

**SOEN 6441 (Advance Programming Practices)**

**Project: Lanterns: The Harvest Festival**

Build 2

GitHub repository - <https://github.com/gugagreen/6441-summer-2015>

Group A Team members

Gustavo Gallindo.

Maxime Lamothe.

Parth Pareshkumar Patel.

Armaan Gill.

Ruixiang Tan

**Submitted to**: DR. Steven Winikoff

Instructions for running the application can be found on the README.txt found at the following link: <https://github.com/gugagreen/6441-summer-2015/tree/master/lanterns>

**The following problems were fixed since the presentation on 28/7/2015**

1. **Dedications are now possible without exceptions**
2. **Endgame state is now possible**

Instructions in the README.txt allow for quick running of these features by using save files included in the repository**.**

*Our JavaDoc is included in our submission in a zip folder titled Build2\_doc*

Table of Contents

[1. Abstract 3](#_Toc425968862)

[2. Introduction 3](#_Toc425968863)

[3. Architecture Style 3](#_Toc425968864)

[4. Package Diagram 3](#_Toc425968865)

[5. Class Diagrams 5](#_Toc425968873)

[6. Software Development Artifacts 8](#_Toc425968875)

[7. References 8](#_Toc425968876)

# Abstract

This document explains the software architecture of the project Lanterns**:** The Harvest Festival. It is the computer-based version of the board game. The document contains the overview of the basic classes involved and the relationships between them and software artifacts used for the incremental development of the game application.

# Introduction

The project Lanterns The Harvest Festival is basically a game played with different combination of playing cards and favor tokens. The players of this game act here as artisans decorating the palace lake with floating lanterns. At the end of the festival which is the end of the game the player who earns the most honor wins the game.

# Architecture Style

Our project Lanterns the Harvest Festival is based on commonly used Model-View-Controller (MVC) architecture. This pattern is used to implement user interfaces. This pattern separates the application into three interconnected components to separate internal representations of information from the user input. Our central components, which are Dao and Services packages, are part of the model and represent the behavior of the application, which is independent from the user interface. The view part represents the output of the information where multiple views of the same information are possible. In our project it could be multiple combinations of different lake tiles and player cards etc. The third part, Game controller, accepts input from the user and converts it for use in the model or view. The model notifies its associated views and controllers of state changes so that could help view to update interfaces and for the controller to modify the available commands.

# Package Diagram

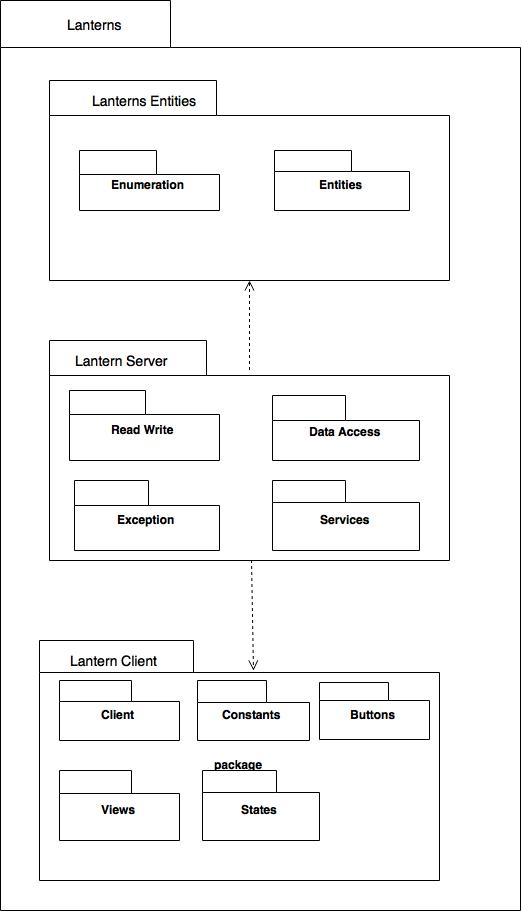
The package diagram on the following page depicts the dependencies between the packages that make up our model. Lanterns module consist of main 3 sub modules such as Lantern Entities ,Lantern Server and Lantern Client. Lantern Client package is for GUI and currently not being used in build two. We are planning to implement it in the next build.

Further breakdown of sub packages are:

Lantern Entities contains Enumerations and Entities, such as colours and players.

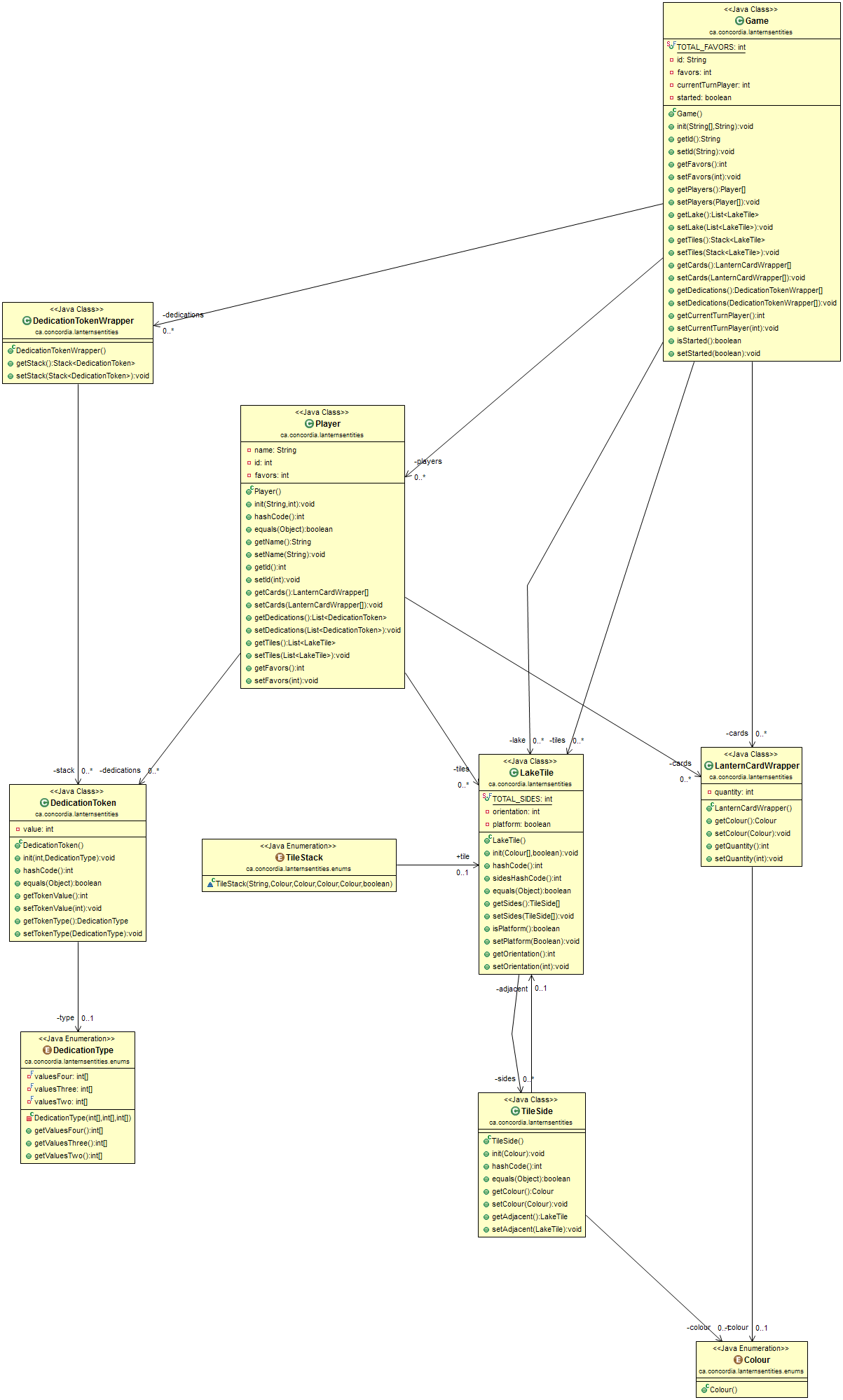
Lantern Server contains Services, Read Write, Data Access classes, and Exceptions.

Lantern Client consist of Client , Constants , Buttons , Views and States, used for the UI.

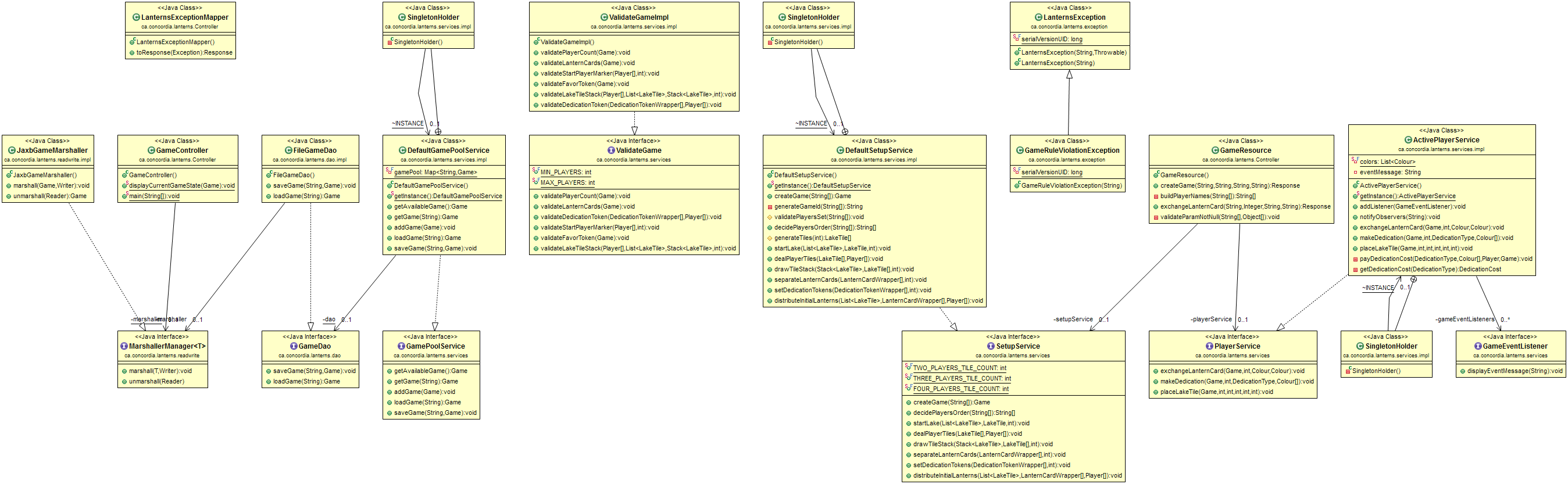


# Class Diagrams

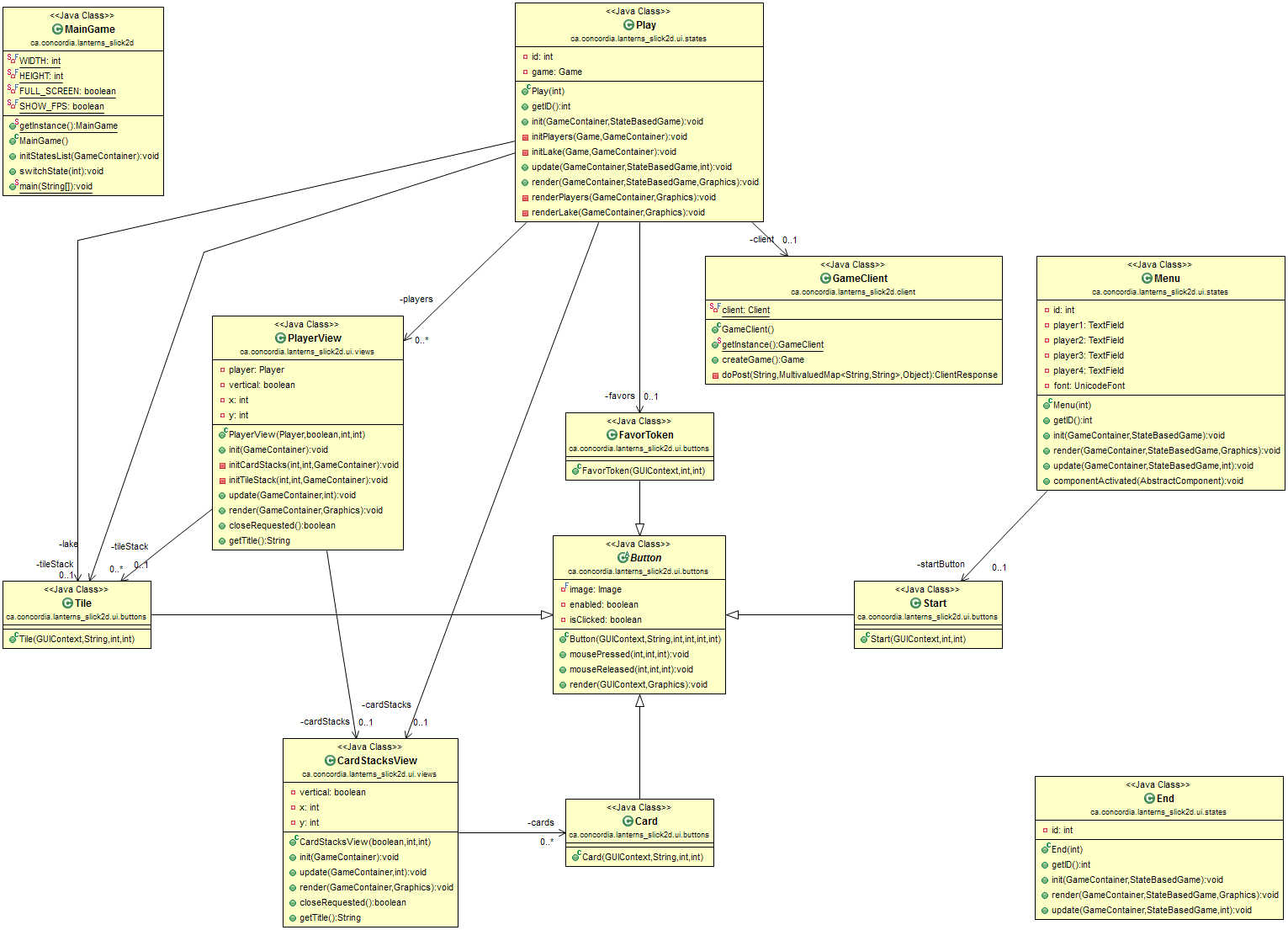
Lanterns Entities class diagram (zoom-in or open separate .png file for proper view).

****

Lanterns Server class diagram (zoom-in or open separate .png file for proper view).



Lantern Client(Slick 2D) class diagram (zoom-in or open separate .png file for proper view).



# Software Development Tools

The following are the tools used for the development of the application:-

Eclipse, GitHub, JUnit, Apache Maven, JAXB, Slick2D

# References

www.oracle.java.docs

www.javaranch.com

Lecture Notes by Dr. Steven Winikoff , SOEN 6441.