UNIVERSITÉ Concordia UNIVERSITY

SOEN 6441
ADVANCED PROGRAMMING PRACTICES

PROJECT: WARZONE

TEAM 15

Gaurang Dobariya	40184707
Jaynil Savani	40156070
Manthan Moradiya	40156072
Vicky Patel	40185238
Yash Vaghani	40155884

SUBMITTED TO: JOEY PAQUET

Contents

INTRODUCTION

In this project, we are developing Warzone. The Warzone is developed using Extreme Programming approach by following its features such as collective ownership, pair programming, continuous integration, and testing. Here, we are specifically following incremental development model and releasing build with small number of features in each iteration. Furthermore, this game is developed using MVC (Model, View, Controller) architecture.

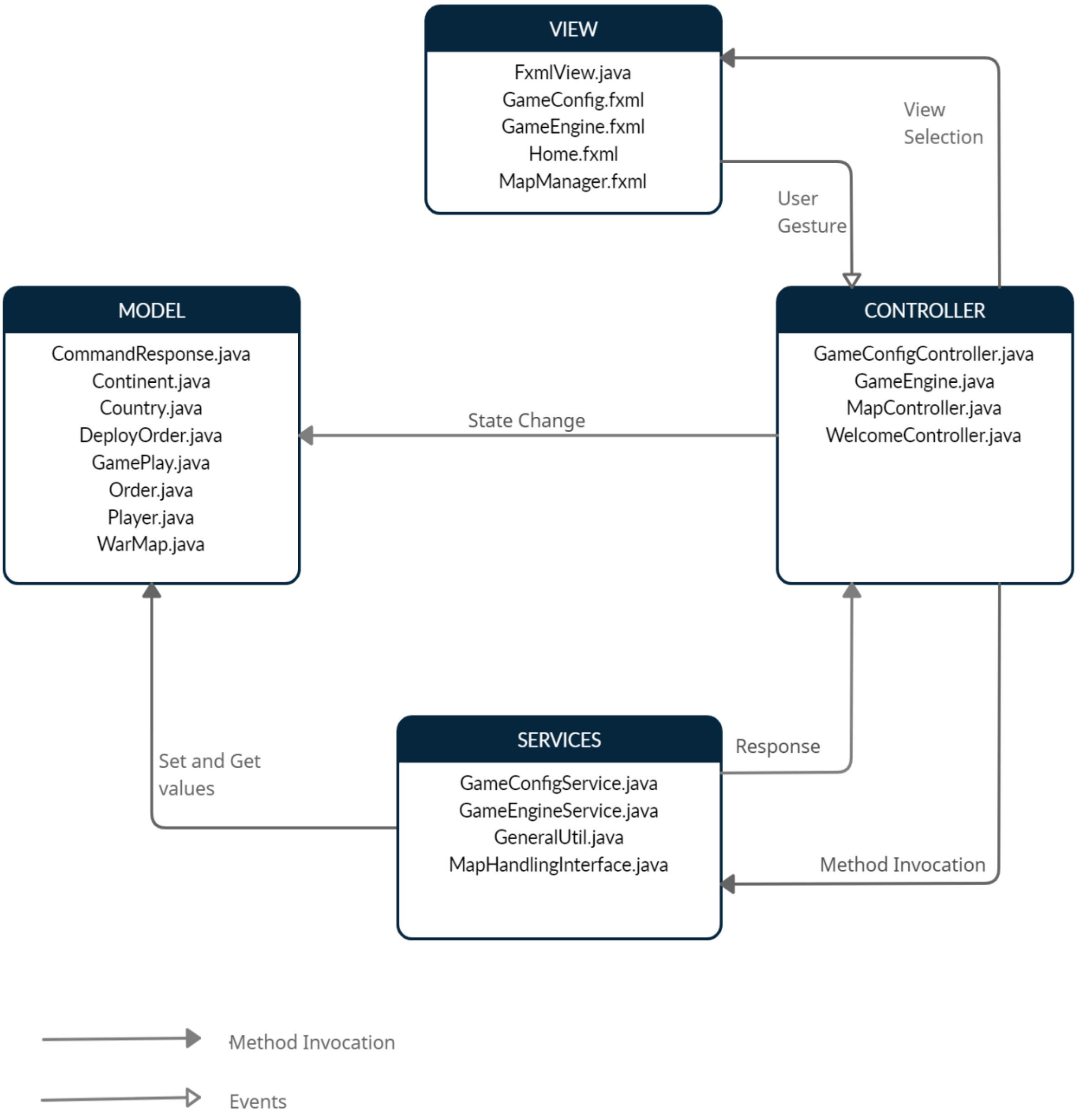
OBJECTIVES

1. Map Editor
    - User Driven creation or deletion of map elements (country, continent, and connectivity between countries)
    - Create a new map
    - Edit an existing map
    - Show map
    - Validate map

2. Game Play
    a) Start-up Phase
        - Create and add player
        - Assign countries to player

    b) Main Game Phase
        - Assign Reinforcements
        - Issue Orders
        - Execute Orders

MVC ARCHITECTURE DESIGN



Module Description

Model

File	Description
CommandResponse.java	This class represents structure of response of all the commands.
Continent.java	This class represents continent in the map, and it has a list of countries as well as control value.
Country.java	This class represents country of the map file. It is also having list of neighbor countries.
DeployOrder.java	This class is used for the deploy order command.
GamePlay.java	This class is used for manipulating game play information.
Order.java	This is the common interface for all order commands.
Player.java	This class is used for storing and manipulating player Information.
WarMap.java	WarMap class is the main model for map management. From here the data structure started. This Map is having Map of continents and continent will have countries and those countries will have their adjacent countries.

VIEW

File	Description
FxmlView.java	This class provides views of Application.
GameConfig.fxml	This file represents View which is responsible for Game Startup phase.
GameEngine.fxml	This file represents View which is responsible for playing game.
Home.fxml	This file represents Main Screen of the game.
MapManager.fxml	This file represents View which represents map related activities.

CONTROLLER

File	Description
GameConfigController.java	This class provides game configuration functionalities such as load map, create player and assign countries to player before the start game phase.
GameEngine.java	This class provides functionalities such as deploy armies, issue orders, and execute orders in round robin fashion.
MapController.java	This class provides map management functionalities such as add an existing map, create a new map.
WelcomeController.java	This class represent main screen of game from which user can navigate to various phases of game.

SERVICES

File	Description
GameConfigService.java	This interface is used for all game related Configuration for game play.
GameEngineService.java	This interface is used to provide function for game playing.
GenerateUtil.java	This interface is used for Utility of General Functions.
MapHandlingInterface.java	This interface is used for Utility of Map Editor related commands.