**3.1. User Interface Management Subsystem**

User Interface Management Subsystem includes 3 classes which are

Main, GamePanel and PauseMenu.

**Main Class**

Main class is used to view all menus that can be reached by the main

menu of the game. This class is the first class that is initialized when user opens

the game. The UI that is created by this class includes a text based interface with following options “Play Game”, “Instructions”, “High Scores”, “Settings” and “Exit Game”. When “Play Game” option is choosed, further commands are asked from the used by providing a set of choices in a text based format. If “Instructions”, “High Scores” or “Settings” option is choosed, new text based interface is created by the class in order to show the information that is wanted by the user. Main class gets required information from GameSystem class. Last choice on the menu, “Exit Game”, simply closes the game.

**GamePlay Class**

GamePlayclass is used to view game-play. When user wants to play the

game, this panel is initialized and user plays the game on this panel. During gameplay, GamePlayupdates itself using the GameSystem class. GameSystem

class is used because update process includes all game objects. If the user wants

to pause game, GamePlaymaintains and then shows PausePanel class.

**PauseGame Class**

PauseGame class is used to view pause menu that is shown when player

wants to pause the game. This panel is maintained by the GamePlay class and it

includes 3 choices which are “Continue Game”, “Main Menu” and “Exit Game”. If the first choice is choosed, user continues to play the game by interacting with the

GamePlay class. If the second choice is choosed, user goes back to the main

menu and if the last choice is choosed, game is closed.

**ConsoleView Class**

ConsoleView class is used to display console to interract with user.

**3.2. Attribute Management Subsystem**

Attribute Management Subsystem includes 2 classes which are HighScores and Help.

**HighScores Class**

HighScores class keeps track of the highest 5 scores scored in the game.

This class is maintained by the GameSystem class.

**Help Class**

Help class is used when the user wants to learn the instructions. Help class

keeps this information and supplies it to the GameSystem class when needed.

**3.3. Game Management Subsystem**

Game Management Subsystem includes 1 class which is GameSystem.

**GameSystem Class**

GameSystem manages all aspects of the game. GameSystem supplies

information to UI-related classes with the information it gets from the classes of the

Attribute Management Subsystem and Data Management Subsystem, and controls the progress of the game and updates the game during game-play. Decision of whether the game is over or not is also made by this class.

**Game Class**

It carries game related information and handles related actions. It reports to GameSystem.

**3.4. Game Map Management Subsystem**

Game Map Management Subsystem includes 2 classes which are MapManager and InteractionManager.

**MapManager Class**

Map holds the information for the position of player and the objects on the map. Contents of the class is initialized by MapManager. List of objects in map are

kept in this class and it is supplied to the GameSystem class when needed.

**InteractionManager Class**

InteractionManager class checks all commands given by user with the MapManager. This class provides information to GameSystem regarding interactions based on user commands.

**3.5. Data Management Subsystem**

Data Management Subsystem includes DataManager class.

**DataManager Class**This class loads and saves data created by the program. It provides information to GameSystem. Classes such as MapManager and InteractionManager need to communicate with GameSystem to make user of DataManager.

**3.6. User Management Subsystem**

User Management Subsystem includes following classes.

**UserManager Class**It handles user management actions. It provides information to GameSystem.

**User Class**It carries user information. It provides information to UserManager.

**Character Class**It carries character information. It provides information to UserManager.

**HostileCharacter Class**It carries hostile character information. It provides information to UserManager.

**Player Class**It carries player information. It provides information to UserManager.

**3.7. Map Objects Subsystem**

Map Objects Subsystem includes following classes.

**Direction Class**It carries direction information. It provides information to MapManager.

**Furniture Class**It carries furniture information. It provides information to MapManager.

**Item Class**It carries item information. It provides information to MapManager.

**Location Class**It carries location information. It provides information to MapManager.

**Reader Class**It carries reader information. It provides information to MapManager.

**Thing Class**It carries thing information. It provides information to MapManager.