

# Software Design Description

## **Windmill Software**

Steven Cornella  
Armon Entezari  
Dustin Ingram  
Aaron Rosenfeld  
Michael Vadovszki

March 14, 2010  
Revision 0

# Contents

<b>1</b>	<b>Introduction</b>	<b>1</b>
1.1	Purpose . . . . .	1
1.2	Scope . . . . .	1
1.3	Definition, Acronyms, and Abbreviations . . . . .	1
<b>2</b>	<b>Static Design Structure Modeling</b>	<b>2</b>
<b>3</b>	<b>Dynamic Behavior Modeling</b>	<b>2</b>
3.1	Start New Game . . . . .	2
3.2	Wait for Opponent Connection . . . . .	3
3.3	Connect to Opponent . . . . .	3
3.4	Move Piece . . . . .	3
3.5	Wait for Opponent Move . . . . .	4
3.6	Game End . . . . .	4
3.7	Change Game Icons . . . . .	5
<b>4</b>	<b>Requirement Traceability Table</b>	<b>5</b>

# 1 Introduction

## 1.1 Purpose

The purpose of this document is to explain all necessary requirements for Windmill's chess software in the IEEE standard. This include functional, non-functional, constraints, use case specifications, and graphical prototypes of our chess software.

## 1.2 Scope

This document is the initial version (1.0) of the requirements specifications for Windmill's chess software. Software developers and testers are the intended readers of this document.

## 1.3 Definition, Acronyms, and Abbreviations

**Bishop** One of six chess pieces in a chess game. It is permitted to only move in a diagonal direction on a chessboard.

**Checkmate** A chess game state wherein one player's king will be inevitably captured.

**Chess** A board game played between two players on a chess board with sixty four squares. Each square may hold only one chess piece.

**GUI** Acronym for Graphical User Interface. It provides a graphical front end for computer programs.

**Internet** A large worldwide system of connected networks.

**IP Address** Unique 32-bit number that identifies any computer connected to the Internet.

**Port** A 16-bit number that indicates a communication channel on a specific machine.

**Java** An object oriented computer programming language.

**King** One of six chess pieces in a chess game. Each player receives one king at the start of the chess game. It can move in any direction on a chessboard. The objective in a chess game is to protect it from capture by the opponent.

**Knight** One of six chess pieces in a chess game. Each player receives two knights at the start of the chess game. It moves in two possible unique ways, two horizontally and one vertically or vice versa.

**Queen** One of six chess pieces in a chess game. Each player receives one queen at the start of the chess game. In a chess game, it can move in any direction.

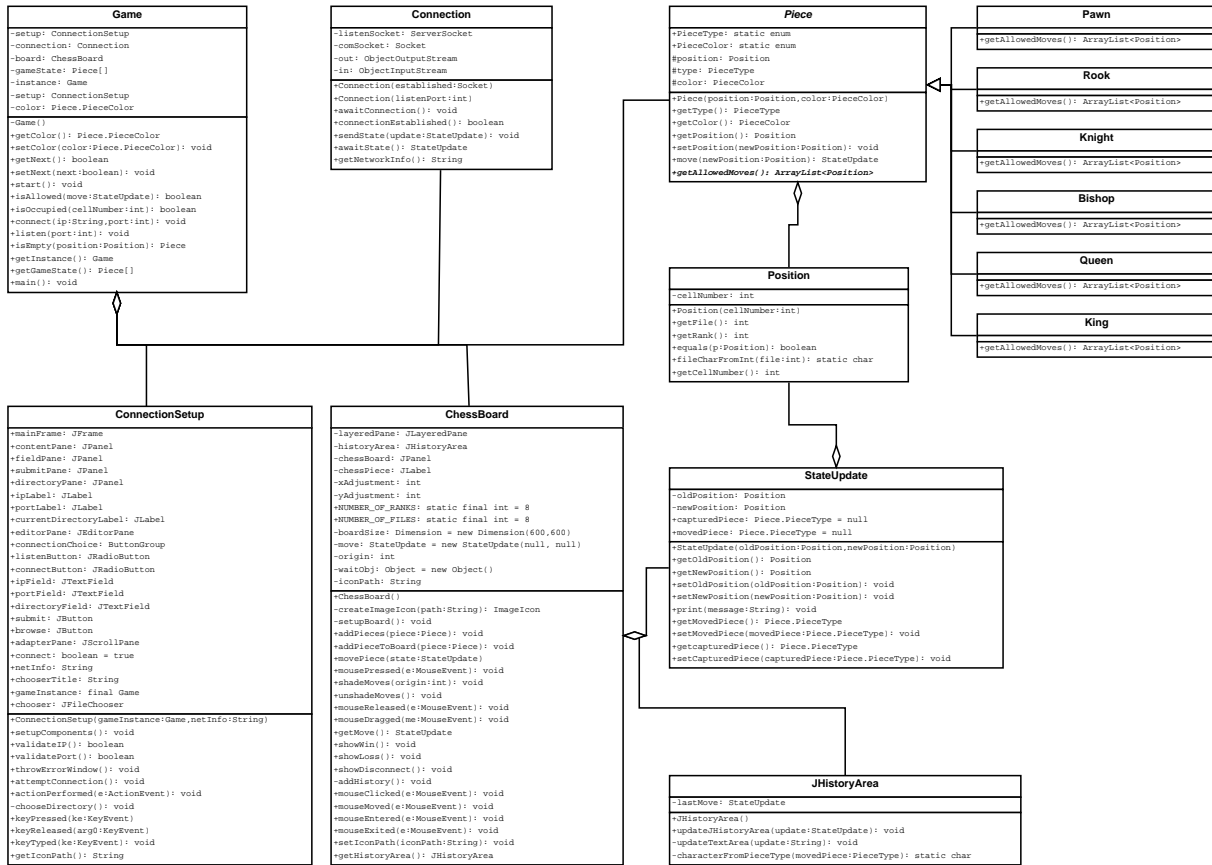
**Rook** One of six chess pieces in a chess game. Each player receives two rooks at the start of the chess game. It can move either horizontally or vertically any number of squares.

**Swing** An API library for providing a GUI to Java programs.

**Thread** Code that runs within the address space of a single process.

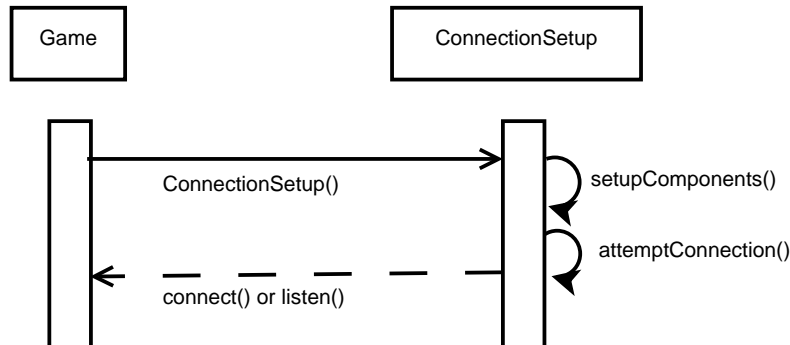
## 2 Static Design Structure Modeling

For a larger version, please see the included file “class-diagram.pdf”.

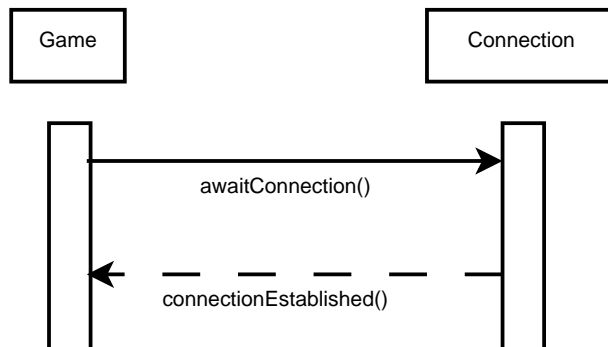


## 3 Dynamic Behavior Modeling

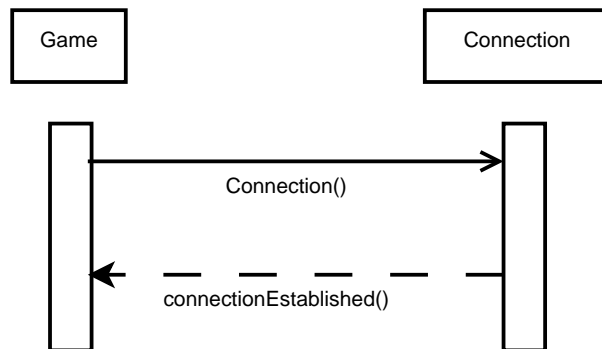
### 3.1 Start New Game



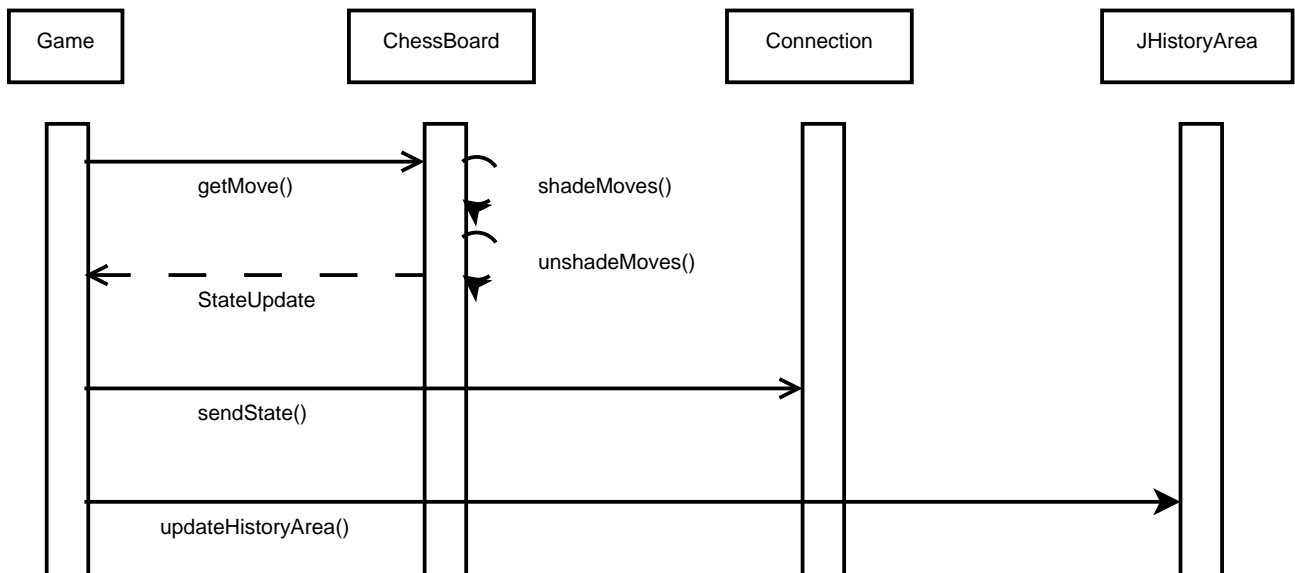
### 3.2 Wait for Opponent Connection



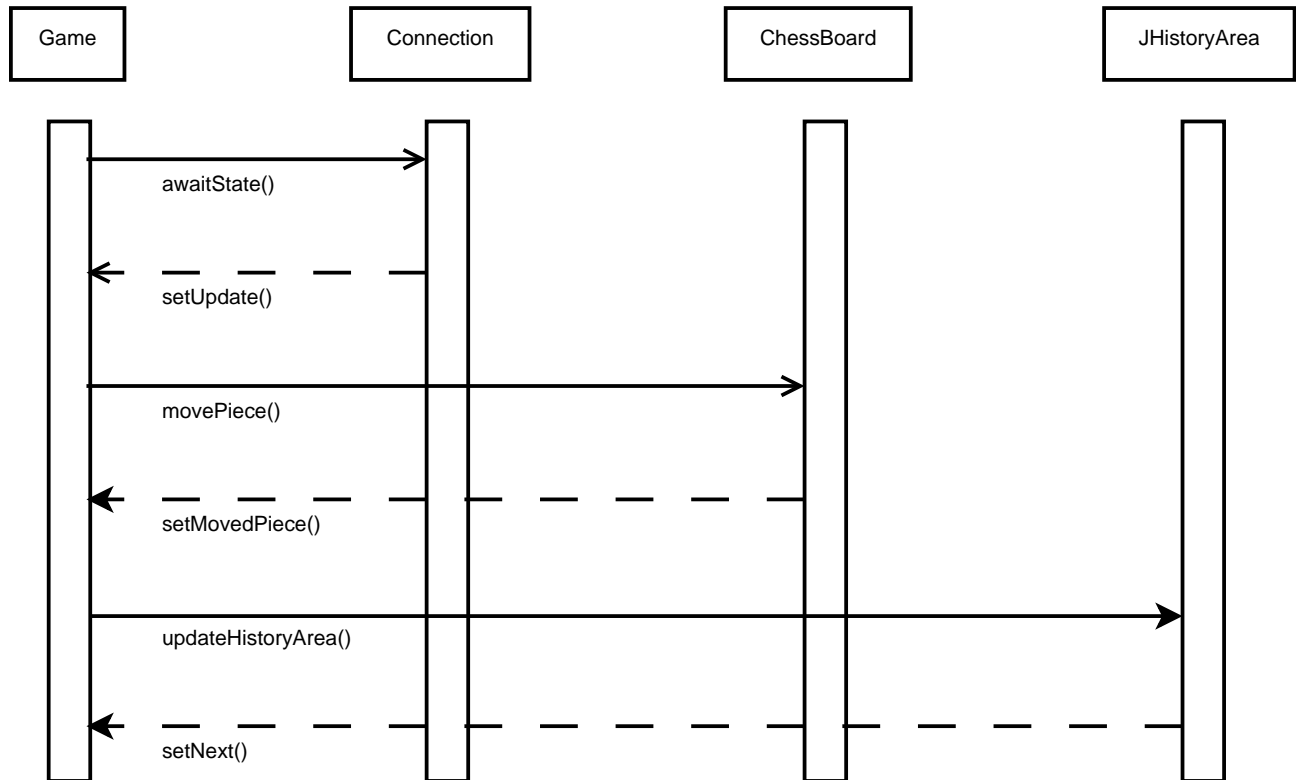
### 3.3 Connect to Opponent



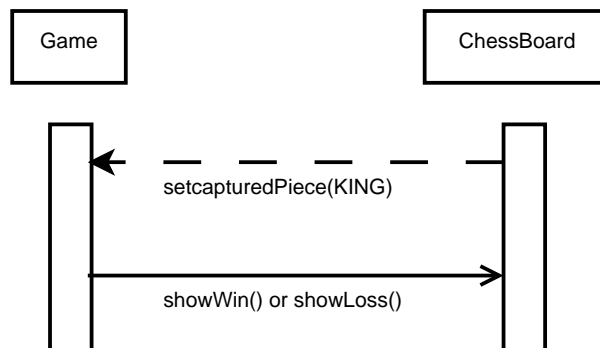
### 3.4 Move Piece



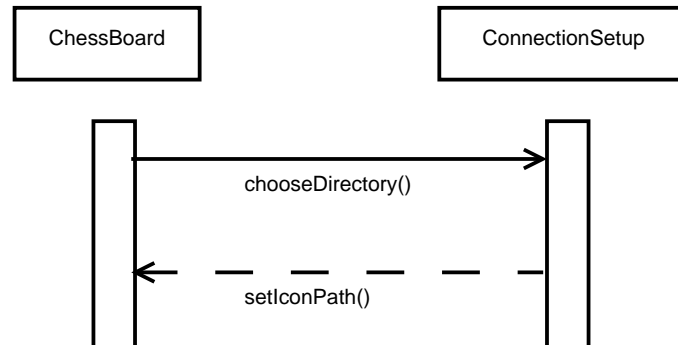
### 3.5 Wait for Opponent Move



### 3.6 Game End



### 3.7 Change Game Icons



## 4 Requirement Traceability Table

Requirement	Component
Handle special chess moves	3.4
Modify piece icons	3.7
Log moves in algebraic notation	3.4
Notify when game has ended	3.6
Specify a specific IP/port for game-play	3.3