

Introduction

Overview

The world is complex and constantly evolving, education is the key to understanding and thriving within it. Educational games are innovative tools that can be utilized to combine the thrill of gaming while providing learning benefits. Digital media provides an unprecedented opportunity to meld play with pedagogy, creating experiences that educate, engage, and entertain. Educational games are designed to entertain players and impart knowledge and skills. The design of Math Match-Up is based on a rather simple interface as its audience is younger children/adults. Math Match-Up is made accessible to all hence simplifying the layout, having interactive elements and providing easy-to-follow steps.

The objective of this project is to create a game that teaches a lesson, life skill, basic ability, or concept in a way that resonates with the intended audience, in this case, a young child. In designing this game, the goal is to create an experience that is not only educational but also enjoyable and memorable. The design includes progressive difficulty, which engages the audience/player, as well as immediate feedback so players can reflect upon their mistakes after the game.

Objectives

This project is designed to provide experience in:

- applying principles of software engineering to a real-world problem.
- working with, interpreting, and following a detailed specification provided.
- creating an interface easy to follow and using designs from the provided specifications.
- incorporating an adaptive learning design
- implementing the design in Java using JSON files.
- working through the design process.
- creating graphical, user-facing content and applications.
- writing robust, clean, and efficient code.
- writing good, well-documented Java code.
- reflecting on good/bad design decisions made throughout this project.

This project is intended to provide a lot of freedom in designing and creating the application while still providing solid direction toward a specified goal.

References

A list of references to other documents that might provide context or otherwise assist in the understanding of this document.

- The original project specification: <https://owl.uwo.ca/access/content/group/aa2311c9-4cef-497f-8d9c-b502023be21c/project/CS2212%20Group%20Project%20Specification.pdf>
- Java Swing: [https://en.wikipedia.org/wiki/Swing_\(Java\)](https://en.wikipedia.org/wiki/Swing_(Java))
- Java FX: <https://openjfx.io/>
- Balsamiq Wireframe: <https://balsamiq.com/wireframes/>
- Downloading a Bitbucket repository: <https://confluence.atlassian.com/bitbucketserver/download-a-repository-archive-913477030.html>
- Activity Diagrams: https://en.wikipedia.org/wiki/Activity_diagram
- Use Case Diagrams: https://en.wikipedia.org/wiki/Use_case_diagram
- Education Games Examples: <https://www.coolmathgames.com/>
- Jackson Library: <https://github.com/FasterXML/jackson>
- BitBucket: <https://bitbucket.org/>
- GitHub: <https://github.com/>
- IntelliJ Idea: <https://www.jetbrains.com/idea/>
- UML Class Diagram: https://en.wikipedia.org/wiki/Class_diagram
- Jackson Library: <https://www.javatpoint.com/jackson#:~:text=Jackson%20is%20a%20high%2Dperformance,concepts%20of%20the%20Jackson%20library.>
- Apache Maven: <https://maven.apache.org/>