

Introduction to Frameworks

In this recitation you will create a TicTacToe plug-in for a simple framework that facilitates the implementation of 2D grid games. Plug-ins must only implement the game’s logic, making creating new grid games fairly straightforward.. Everything else (plug-in registration, player management, GUI implementation, etc.) is done by the framework.

The framework’s core implementation is located in the `edu.cmu.cs.cs214.rec09.framework.core` package and provides you with a `Player` class, a `GamePlugin` interface, and a `GameFramework` interface defining the methods `Plugins` can call on the framework. Grid game plug-ins must implement the `GamePlugin` interface in order to be registered with the framework. The `GamePlugin` interface contains several lifecycle methods that are called at various times throughout the period of a game. It also has a few getter methods that the framework will call to obtain the name of the plug-in game, the width/height of the plug-in game’s board, etc.

Two example plug-ins, “Rock-Paper-Scissors” (a classic grid-based game, seriously) and “Memory”, are already implemented for you in the `edu.cmu.cs.cs214.rec09.plugin` package.

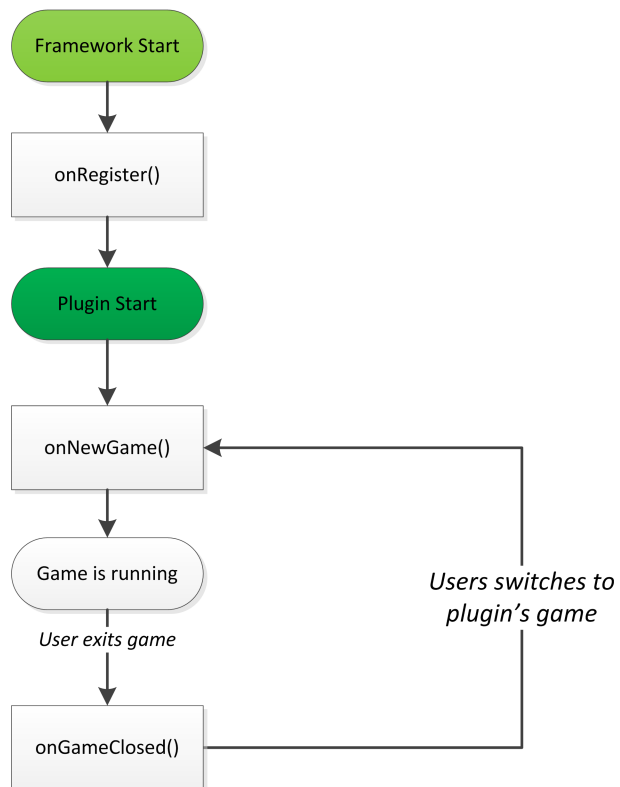
Concepts

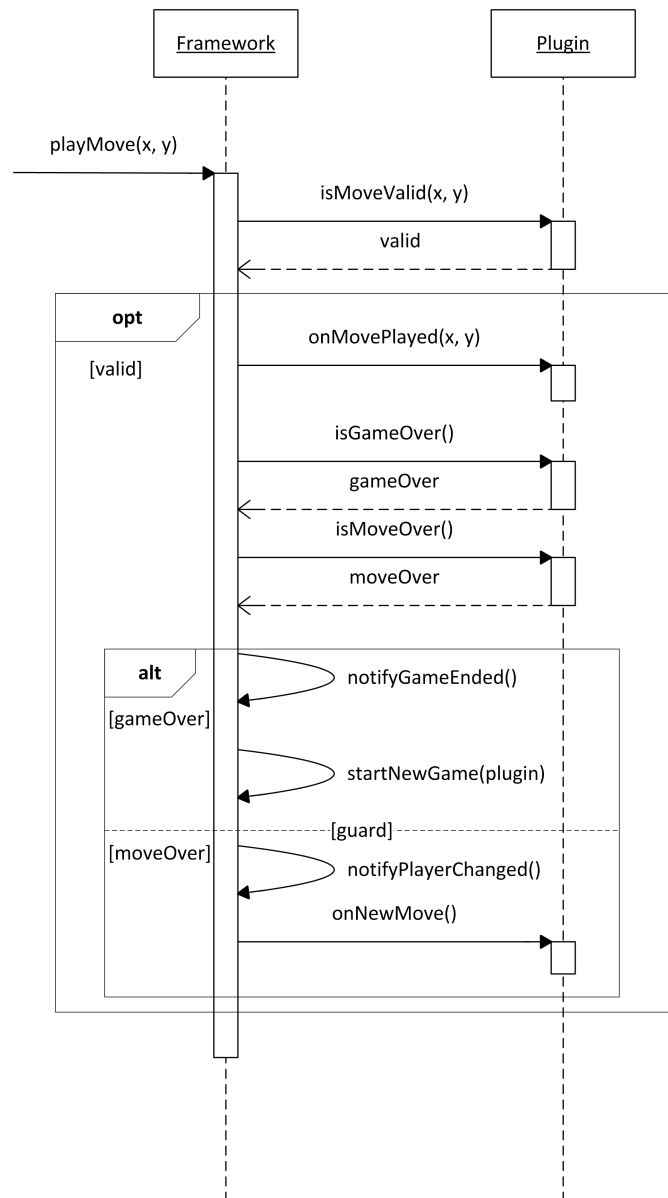
- Discuss similarities and differences shared between grid-based games. What common functionalities does the framework provide (cold spots), and what decisions are left up to the plug-ins (hot spots)?

Instructions

- Read the framework implementation you were given to familiarize yourself with what it provides you.
- Read the already implemented example plug-ins to get an idea of how you can implement your own plug-in.
- Read the documentation for the `GamePlugin` interface; in addition to the Javadoc comments, we have provided a lifecycle diagram for framework plugins and several interaction diagrams in this document. For each lifecycle method, be sure you understand when and why it is called.
- Implement a simple TicTacToe game plug-in in the `edu.cmu.cs.cs214.rec09.plugin` package. You may implement a different game if you have an idea you’d rather pursue. Note that you will need to update the `Main.java` file in `edu.cmu.cs.cs214.rec09` to register your plug-in with the framework.

Framework Lifecycle



Sequence Diagram: Playing a move.

Sequence Diagram: Starting a new game

