DocNo: 001.D.1:1

Software Architecture Document

Version 1.2

**Group Member:**

施宇

鲁皓

万成城

曹翼丰

**Document Language:**

English

Revision History

|  |  |  |  |
| --- | --- | --- | --- |
| **Date** | **Version** | **Description** | **Author** |
| 2015-11-19 | <1.0> | Finish the main structure of the document | 万成城 |
| 2015-12-20 | <1.1> | Fix faults | 万成城 |
| 2016-01-05 | <1.2> | Final fix | 万成城 |

**Key Word**

Chinese chess, architecture, design

**Digest**

This document is to describe the software architecture of Chinese chess game. It is the guideline of our Chinese chess game developing. This document describes the detailed design idea with enterprise architect design model.

**Table of Contents**

[1. Introduction 4](#_Toc439755557)

[1.1 Purpose 4](#_Toc439755558)

[1.2 Scope 4](#_Toc439755559)

[1.3 References 4](#_Toc439755560)

[2. Architectural Representation 4](#_Toc439755561)

[3. Use-Case View 4](#_Toc439755562)

[3.1 Architecturally Significant Use Cases 5](#_Toc439755563)

[4. Logical View 6](#_Toc439755564)

[4.1 Overview 6](#_Toc439755565)

[4.2 Architecturally Significant Model Elements: 6](#_Toc439755566)

[5. Process View 7](#_Toc439755567)

[6. Deployment View 7](#_Toc439755568)

[7. Implementation View 8](#_Toc439755569)

# Introduction

## Purpose

This document provides a comprehensive architectural overview of the Chinese chess game. It depicts different aspects of the system with different architectural views. It intends to capture and convey the significant architectural decisions.

## Scope

This document should be an overview of the architecture and how it should be modeled. Decisions in this document affect how the system is modeled.

## References

1. User Case Specification\_3D Effect.doc
2. User Case Specification\_Artificial Intelligence.doc
3. User Case Specification\_Basic logic.doc
4. User Case Specification\_Game Mode.doc
5. User Case Specification\_Network Connection.doc
6. SystemRequirementsDocument.doc
7. Chinese chess enterprise architect design model.

# Architectural Representation

The architecture of the application is represented following the recommendations of the Rational Unified Process and the Rational Architecture Practice guidelines. This document presents the architectural as a series of views:

1. Use Case View
2. Logical View
3. Process View
4. Implement View
5. Deploy View

# Use-Case View

The following picture shows the use-case view.

WANCC:Use Case Model.pdf

Each implemented use case has an associated Use Case Specification document. Extra details are dropped from this document.

## Architecturally Significant Use Cases

Architecturally significant use cases are defined as use cases with important functions or architectural importance. In this system they are:

1. Basic logic

Definition:

Basic logic use case is for system to manage the whole chess board and decide whether the game ends.

Basic Scenarios:

a) An action of player is detected.

b) An action of AI is generated.

c)Receive information from Network connection user case.

1. Artificial intelligence

Definition:

Artificial intelligence use case is for system to support a man-machine chess game.

Basic Scenarios:

a) It is AI’s turn to move.

1. Network connection

Definition:

Network connection use case is for system to support a man-man chess game.

Basic Scenarios:

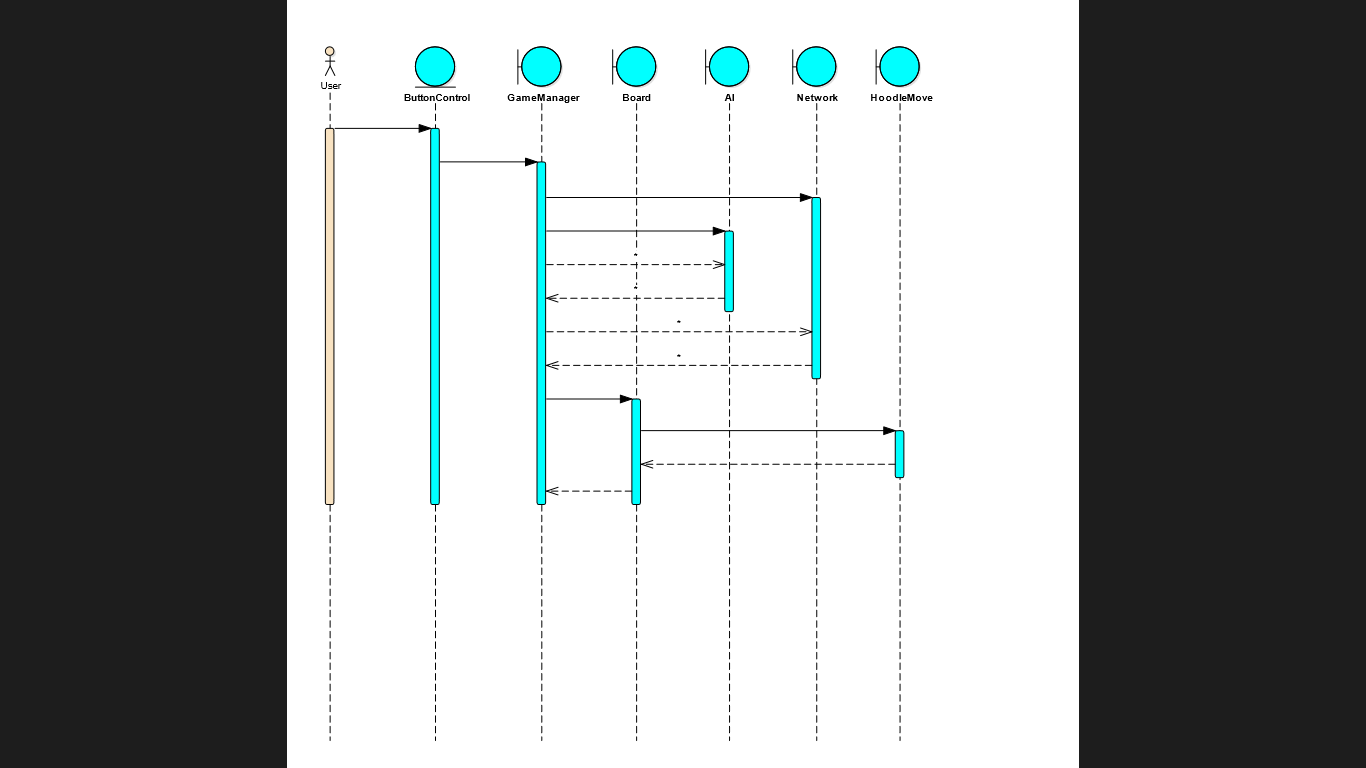
a) It is player’s turn to move.

# Logical View

This section describes the architecturally significant parts of the design model and the logical structure of the system. It starts from the overview of the architecture and then presents its key structure, behavioral elements and mechanisms.

## Overview

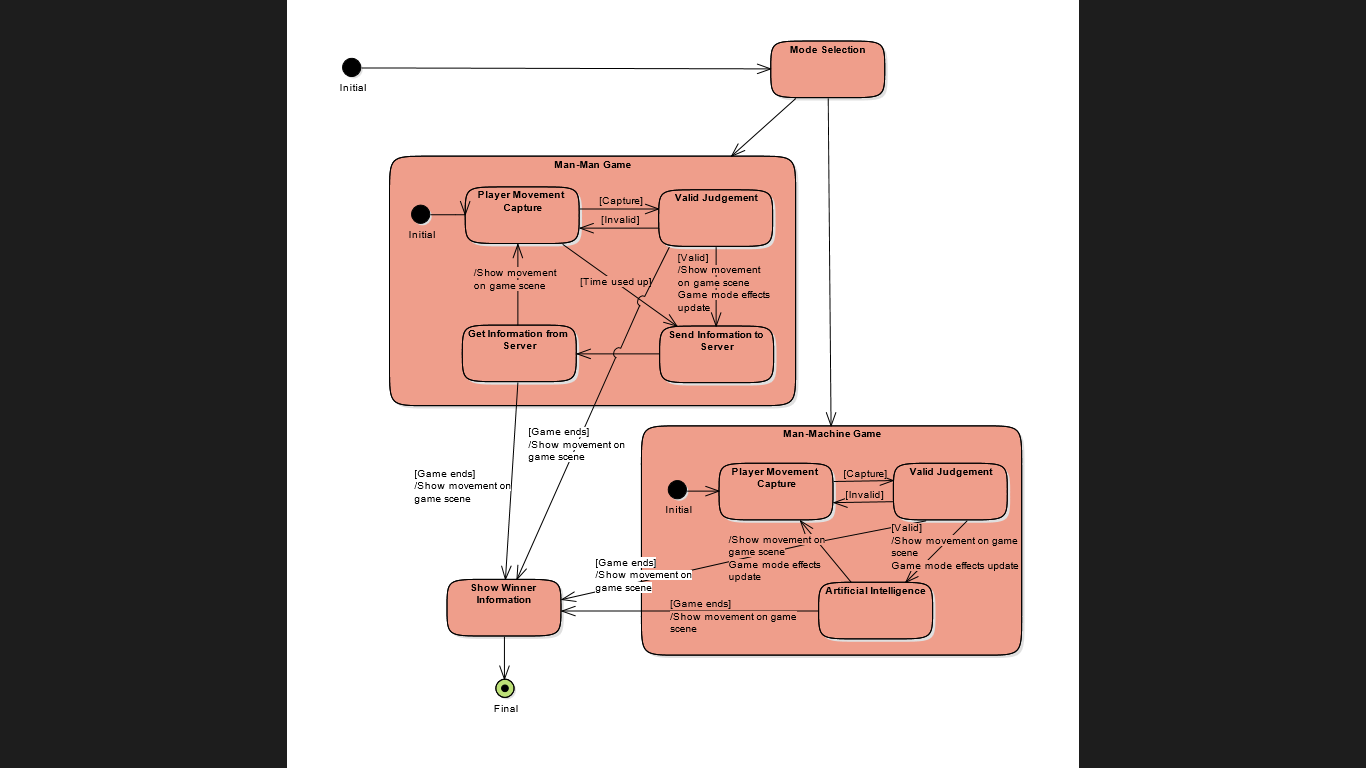
The sequence model goes as follows.



## Architecturally Significant Model Elements:

GameManager class supports the most architecturally significant part of the whole project.

The State Model goes as follows.



# Process View

We adapt Unity3D as platform, thus the process handling details are encapsulated.

# Deployment View

The deployment view of a system shows the physical nodes on which the system executes and the assignment of the system processed to the nodes. The diagram below shows the most typical deployment configuration used by the development team.

Macintosh HD:Users:kami:Documents:本科:学习:计算机:软件工程:SE project:03 Design:Deployment Model.pdf

# Implementation View

In order to simplify the implement, we adapt Unity3D as platform.