a digital-centric world.

VII. References

Typing Game for Java. GitHub. (n.d.). <https://github.com/topics/typing-game?l=java>

Codementor. (n.d.). *Speed typing game.* DevProjects.
<https://www.codementor.io/projects/web/speed-typing-game-c51led1afn>

GeeksforGeeks. (2022, August 1). *Design a typing speed test game using JavaScript*.
GeeksforGeeks.

<https://www.geeksforgeeks.org/design-a-typing-speed-test-game-using-javascript/>