



Map Models

ContinentModel: It is the class to define a continent

CountryModel: It is the class to define a country

MapModel: It is the class to define a map. It is a observable object.

GamePlay Models

ArmyModel: It is the class to define an army

CardModel: It is the class to define a card

DiceModel: It is the class to define a dice and its actions

PlayerModel: It is the class to define a player and its actions

GameStateModel: It is the class to keep track all the information of the game

Map Views

BasicInfoView: it defines the window to show/edit the basic information of the map

ConnectionView: it defines the window to edit/delete a connection in the map

ContientView: it defines the window to create/edit a continent in the map

CountryView: it defines the window to create/edit a country in the map

MapContinentPanel: it defines the panel for showing continents information of the map

MapCountryPanel: it defines the panel for showing countries information of the map

MapEditorView: It defines the window for the map editor, it is the observer of the map model

GamePlay Views

BasicView: It defines the start window for the program.

FortificationView: It defines the window for the fortification phase of the game.

PlayerStartUpView:It defines the window for the start up phase of the game.

ReinforcementView: It defines the window for the reinforcement phase of the game.

ViewState: It defines the general window.

Map Controller

BasicinfoController:It is the class to edit the basic information of the map

ConnectionController: It is the class to create/update/delete a country connection in a map

ContinentController: It is the class to create/update/delete a continent in a map

CountryController: It is the class to create/update/delete a country in a map

MapEditorController: It is the class to initiate the related function after the user click a button on map editor view

MapEditorStartController: It is the class to open the map editor window.

GamePlay Controllers

AttackController:It is the class for player to perform attacks according to the rule of risk

FortificationController:It is the class to receive player action and call methods from correspondence model

GameStartController:It is the class to start the game

PlayerSetUpController:It is the class to start the setup phase of the player

ReinforcementController: It is the class to start the Reinforcement phase of the game

FinshFortificationController:It is the class to finish the Fortification phase of the game

FinshAttackController: It is the class to finish the Attack phase of the game

CardController: It is the class to start to start card phase of the game

DiceController: It is the class to start play Dice phase of the game