**Software Design  
Document**

for

Project 1 – Chess Game

Version 1.0 approved

Prepared by Madison Kuyawa, William Widmer, Bunrith Seng, Robert (James) Castleberry, Scott Flolid

CS3398

February 20, 2018

# Table of Contents

Table of Contents i

1. Introduction 1

1.1 Purpose 1

1.2 System Overview 1

1.3 Definitions, Acronyms and Abbreviations 1

1.4 Supporting Materials 1

2. Architecture 1

2.1 Overview 1

2.2 Component 1..N 1

3. High-Level Design 2

3.1 Sequence Diagram for pieces 2

3.2 Use case diagram 2

3.3 Sequence diagram for when a piece is captured 2

3.4 state diagram for settings/move history panel 2

3.5 sequence diagram for start button 2

3.6 sequence diagram for ending a game

# Introduction

## Purpose

To design and implement a game of chess that a user can play and use against another player or against an AI.

## System Overview

Our system will be implemented in java using javafx and fxml to build our Graphical User Interface(GUI). Our system will use a controller class to handle all user actions with in the GUI and to implement certain functionalities that we create in scene builder. Our system will also make use of other classes including classes for the pieces and game board.

## Definitions, Acronyms and Abbreviations

GUI – Graphical User Interface

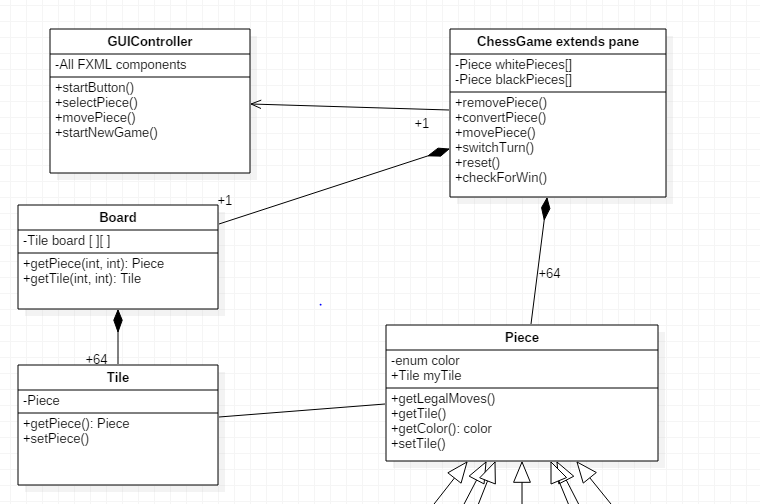
PvP – Player vs. player

PvAI – Player vs artificial intelligence.

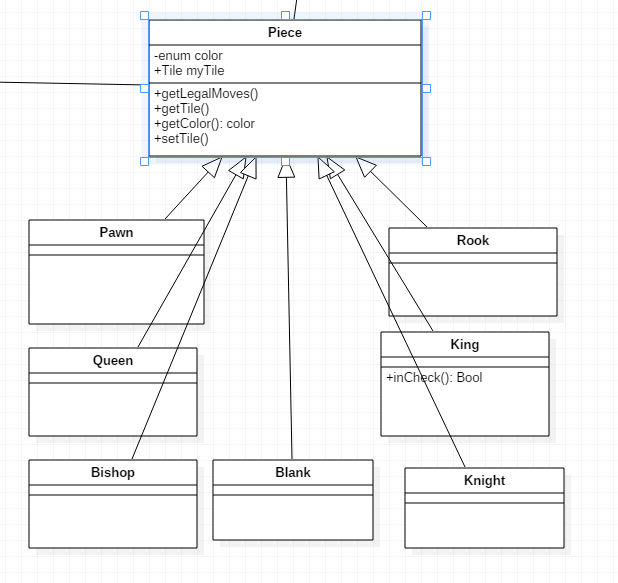
# Architecture

<The architecture provides the top level design view of a system and provides a basis for more detailed design work. This is the section where you should include your High-Level design Component Diagram.

Brief description and the class diagram



Additional view of the Piece class of the class Diagram:



# Overview

<This section provides a high level overview of the structural and functional decomposition of the system. Focus on how and why the system was decomposed in a particular way rather than on details of the particular components. Include information on the major responsibilities and roles that the system (or portions of it) must play.

Why we separated certain classes from our class diagram

# Component 1..n

<Describe an element (subsystem, component, etc...) from architecture in further detail. When appropriate, include information on how the element is further broken down and the interactions and relationships between these subcomponents.

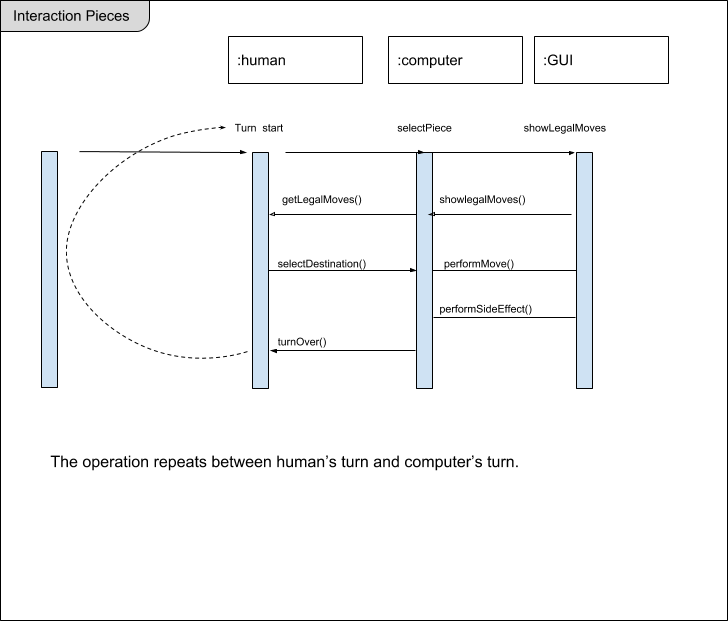
Talk about each class

# High-Level Design

<This section describes in further detail elements discussed in the Architecture. Normally this section would be split into separate documents for different areas of the design.

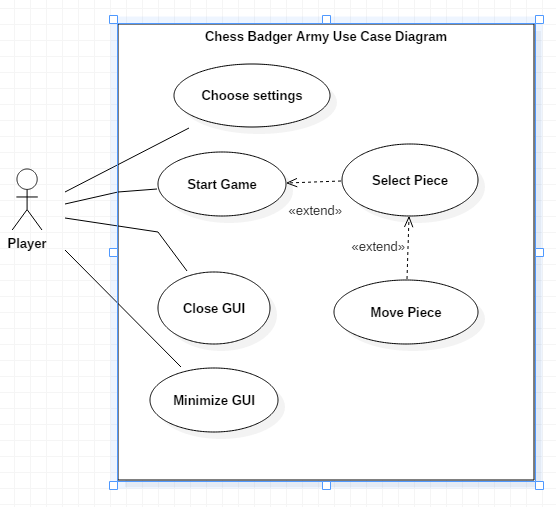
High-level designs are most effective if they attempt to model groups of system elements from a number of different views.

## Sequence Diagram for Pieces



## Use Case Diagram

This Use Case Diagram goes over al of the various interactions that a player will be able to make with the System.



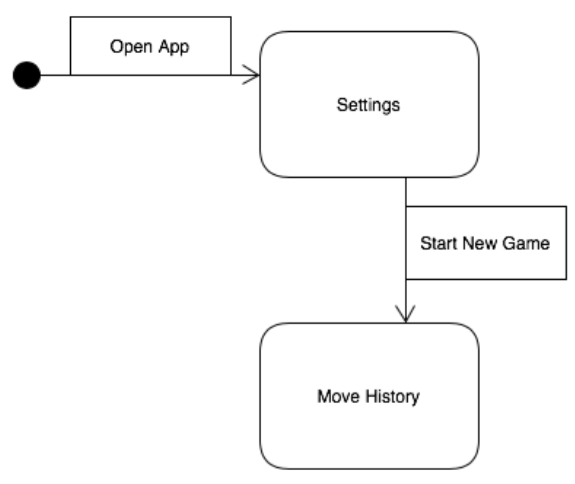
## Sequence Diagram for When a Piece is Captured

**\*\*William’s section**

<Provide a description and diagrams of a system component or set of components that describes a clearly defined view or model of the entire system or a subset of the system.

## State Machine for the Settings/Move History Panel

The following state machine diagram displays the state of the side panel. There are two states that the Settings/Move History panel may have: ‘Settings’ or ‘Move History’.



## Sequence Diagram for Start Button

**\*\*James Section 1**

<Provide a description and diagrams of a system component or set of components that describes a clearly defined view or model of the entire system or a subset of the system.

## Sequence Diagram for Ending a Game

**\*\*James Section 2**

<Provide a description and diagrams of a system component or set of components that describes a clearly defined view or model of the entire system or a subset of the system.