ChessEDU

s

Version <1.1>

Revision History

|  |  |  |  |
| --- | --- | --- | --- |
| **Date** | **Version** | **Description** | **Author** |
| 31/10/22 | 1.0 | First Draft | Grant Jones |
| 2/11/22 | 1.1 | Added Initial Use Cases. | Adair Torres |
|  |  |  |  |

Table of Contents

1. Introduction 4

1.1 Purpose 4

1.2 Scope 4

1.3 Definitions, Acronyms, and Abbreviations 4

1.4 References 4

1.5 Overview 4

2. <Use-Case Name One> 4

2.1 Flow of Events - Design 4

2.2 Interaction Diagrams 4

2.2.1 Sequence Diagrams 4

2.2.2 Collaboration Diagrams 4

2.2.3 Participating objects 4

2.3 Class Diagrams 5

2.4 Derived Requirements 5

3. <Use-Case Name Two> 5

# Introduction

## Purpose

This document grants a detailed overview of the system through the use of several diagrams representing the system functions.

## Scope

ChessEDU will allow a user to learn and improve at the game of chess at their own pace by providing an interface to view courses and modules. A user’s progress through a module will be tracked and saved in order for a user to resume a lesson from where they last left off. This Use-Case Realization document provides an overview of the use cases developed in ChessEDU.

## Definitions, Acronyms, and Abbreviations

See Glossary, document chessedu\_gloss.pdf

## References

1. ChessEDU – Glossary

2. ChessEDU – Use-Case Specifications

3. ChessEDU – Supplementary Specifications

## Overview

The sections of the Use-Case Realization document describes use cases in terms of their flow of events, participant objects, and corresponding diagrams.

# USE CASE <User Sign Up>

## Flow of Events - Design

[A textual description of how the use case is realized in terms of collaborating objects. Its main purpose is to summarize the diagrams connected to the use case and to explain how they are related.]

## Interaction Diagrams

[The diagrams connected to the use case.]

### Sequence Diagrams

### Collaboration Diagrams

### Participating objects

[Objects participating in interaction diagrams of the use-case realization. A textual description of the collaborating objects related to a specific use case]

|  |  |  |
| --- | --- | --- |
| **Object** | **Class** | **Description** |
|  |  |  |
|  |  |  |

## Class Diagrams

[The diagrams connected to the use case.]

## Derived Requirements

[A textual description that collects all requirements, such as non-functional requirements, on the use-case realizations not considered in the design model, but that need to be taken care of during implementation.]

# USE CASE <USER LOGIN>

## Flow of Events - Design

[A textual description of how the use case is realized in terms of collaborating objects. Its main purpose is to summarize the diagrams connected to the use case and to explain how they are related.]

## Interaction Diagrams

[The diagrams connected to the use case.]

### Sequence Diagrams

### Collaboration Diagrams

### Participating objects

[Objects participating in interaction diagrams of the use-case realization. A textual description of the collaborating objects related to a specific use case]

|  |  |  |
| --- | --- | --- |
| **Object** | **Class** | **Description** |
|  |  |  |
|  |  |  |

## Class Diagrams

[The diagrams connected to the use case.]

## Derived Requirements

[A textual description that collects all requirements, such as non-functional requirements, on the use-case realizations not considered in the design model, but that need to be taken care of during implementation.]

# USE CASE <USER LOGOUT>

## Flow of Events - Design

[A textual description of how the use case is realized in terms of collaborating objects. Its main purpose is to summarize the diagrams connected to the use case and to explain how they are related.]

## Interaction Diagrams

[The diagrams connected to the use case.]

### Sequence Diagrams

### Collaboration Diagrams

### Participating objects

[Objects participating in interaction diagrams of the use-case realization. A textual description of the collaborating objects related to a specific use case]

|  |  |  |
| --- | --- | --- |
| **Object** | **Class** | **Description** |
|  |  |  |
|  |  |  |

## Class Diagrams

[The diagrams connected to the use case.]

## Derived Requirements

[A textual description that collects all requirements, such as non-functional requirements, on the use-case realizations not considered in the design model, but that need to be taken care of during implementation.]

# USE CASE <Load Course>

## Flow of Events - Design

[A textual description of how the use case is realized in terms of collaborating objects. Its main purpose is to summarize the diagrams connected to the use case and to explain how they are related.]

## Interaction Diagrams

[The diagrams connected to the use case.]

### Sequence Diagrams

### Collaboration Diagrams

### Participating objects

[Objects participating in interaction diagrams of the use-case realization. A textual description of the collaborating objects related to a specific use case]

|  |  |  |
| --- | --- | --- |
| **Object** | **Class** | **Description** |
|  |  |  |
|  |  |  |

## Class Diagrams

[The diagrams connected to the use case.]

## Derived Requirements

[A textual description that collects all requirements, such as non-functional requirements, on the use-case realizations not considered in the design model, but that need to be taken care of during implementation.]