CS319 PROJECT ANALYSIS REPORT

PART 1

GROUP G

ERKUT ALAKUŞ

DOĞANCAN DEMİRTAŞ

GÜNÇE KALYONCU

CANSU TÜZMEN

TABLE OF CONTENT

1. Introduction

Digger Unlimited is a developed and modified version of Digger that is a basic arcade game like *Pac-man* and *Dig Dug*. As in those games, the main aim is that collecting all gold and silvers without being caught by the monsters to finish the level. Unlike the original version, the Digger Unlimited will include new features like themes, bonuses and maps to make it more entertaining. While deciding these features, it is inspired by *Digger*, *Temple Run* and *Jetpack Joyride*:

Digger:

http://en.wikipedia.org/wiki/Digger_(video_game)

Temple Run:

http://en.wikipedia.org/wiki/Temple_Run

Jetpack Joyride:

http://en.wikipedia.org/wiki/Jetpack_Joyride

The game will only available as a desktop application so the player will need a mouse and a keyboard to control and play the game.

2. Overview

Unlimited Digger is a basic arcade game likely its ancestors. Basically, the game starts in a mine and there will be at least one tunnel for the Miner and two different monsters to move.

There will be also some gold and silvers for scoring and bonuses for changing the course of the game. The main goal of the player is expanding the existing tunnel by digging to collect all the stuff without being touched by these monsters to complete all levels successfully.

2.1. Gameplay

The player needs a keyboard to lead the Miner to move by direction keys and a mouse to click on the buttons like pause.

2.2. Leveling

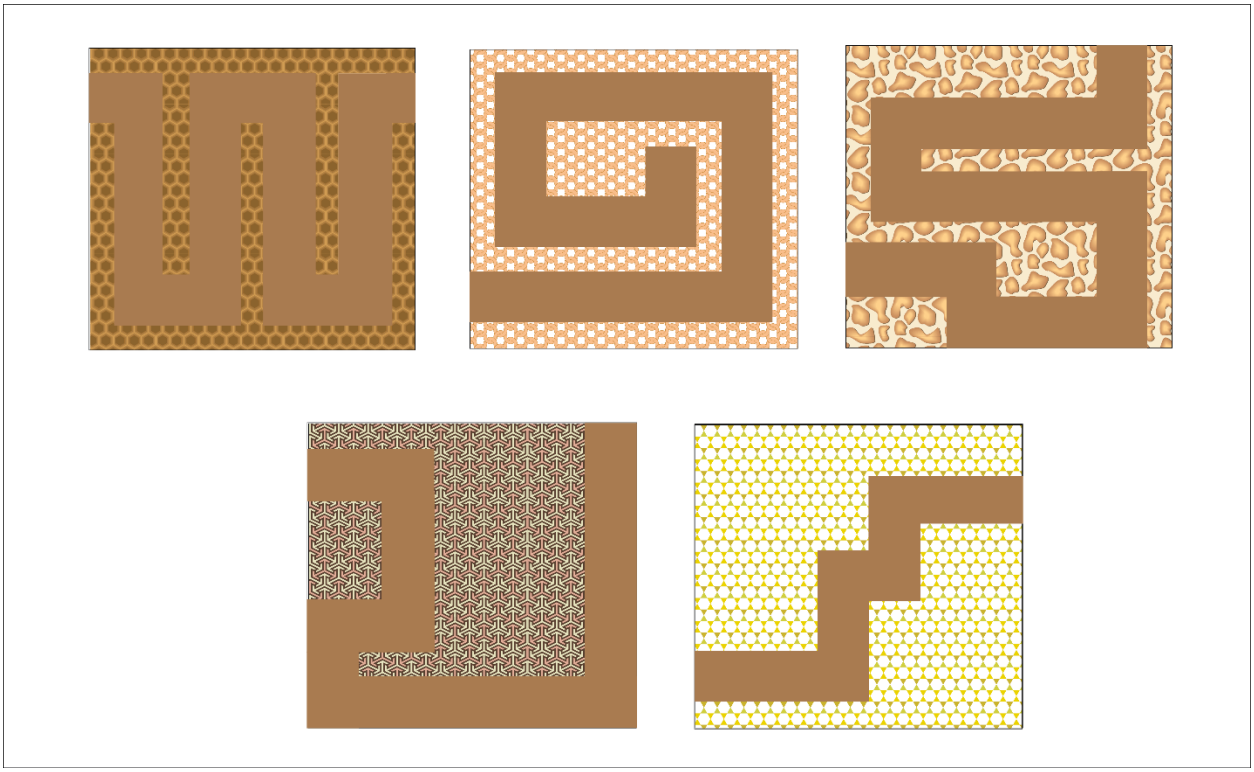


Figure 1: A representative View of 5 Levels

The game has 5 levels and they are sorted according to their difficulties. When the number of level increases, the total of excavated area at the beginning decreases. Therefore, it

gets harder to complete the game without being caught by a monster and collect all gold and silvers.

2.3. The Miner

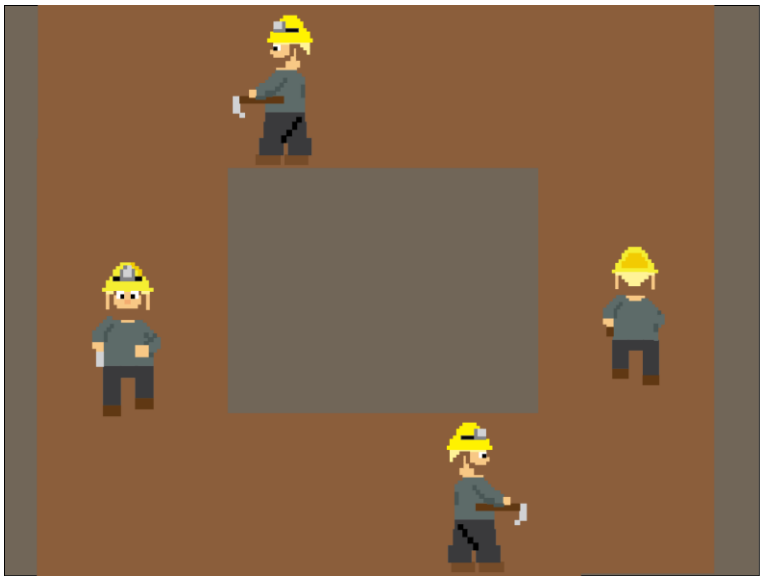


Figure 2: The movement of the Miner

The main character of the game is the miner as the digger. The player could only lead the miner by using the direction buttons of keyboard to make him dig vertically and horizontally.

2.4. The Monsters

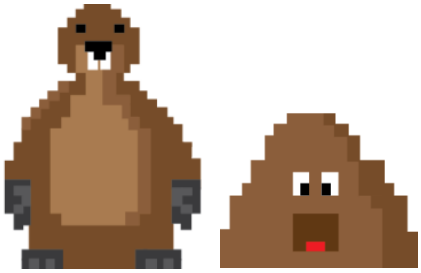


Figure 3: The Mole and the Earth Monster

The game contains two different types of monsters which are a mole and an earth monster. Both of them are able to move vertically and horizontally to catch the player without collecting all silvers and gold to finish. The Earth Monster is also able to dig like the Miner. When the player is caught by one of them, one the life is lost.

At the beginning of the game, there is one for the Earth Monster but the number of the mole increase pass over the levels. Until the player collects the one of the power-downs, the number of them does not change. They start the game in opposite corner of the miner so that the player could complete the game successfully.

2.5. Coins

There are some gold and silvers at unexcavated areas at the beginning of the game and the gold gain more score than silvers. The player scores 200 points for gold and 100 points for silver. However, the program selects the number of them randomly so the score is changing. The player should try to collect some power-ups to increase his score for a high one.

2.6. Bonuses

This new version of the game includes some power-ups and power-downs unlike the original to make it more amusing and challenging so these are unknown until the player collects them. The program selects this power-ups and power-downs randomly and after the player collects, the program will renew it. In a sense, there is a big possibility that the player collect the same, useful or harmful ones repeatedly and this makes the game more fun. The power-ups and power-downs are:

Power-ups:

ExtraLife: It gives an extra life to the player.

SilverToGold: It changes all silvers to gold.

DoubleGold: It multiplies the score of gold by two.

DoubleSilvers: It multiplies the score of silvers by two.

Triple Gold: It multiplies the score of gold by three.

TripleSilvers: It multiplies the score of silvers by three.

InvisibleMiner: It makes the miner invisible so that he can escape from the monsters more easiliy.

DigAll: It digs all maze.

DestroyMonters: It destroys two of the monsters.

Power-downs:

KillMiner: It takes one of the life of the player and if the player has only one life, he dies.

GoldToSilver: It changes all gold to silvers.

DigBack: It sends the player to the start point and cancels the excavated area during the game.

DoubleMonsters: It multiplies the number of the monsters by two.

3. Functional Requirements

3.1. Play Game

The player will be able to collect gold and silvers while keeping the digger away from the monsters with Digger Unlimited. It is an arcade game based on a simple “Pacman” concept. In the beginning of the game player has 3 lives. Whenever digger touches a monster, it dies and loses one of its lives and starts digging from its beginning location. After losing all 3 lives, game ends and player is directed to the main screen. Also a prompt is shown to the player asking him enter his nickname in order to show his score in the game in the high score table.

If player collects all the gold and silvers another level is started with a different map and harder game fiction. Amount of monsters are increased with respect to levels.

3.2. Select the Game Theme

Player will be allowed to select one of the 4 available themes as a game theme. Depending on the choice, digger and monsters will be changed. Available themes will be:

- Digger Unlimited Theme
- Halloween Theme
- Original Digger Theme
- Christmas Theme

3.3. Change Settings

Player should be able to change game settings. Available settings are:

- Sound on/off
- Music on/off

These preferences will be stored accordingly and new games will be started with respect to previous choices of user.

3.4. Pause Game

Player will be able to pause the game, by pressing the “Escape” button. A sub menu will be displayed with this action. In this sub menu the following menu items will be shown to the player:

- Resume
- Retry
- How to Play

When player clicks on resume item, game will be resumed. If player clicks on “Go to Main Menu” item, he will be redirected to main menu.

3.5.View Credits

Player will be able to see the credits, after clicking it from the main menu. Developers’ name and their pixel art drawings will be displayed on the screen. As Digger Unlimited is a modified version of original Digger game, a link to its web site will be given.

3.6.How to Play

Player will find useful information about how to move the digger, and what is the main purpose of the game. This will also include definitions of power-up/downs and more information about the monster types.

4. Nonfunctional Requirements

4.1. Frame Rate

Average frame rate of the game will be 30 frames per second.

4.2. Response Time

Since Digger Unlimited game will be played locally on a computer, average response time will not be more than 0.5 second.

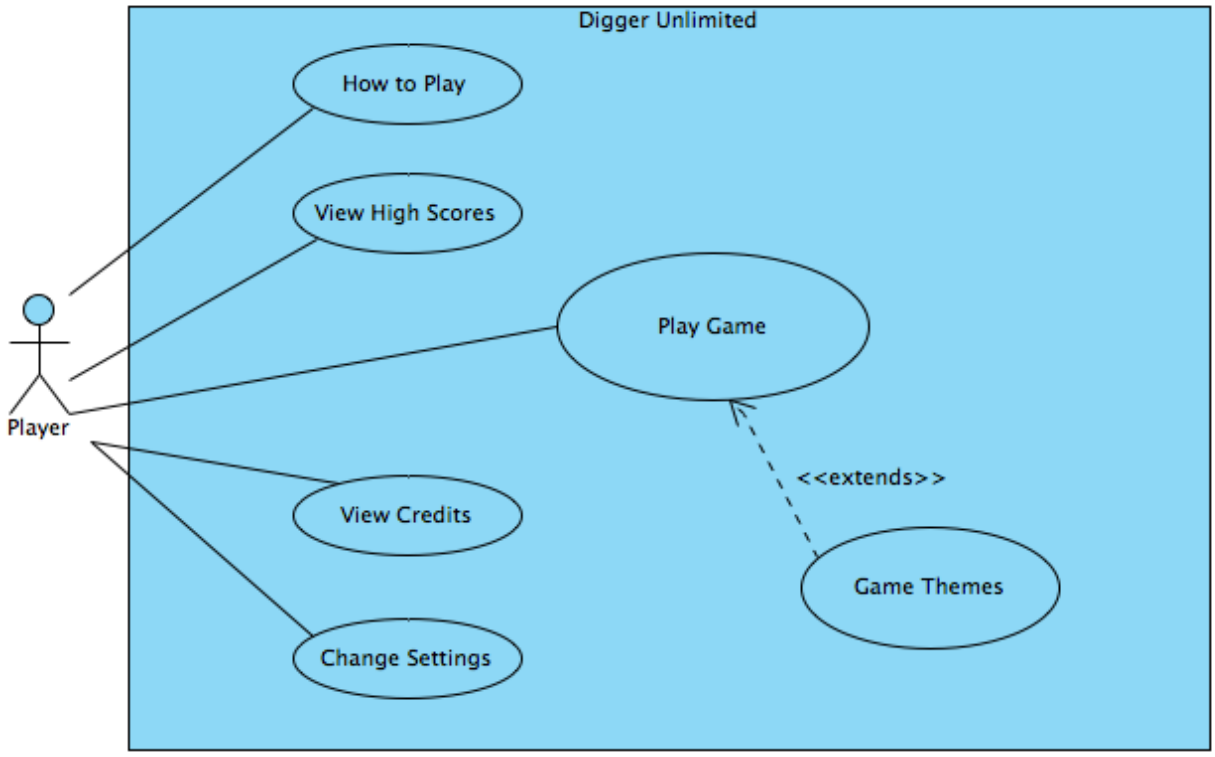
4.3.Platform

Since the game will be written in Java, any computer with Java installation could run the game.

5. System Models

5.1. Use Case Model

In this section, we will provide use case diagram and models of Digger Unlimited game to express it more clearly.



5.1.1. How to Play

Use Case ID:	UC_HP_001		
Use Case Name:	How to Play		
Created By:	GncKlync	**Last Updated By:**	-
Date Created:	13/10/2014	**Last Revision Date:**	-
Actors:	Player		
Description:	Player could read the manual of the game to learn how to play the game.		
Trigger:	Player selects “How to Play” from Main Menu.		
Preconditions:	-		
Postconditions:	-		
Normal Flow:	1. Player selects “How to Play” option from the Main Menu. 2. How to Play will be displayed to the Player on screen.		
Alternative Flows:	A. If Player wants to exit from the How to Play screen A.1. Player select “Back” button to return “Main Menu” A.2. System displays “Main Menu”		
Exceptions:	-		

Includes:	-
Frequency of Use:	-
Special Requirements:	-
Assumptions:	Player will read the manual and ACTUALLY learn the game and buttons to play it.
Notes and Issues:	-

5.1.2. View High Scores

Use Case ID:	UC_HS_001		
Use Case Name:	View High Scores		
Created By:	GncKlync	**Last Updated By:**	-
Date Created:	13/10/2014	**Last Revision Date:**	-
Actors:	Player		
Description:	This event will display “High Scores” menu when its triggered by the user. Also makes the transition between “High Scores” and “Main Menu”.		
Trigger:	Player select “View High Scores” from Main Menu.		
Preconditions:	**System should keep a record of top ten scores.**		
Postconditions:	-		
Normal Flow:	1. Display top ten high scores with player names.		
Alternative Flows:	A. To go back to main menu at any time: A.1. Player select “Back” button to return “Main Menu” A.2. System displays “Main Menu”		

Exceptions:	
Includes:	HighScoreChart
Frequency of Use:	-
Special Requirements:	-
Assumptions:	If there is not enough high scores saved by the player(less than 10), System will fill out the chart with player name “XXX” and score of “0”.
Notes and Issues:	-

5.1.3. Play Game

Use Case ID:	UC_PG_001		
Use Case Name:	Play Game		
Created By:	GncKlync	**Last Updated By:**	-
Date Created:	13/10/2014	**Last Revision Date:**	-
Actors:	Player		
Description:	Player tries to finish the game by completing all the levels with the highest score. System will keep the score, with a desired name from the Player in the end if Player gets a high score.		
Trigger:	Player select “Play Game” from Main Menu and select a theme from the "Game Themes" menu.		
Preconditions:	**Game loaded according to the game theme and settings. At start, if Player did not change game settings, game will begin with default settings.**		
Postconditions:	If score is greater than the top 10 high scores, system will update the high score list with this new score by letting Player		

	enter a name for the score.
Normal Flow:	1. System starts the game. 2. Player will start from the first level. 3. Player will gather all the silver and gold in the level. 4. System will load the next level, when Player finishes the current one. 5. Play will start to play the next level. - Steps 3-5 will be repetitive for the all 5 levels of the game. 6. If Player scores higher than a value in the high score chart, System will let player enter a name to save his score to the high score list. Otherwise System will show the high score list only. 7. System returns to Main Menu.
Alternative Flows:	3A. Player tries to finish the level by collecting all the silver and gold coins while running away from the creatures. 3A.1. Player creates roads to coins by digging the map. 3A.2. Map is getting cleaner as the Player moves and digs. 3A.3. If Player comes upon a coin, he will be rewarded with points accordingly. 3A.4. System will update Player's score. 3A.5. System will check if all the coins are gathered.

	-If not player will continue again from the 3A.1.-3A.5. 3A.6. If all the coins gathered by the Player, level ends and System will upload the next level with a new map. 3B. Player gets a power-up or power down. 3B.1. Player will gather the bags, which is not required to finish the level. 3B.2. These bags will drop power-ups/downs. 3B.3. Player will collect a power-up/down. 3B.4. System will clear the power-up/down from the screen, whether Player collects it or it is removed after 15 seconds. 3B.5. If Player claims a power-up/down; System will do the necessary changes over the game accordingly. -This process is same for every power-up/down process. A. If Player pauses the game. A.1. Player press the proper key to pause the game. A.2. System will pause the game. A.3. System will show the “Pause Menu”

	A.3.1. If Player selects “Resume”, system will return to game. A.3.2. If Player selects “Return to Main Menu”, System will close the current game and opens the “Main Menu” A.3.3. If Player selects “Change Settings”, System will call “UC_CS_001” use case. A.3.4. If Player selects “How to Play”, System will call “UC_HP_001” use case. A.3.5. If Player selects “Exit Game”’, System will return to the desktop by closing the game.
Exceptions:	-
Includes:	-
Frequency of Use:	-
Special Requirements:	-
Assumptions:	-
Notes and Issues:	-

5.1.4. View Credits

Use Case ID:	UC_CR_001		
Use Case Name:	View Credits		
Created By:	GncKlync	**Last Updated By:**	-
Date Created:	13/10/2014	**Last Revision Date:**	-
Actors:	Player		
Description:	This event will display “Credits” screen when its triggered by the user. Also makes the transition between “Credits” and “Main Menu”.		
Trigger:	Player select “View Credits” from Main Menu.		
Preconditions:	-		
Postconditions:	-		
Normal Flow:	1. Display information about developers and their messages about the game.		
Alternative Flows:	A. To go back to main menu at any time: A.1. Player select “Back” button to return “Main Menu”		

	A.2. System displays “Main Menu”
Exceptions:	-
Includes:	-
Frequency of Use:	-
Special Requirements:	-
Assumptions:	-
Notes and Issues:	-

5.1.5. Change Settings

Use Case ID:	UC_CS_001		
Use Case Name:	Change Settings		
Created By:	GncKlync	**Last Updated By:**	-
Date Created:	13/10/2014	**Last Revision Date:**	-
Actors:	Player		
Description:	Player could change the default game settings such as “theme”, “sound on/off”, “music on/off” from the “Change Settings” screen.		
Trigger:	1. Player selects “Change Settings” from Main Menu. 2. Player selects “Change Settings” from Pause Menu.		
Preconditions:	**Current – if not changed by the player, default – settings will be shown to the player.**		
Postconditions:	Game settings are updated.		
Normal Flow:	1. Player selects “Change Settings” option from the main menu or Pause Menu. 2. Game Settings will be displayed to the Player on screen.		

	3. Player will change/update game settings. 4. System will update the game accordingly.
Alternative Flows:	To A. If Player wants to exit from the Game Settings screen A.1. Player select “Back” button to return previous menu A.2. System displays “Main Menu” or “Pause Menu” accordingly
Exceptions:	-
Includes:	-
Frequency of Use:	-
Special Requirements:	-
Assumptions:	-
Notes and Issues:	-

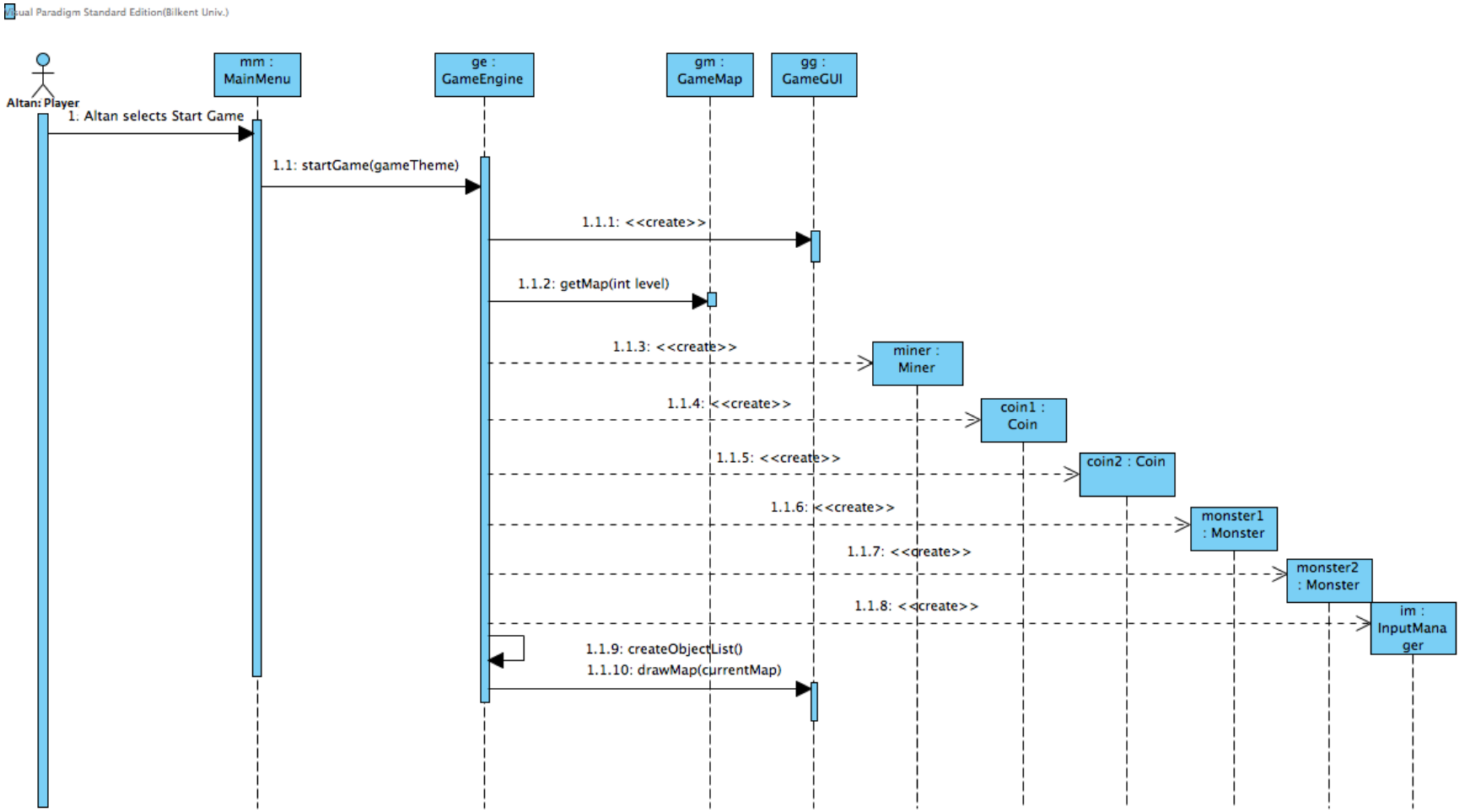
5.2.Dynamic Models

5.2.1. Scenarios

5.2.1.1. Start Game

Scenario :

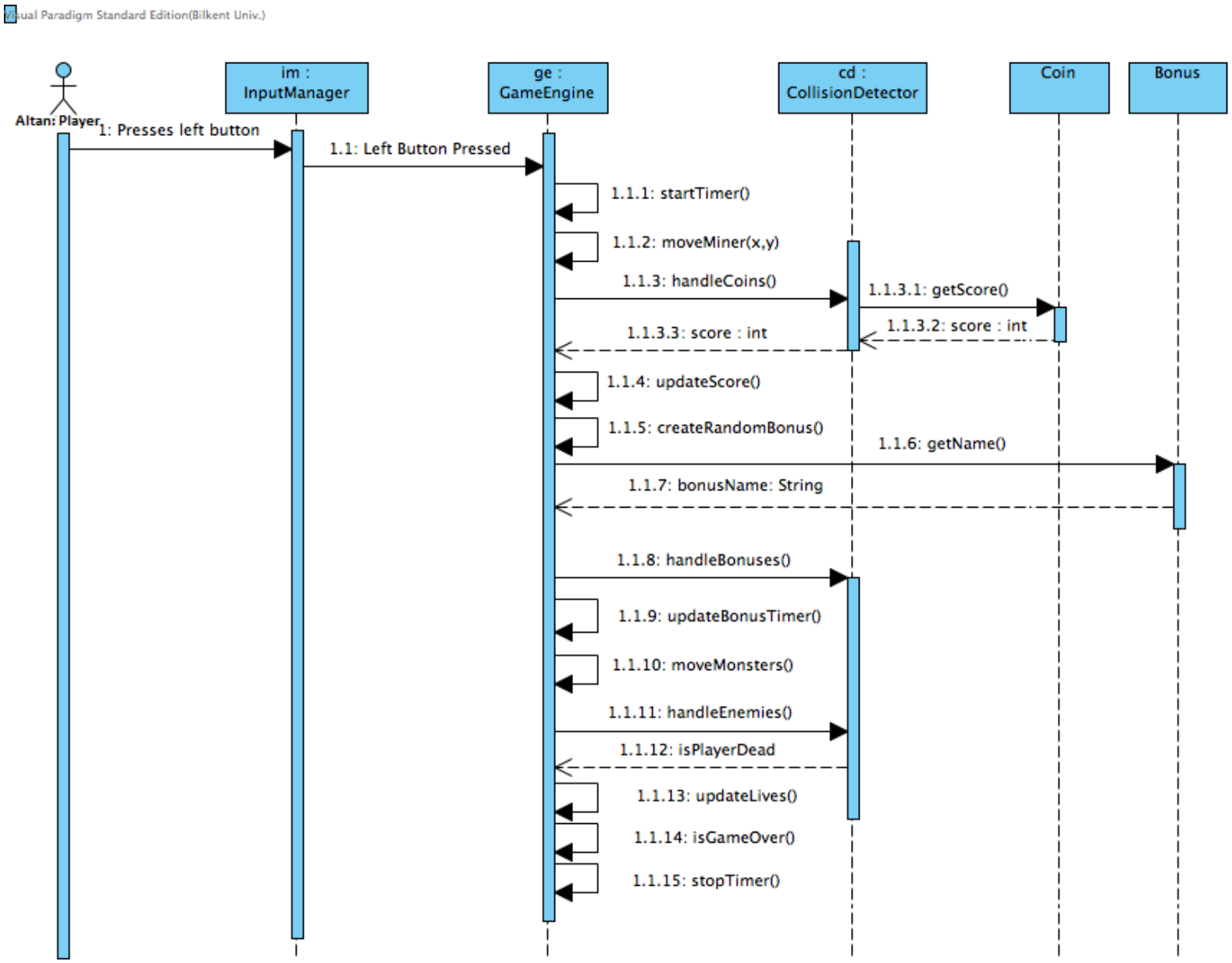
Actor Altan clicks on icon of the game and selects “ start game” from main menu. Game Engine is created. Game Engine creates GameGUI and gets the map according to level and then Game Engine creates game objects. Finally, it calls GameGUI to draw the current map.


Visual Paradigm Standard Edition(Bilkent Univ.)
Altan: Player
mm : MainMenu
ge : GameEngine
gm : GameMap
gg : GameGUI
1. Altan selects Start Game
1.1: startGame(gameTheme)
1.1.1: <<create>>
1.1.2: getMap(int level)
1.1.3: <<create>>
miner : Miner
1.1.4: <<create>>
coin1 : Coin
1.1.5: <<create>>
coin2 : Coin
1.1.6: <<create>>
monster1 : Monster
1.1.7: <<create>>
monster2 : Monster
1.1.8: <<create>>
im : InputManager
1.1.9: createObjectList()
1.1.10: drawMap(currentMap)

5.2.1.2. Play Game

Scenario:

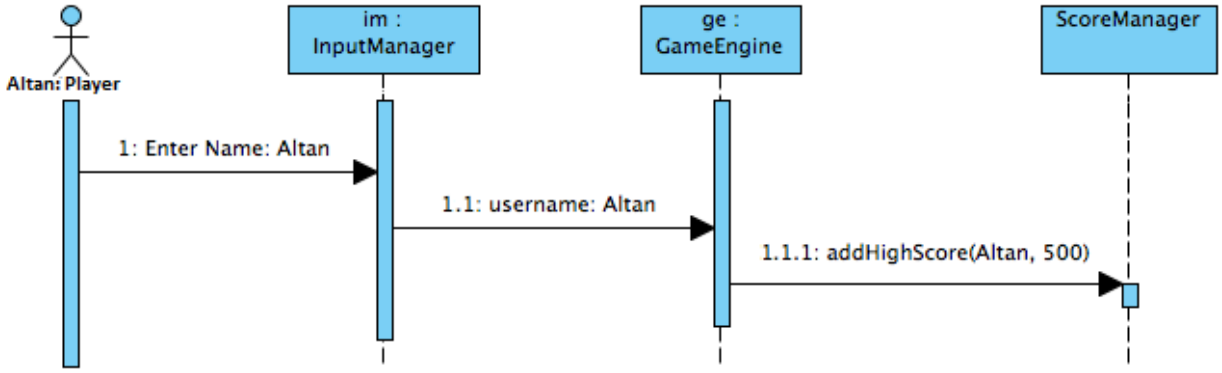
After the scenario above takes place, Actor Altan starts game by pressing left button. Miner eats one of the coins, his total score is increased depending on its type. He collects one of the bonuses, it affects the course of game according to its type. He touches a monster, he loses his last life and the game is over.


Visual Paradigm Standard Edition(Bilkent Univ.)
Altan: Player
im : InputManager
ge : GameEngine
cd : CollisionDetector
Coin
Bonus
1: Presses left button
1.1: Left Button Pressed
1.1.1: startTimer()
1.1.2: moveMiner(x,y)
1.1.3: handleCoins()
1.1.3.1: getScore()
1.1.3.2: score : int
1.1.3.3: score : int
1.1.4: updateScore()
1.1.5: createRandomBonus()
1.1.6: getName()
1.1.7: bonusName: String
1.1.8: handleBonuses()
1.1.9: updateBonusTimer()
1.1.10: moveMonsters()
1.1.11: handleEnemies()
1.1.12: isPlayerDead
1.1.13: updateLives()
1.1.14: isGameOver()
1.1.15: stopTimer()

5.2.1.3. New High Score

Scenario:

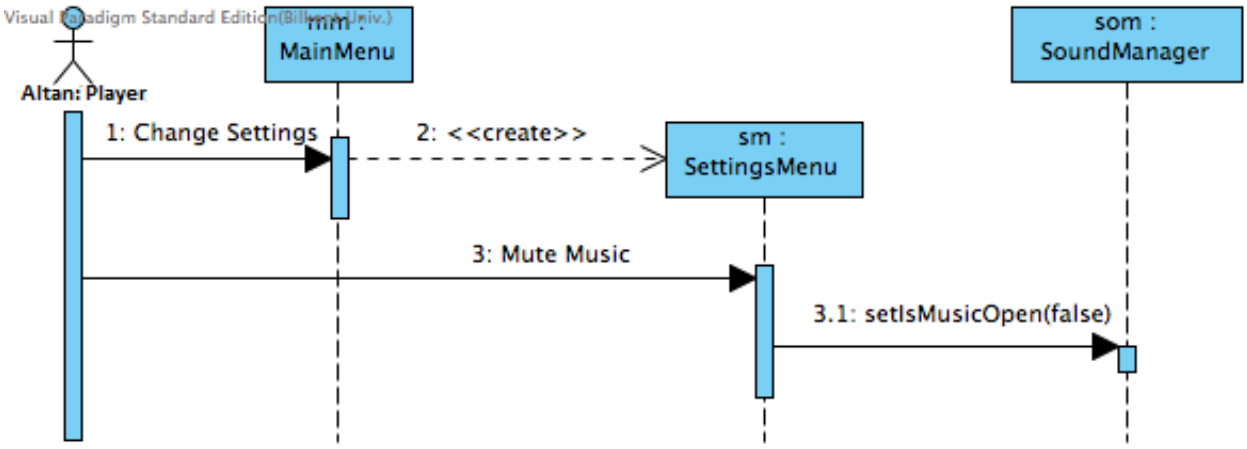
Actor Altan makes a high score and enters his name and his name appears on high score list.



5.2.1.4. Change Settings

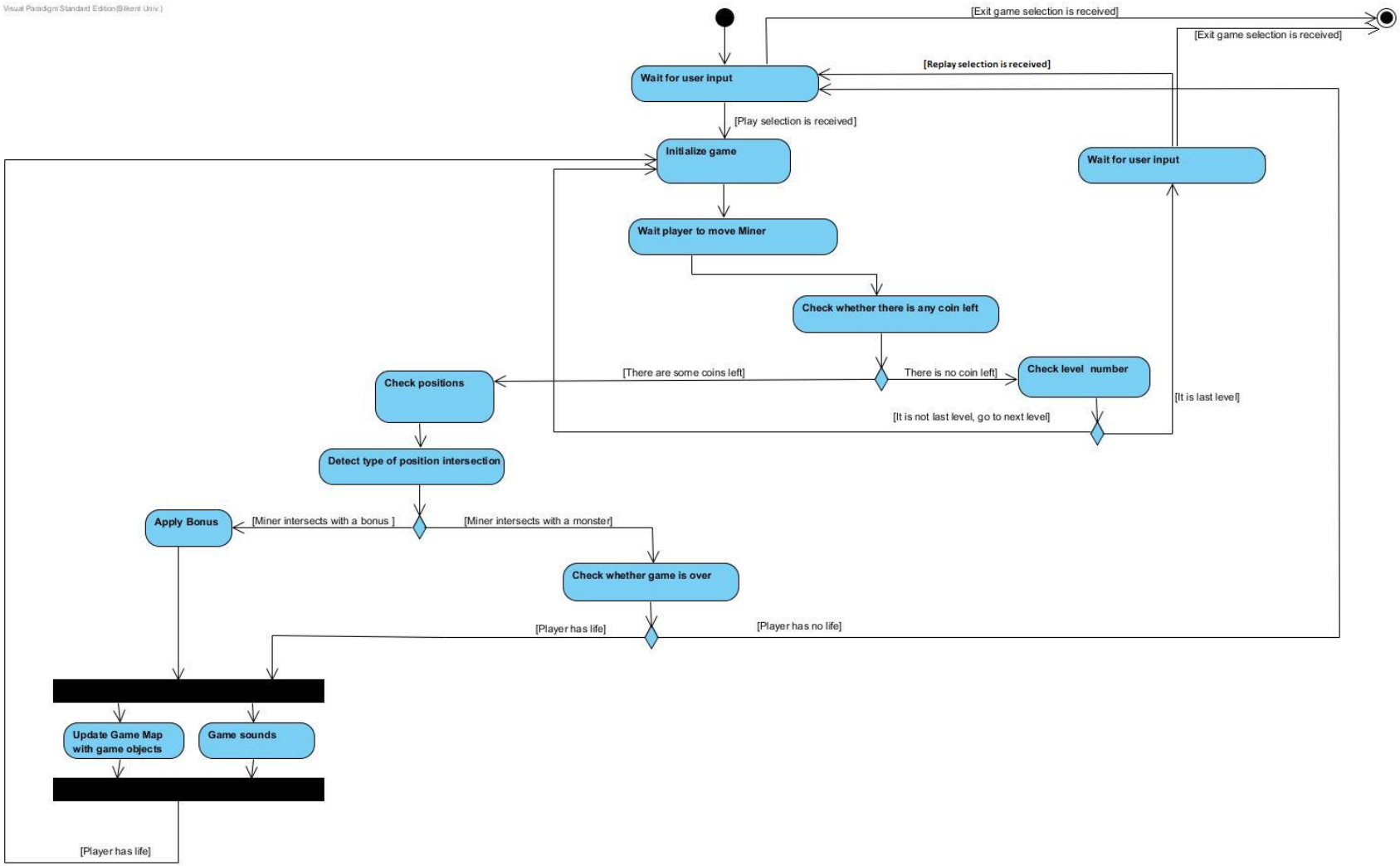
Scenario:

Actor Altan selects “Change Settings” option from main menu and Settings screen is opened. He mutes music of the game.

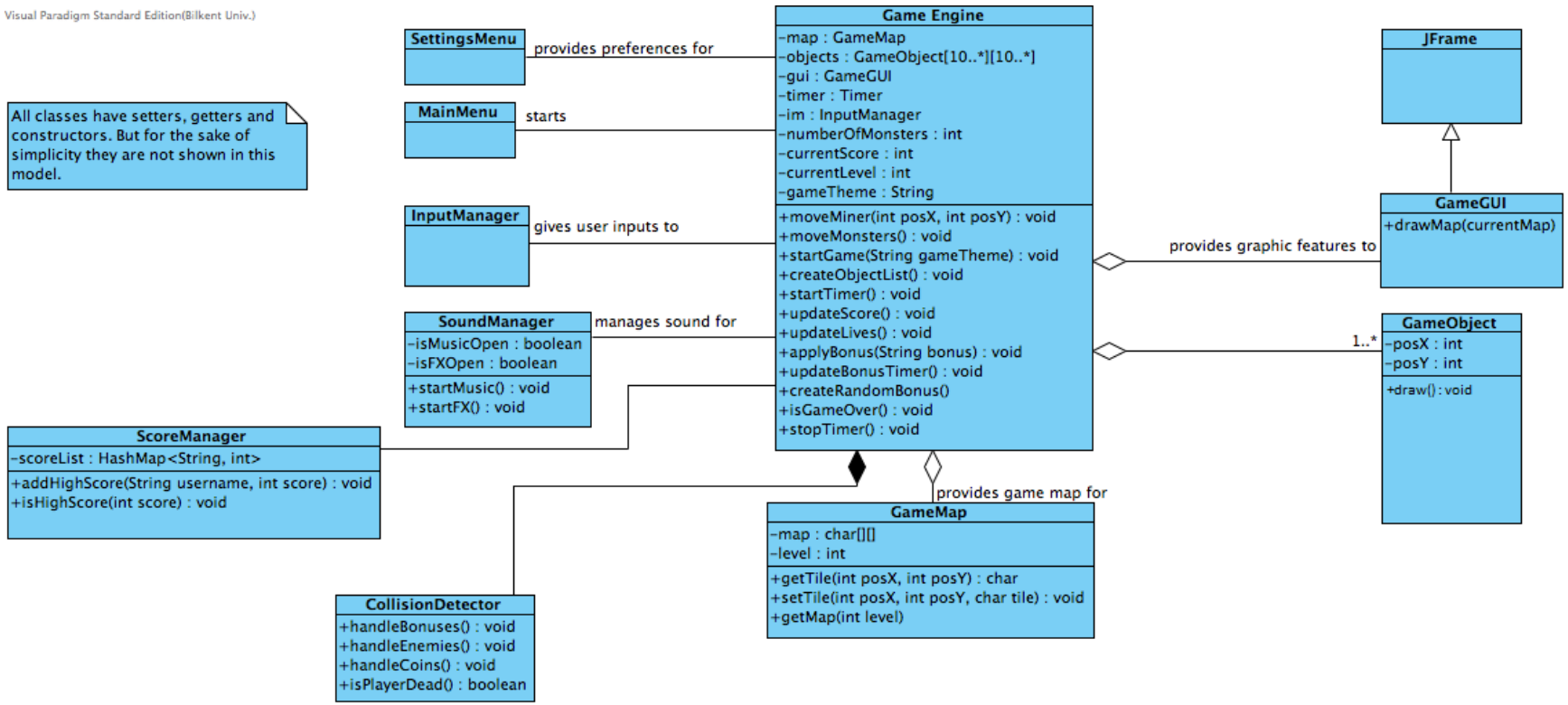


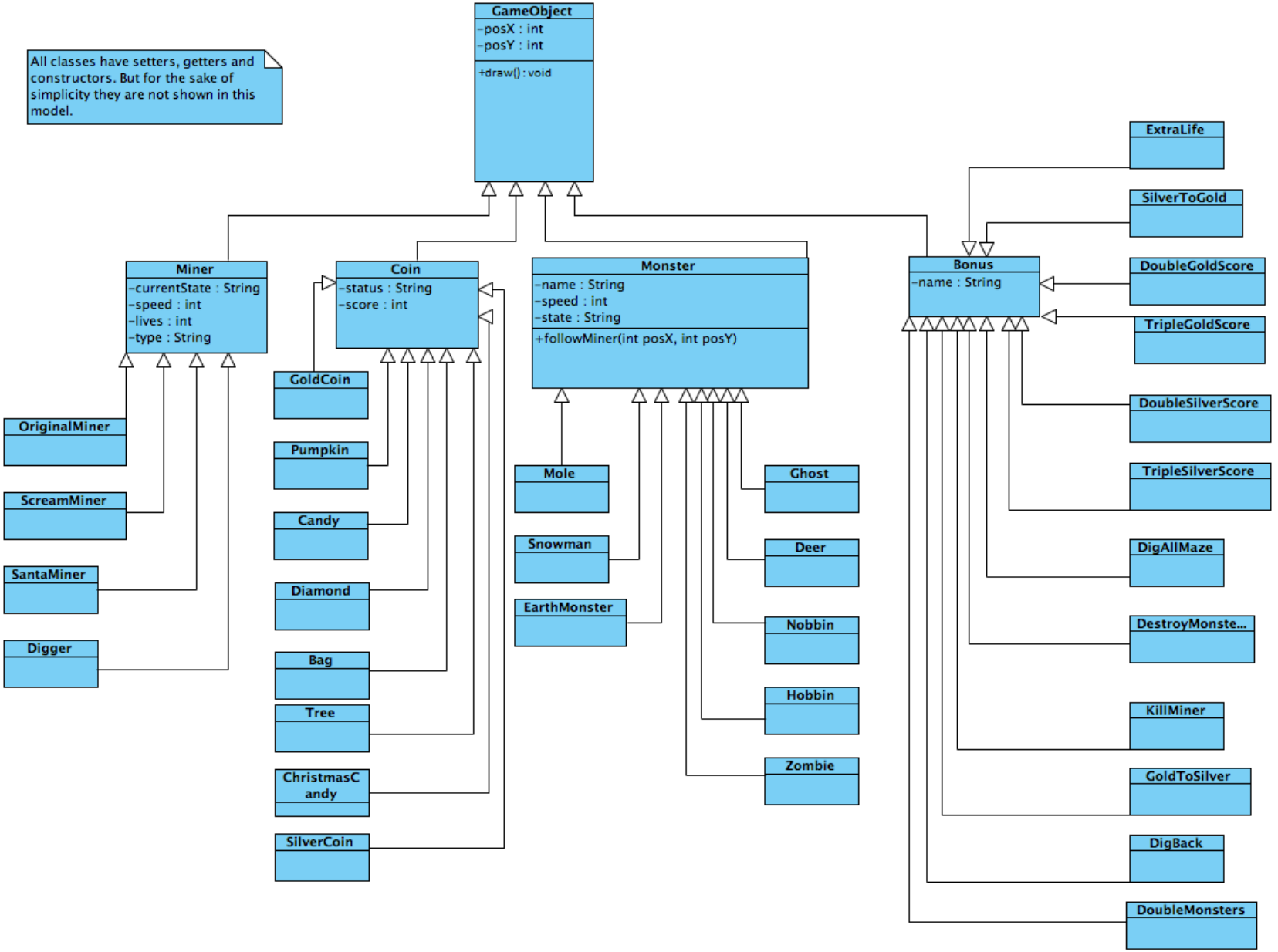
5.2.2. Activity Diagram

It is showed that how system maintains gameplay by this activity diagram.



5.2.3. Object and Class Model



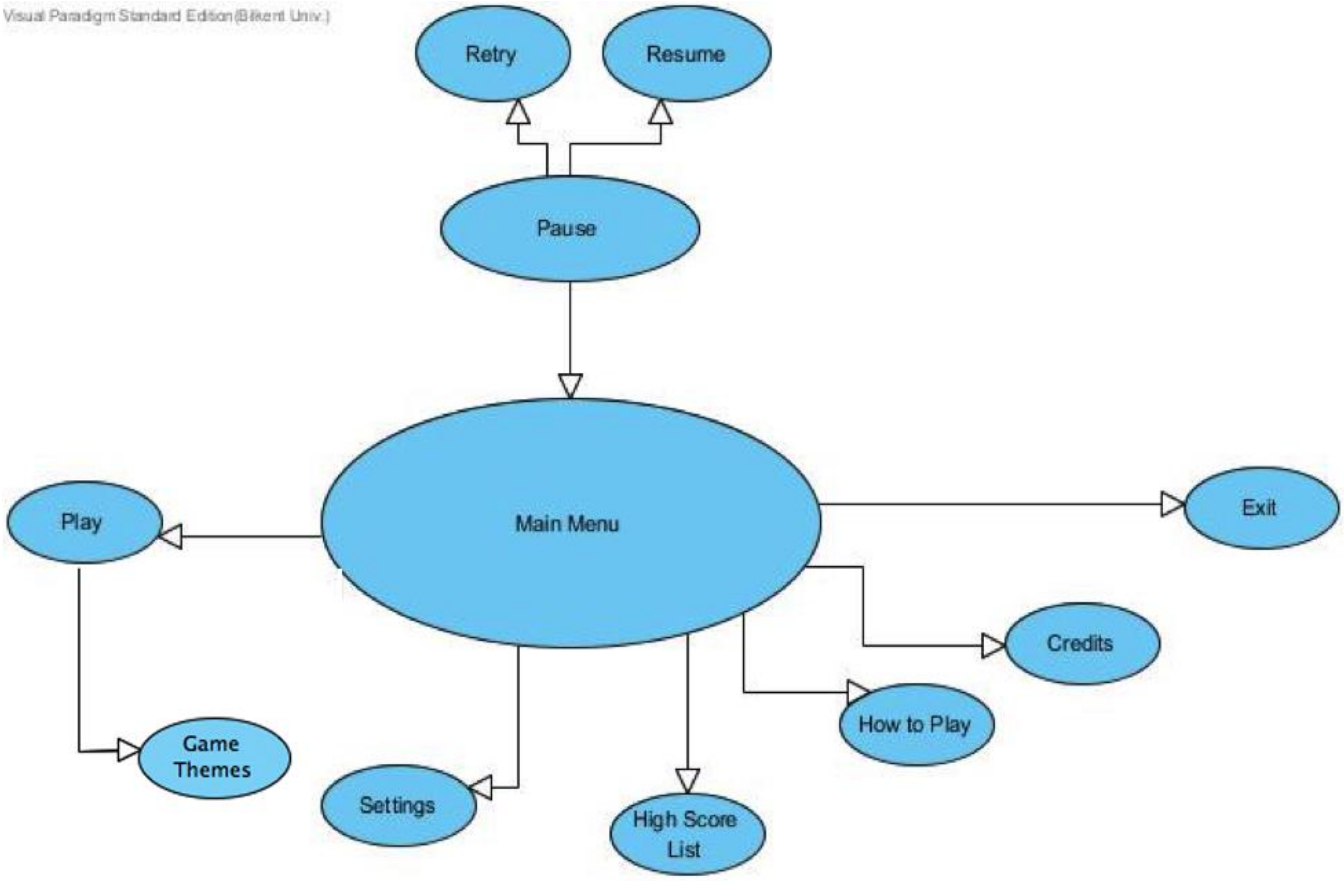

All classes have setters, getters and constructors. But for the sake of simplicity they are not shown in this model.
GameObject
-posX : int
-posY : int
+draw() : void
Miner
-currentState : String
-speed : int
-lives : int
-type : String
Coin
-status : String
-score : int
Monster
-name : String
-speed : int
-state : String
+followMiner(int posX, int posY)
Bonus
-name : String
OriginalMiner
ScreamMiner
SantaMiner
Digger
GoldCoin
Pumpkin
Candy
Diamond
Bag
Tree
ChristmasCandy
SilverCoin
Mole
Snowman
EarthMonster
Ghost
Deer
Nobbin
Hobbin
Zombie
ExtraLife
SilverToGold
DoubleGoldScore
TripleGoldScore
DoubleSilverScore
TripleSilverScore
DigAllMaze
DestroyMonste...
KillMiner
GoldToSilver
DigBack
DoubleMonsters

Object model of Digger Unlimited game is given above 47 classes and the important ones are:

- **GameEngine** is basically the center of all interactions. Updating game is managed here.
- **GameMap** is the class where the current map of game is hold.
- **CollusionDetector** controls the positions of all game objects. If miner and a monster or miner and a coin is at the same place at the same time, this intersection is detected by CollusionDetector and GameEngine is informed by this class.
- **GameObject** is base class of all game objects.
- **InputManager** basically informs GameEngine about the user input.
- **GameGUI** is the class where all the drawings are done.

6. User Interface

6.1. Navigational Path



6.2. Screen Mock-Ups

6.2.1. Main Menu

When the player click on the application, the main menu will be opened. This menu includes six buttons which are Play, Game Themes, High Scores, Settings, How to Play, Credits and Exit.

Figure 1: Main Menu

6.2.1.1. Game Themes

When “Play” button is clicked on, the consept menu shows up which includes Digger Unlimited, Halloween, Christmas and Digger buttons. The players could select which theme they want by using this buttons.

Figure 2: Game Theme

When the game theme is changed, the appearance of miner,monsters, coins and bonuses are changed. The view of game members according to themes:

Digger Unlimited:

 **Life***

 **Mole***

 **Miner***

 **Earth Monster***

 **Gold***

 **Silver***

Halloween:

 **Scream Miner***

 **Ghost***

 **Zombie***

 **Pumpkin***

 **Halloween Candy***

Christmas:

 **Santa Miner***

 **Snowman***

 **Deer***

 **Tree***

 **Christmas Candy***

Digger:

 **Miner***

 **Nobbin***

 **Hobbin***

 **Emerald***

 **Bag***

6.2.1.2. Play

When the players click on “Play” button, the play screen shows up and player chooses a theme, a new game starts. It could be easily fallowed that how many lives the player has and point the player scored from screen.

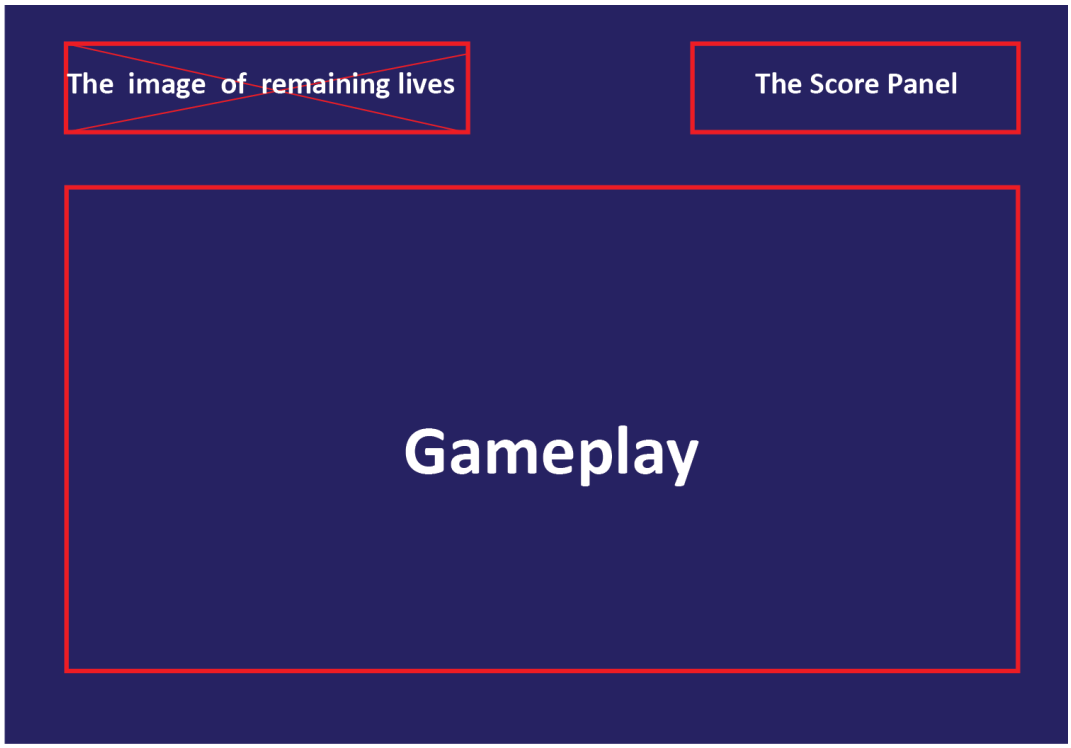


Figure 1: Play Screen

6.2.1.3. High Score List

When the player complete the levels or lose the game the name screen shows up and asks them to write their names. If the player do not want to write it, they could skip and go to high score list directly and the program assign it as “Anonymous” automatically.

Figure 4: Name Screen

The high score list includes the best eight score and players’ scores are displayed on the

Figure 5: High Score List

list whether it is bigger or smaller than others. The players also could look at high score list from Main Menu by clicking on “High Score List” button.

6.2.1.4. Settings

The players could change audio setting by clicking on “Settings” button.

Figure 6: Settings

6.2.1.5. How to Play

Players could learn that how to play the game and the differences between themes, monsters, coins and bonuses by pressing “How to Play” button.

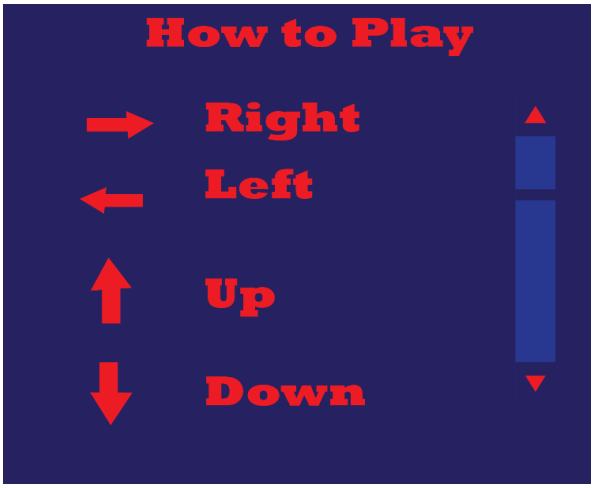


Figure 7: How to Play

6.2.1.6. Credits

Players could reach the name of the game's developers and their contact information by cliking on "Credits" button.

Figure 8: Credits

6.2.1.7. Exit

When "Exit" button is pressed, the game quits and the game window is closed.

6.2.2. Pause Menu

When “esc” button is pressed during game, the “Pause” menu show up and by this menu the players could select 3 different option which are Resume, Retry and Main Menu.

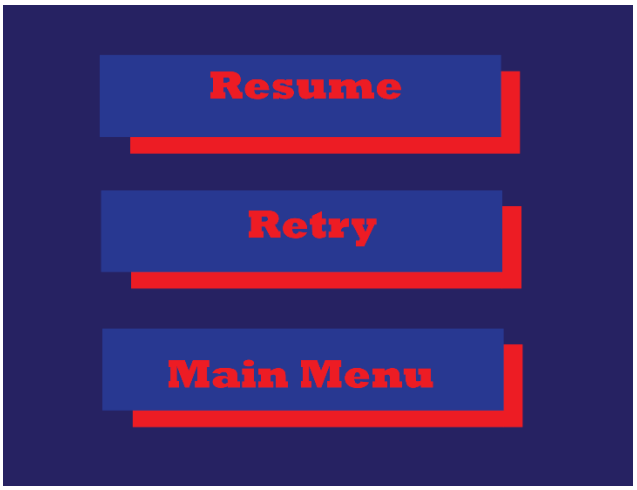


Figure 9: Pause Menu

6.2.3. Bonuses

The bonuses according to themes:

Bonus*

Christmas Bonus*

Halloween Bonus*

Digger Bonus*

7. Conclusion

This analysis report consists of two main parts before going on to design of our remake of "Digger" game, "Digger Unlimited". First, we talked about requirement specification of the project and secondly, we introduced our System Model.

On requirement specification, we covered all requirements, both functional and non-functional, that player can perform. We aimed these requirements to be complementary and not overlap. It was crucial to define requirements completely to prepare a better System Model, so we make sure that we spent significant time upon them.

The System Model of "Digge Unlimited" consists four sections:

- Use Case Model
- Dynamic Models
- Class Model
- User Interface

In Use Case Modelling, first we identify actual actions and requirements and then create use cases to avoid creating unnecessary and meaningless use cases.

In Dynamic Modelling, we use activity, sequence and class diagrams to help us in the design and implementation part. We created sequence diagrams to show interactions between Player and System, and actions of System in specific parts of the game. We covered all possible states, and check it twice before going on to design. For hardest part, explaining the game, we used Activity Diagram. It shows step by step operation of system depending player inputs and give us a better understanding about the game. To show class models, we used

Class Diagram, which will be very helpful later in the project. All possible classes and connections are well though and defined before creating a Class Diagram for it.

In the last part, we introduced our user interface and Navigational Path Diagram, both of them help to create a mock-up on reader's mind and ours. We took most of the interface from "The Digger" game legacy, but included our touch as well. Interface is simple and perfectly ordered with high quality planning.

This report is a basis for our design and implementation, that is why it includes all the materials we will need in the future. It is well-made that could be passed to another group for design and implementation phases and they do not have to deal with extra problems or missing components from analysis.

8. References

All sketches of game is prepared by our group member Cansu Tüzmen.