**ChessEDU
Iteration Plan <Iteration 1>**

Version <1.1>

Revision History

Date	Version	Description	Author
1/11/2022	1.0	First Draft	Rylan DeGarmo
4/11/2022	1.1		Adair Torres

Table of Contents

Iteration Plan <Iteration ID>

1. Introduction

1.1 Purpose

In this first iteration, the team's mission is to develop an initial version of the ChessEDU system. All of ChessEDU's core functionalities must be implemented and operational. All graphical user interfaces must be able to fulfill their basic functionalities.

1.2 Scope

This plan targets tasks and activities assignments especially in relation with artifacts production and code breakdown. This plan offers a vision on how these tasks and activities will be assigned among team members and what roles are involved during this iteration:

Implementer(s)

The implementer(s) participate to the following activities:

- Implement all graphical user interfaces for the web browser pages rendered by Flask.

- Implement credentials, user data, and course file archive databases.

- Implement REST API using Flask to handle HTTP request methods and render templates.

- Implement Manager objects that take in requests to retrieve and process entries from their respective databases and systems.

Related Artifacts are: Code (Build)

Integrator

The Integrator participates to the following activities: Integrate the system and subsystems (Flask REST API, Manager objects, and databases).

Tester

The tester participates to the following activities: Plan tests for system functionalities and for all graphical interface(s) interactions.

Project Manager

The project manager participates to the following activities: Plan phases and iterations, develop iteration plan, schedule and assign work.

Related Artifacts are: Iteration Plan

1.3 Definitions, Acronyms, and Abbreviations

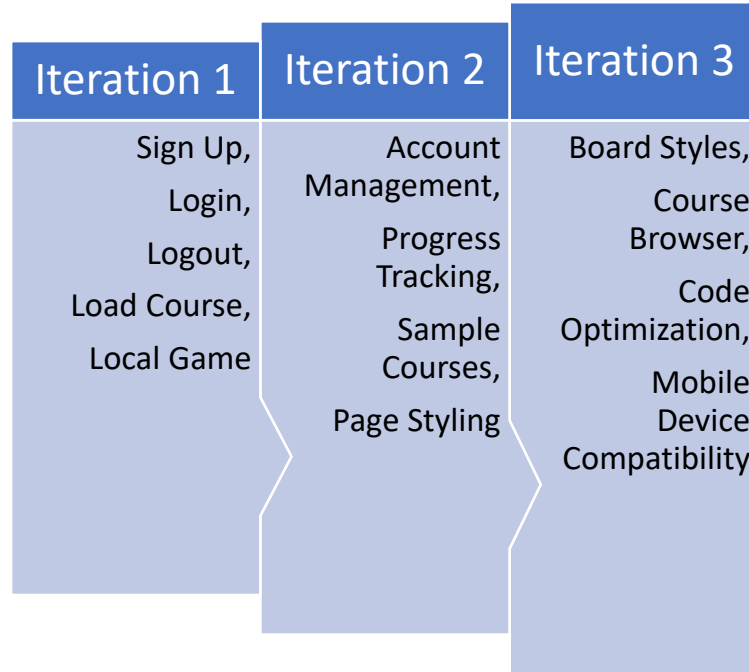Refer to the Glossary Document (See References)

1.4 References

- Glossary Document, Glossary, LearningEDU, 2022

- The course web page https://people.eecs.ku.edu/~saiedian/Teaching/448/

1.5 Overview

This document presents the planning for the iteration and all resources needed.

2. Plan

Iteration 1	Iteration 2	Iteration 3
Sign Up, Login, Logout, Load Course, Local Game	Account Management, Progress Tracking, Sample Courses, Page Styling	Board Styles, Course Browser, Code Optimization, Mobile Device Compatibility

3. Resources

3.1 Human Resources

- *Project's Team:* Adair Torres, Chinh Nguyen, Jack Reynolds, Grant Jones, Rylan DeGarmo

- All course members: professors and lab assistant.

3.2 Software Resources

- Microsoft Office 365 Student

- Google Chrome

- Mozilla Firefox

3.3 Hardware Resources

- The course labs.

- Personal computers.

4. Use Cases

Iteration-Related Use-Cases:

- Sign Up

- Login

- Logout

- Load Course

- Local Game

5. Evaluation Criteria

- Functionality:
    - HTTP requests must be properly handled based on validity.
    - Pages must successfully load across different web browsers, including but not limited to, Mozilla Firefox and Google Chrome.
    - Interactions with databases must keep them consistent.
    - System functionalities are compartmentalized into microservices with a Manager object for each service.
    - Users must be able to sign up, login, and logout without error. Invalid methods of logging in or signing up, such as creating an account with credentials already in use or accessing an account with invalid credentials, must be properly handled.
    - A base template for a course's page must successfully load.
    - A base template for a local game's page must successfully load. In addition, a full game of chess should be playable through this page.
- Performance:
    - Web pages must load on all test client devices within a timeframe of 10 seconds.
    - Small volumes of requests do not cause services to slow dramatically or stop.
    - Minimal amounts of data should be collected from the user outside of their credentials and progress.
    - The system must operate within 8GB of RAM and 100GB of storage memory.