Cat Vs Covid Game Documentation

Created by

Noppanara   Sakulsaowapakkul   6130638121
Seree   Yengskunpysan   6130640321

2110215 Programming Methodology

Semester 2 Year 2019

Chulalongkorn University

Cat Vs Covid

Introduction

Cat Vs Covid game is inspired from our current situation around the world. Well, except our main protagonist is a cat. The game style is dungeon-type game in which you control your character and goes into the dark dungeon which contain a vaccine that can cure the on going plague of Covid-19

You will need to side step Covid-19 viruses and some glass shards that are dropped on the floor. You will find a door that can't be opened but fear not. Inside this dark dungeon contains some items such as medicines, keys, etc to help you along the way!
Your objective is simple: find the vaccine and save the world!!

Rule

You will control the character through keyboard by pressing

'W' : move the player character up
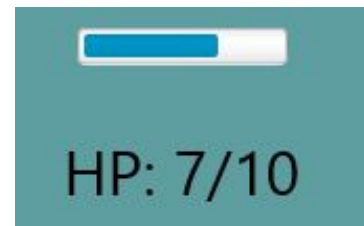'S' : move the player character down
'A' : move the player character left
'D' : move the player character right

Since the dungeon is very dark and can be seen only by holding lanterns, we recommend you pick up the lantern at first sight.
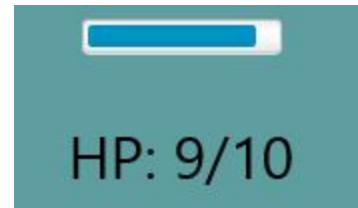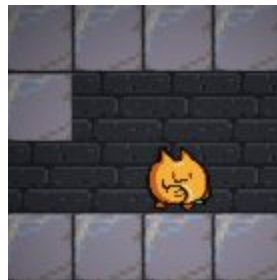

In this image show the player character and lantern.notice that it gets darker when stay away from lantern

If you came upon virus and not sidestep it you will take a considerable damage(3 damage) from your hp

In this image show the player character and virus.after not sidestep it the character get damage by 3

If you come upon glassShard and walk directly into it you will take some damage(1 damage) from your hp.

In this image show the player character and glassShard.after not sidestep it the character get damage by 1

Some items that you can pick up along the way to help you go through this dungeon include:

1.Medicine - if you pick up the medicine.It will heal you by 3 point

2.Alcohol - if you pick up alcohol.It will prevent you from getting damage by virus.

3.Key - if you pick up a key.It will help you open the doors that are in this dungeon.Normally opening the door will lead you to more items to collect!!.

4.Medical Clothes - if you pick up medical clothes.It will prevent you from taking damage by glassShard. There are only 1 clothes in the game hidden somewhere inside the dungeon.

5.Vaccine - your ultimate goal in this game.If you can pick it up before your health reaches 0 you win the game.

In this image show the medicine ,alcohol ,door and the key used to open it.
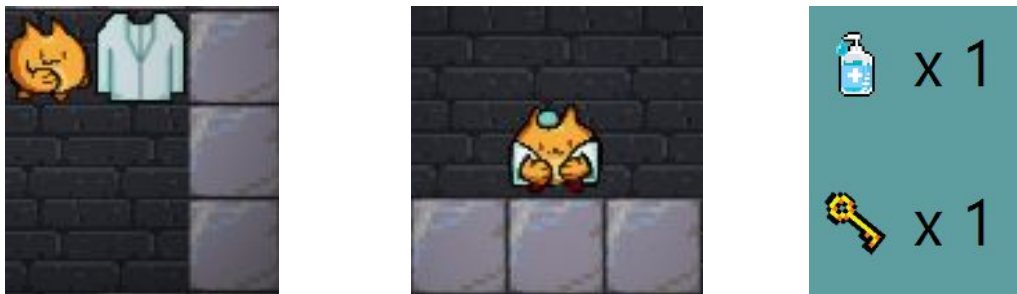
In this image show the medical clothes before and after.We can collect multiple key and alcohol throughout the game

Game will end when you collect the key or your health reach 0
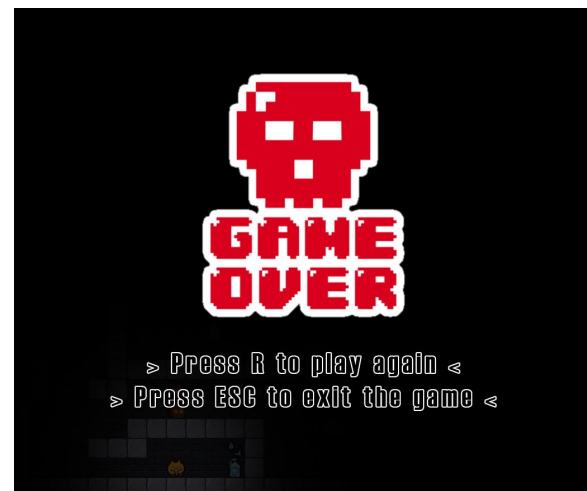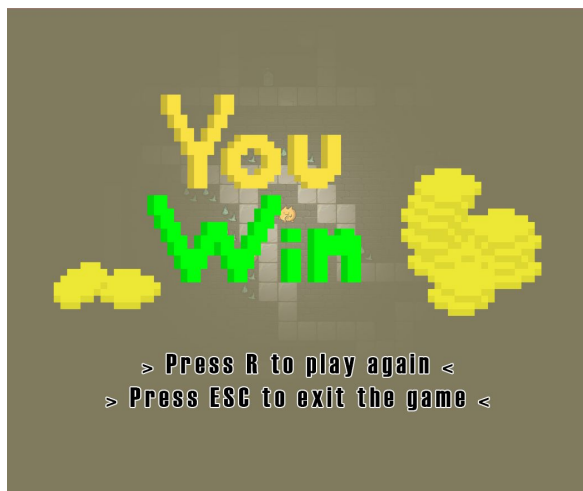
GUI Interface

When you open the game,It will start playing.You can press Esc to exit the game or restart by pressing R.

There will be a panel tab which will show you your health,the amount of alcohol and key you have already collected.

Game will show the ending scene when the game is over.

Game Interface when the game is running.

Class Diagram

Main
<<Java Class>>
Main
application

- gameRoot: GameRoot

- Main()
- start(Stage):void
- main(String[]):void
- addEventListener(Scene,Stage):void

MapCodeInvalidException
<<Java Class>>
MapCodeInvalidException
logic.exception

- serialVersionUID: long

- MapCodeInvalidException(char,int,int)

GameStatusNotFoundException
<<Java Class>>
GameStatusNotFoundException
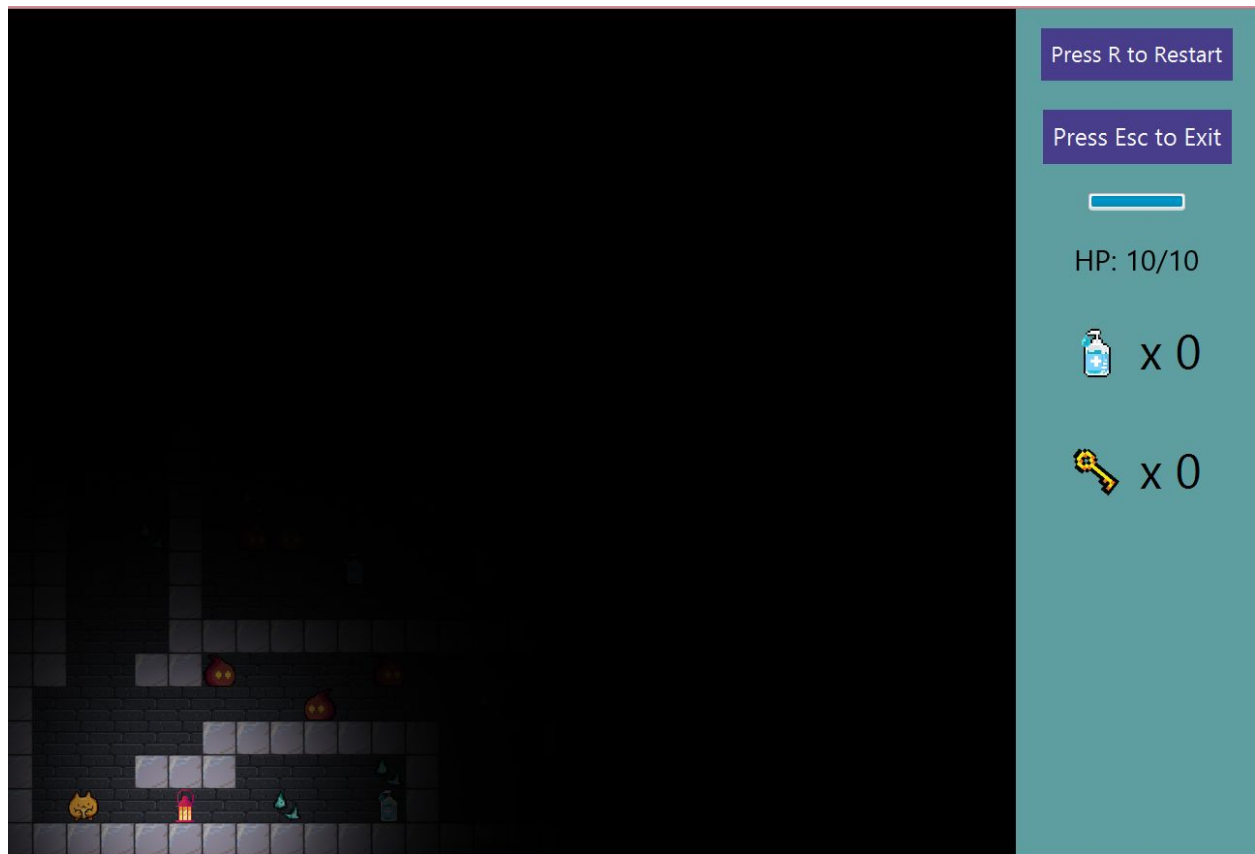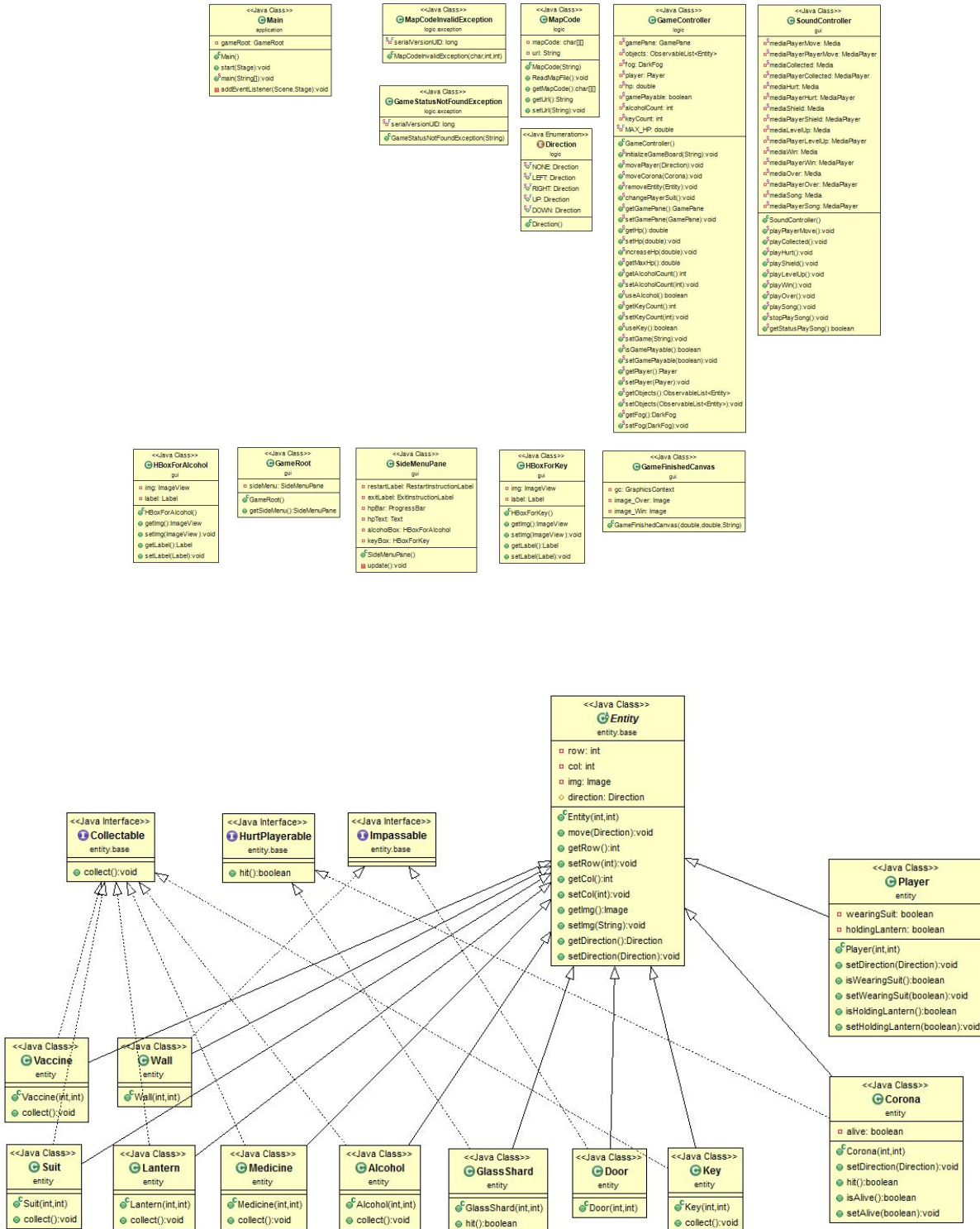logic.exception

- serialVersionUID: long

- GameStatusNotFoundException(String)

MapCode
<<Java Class>>
MapCode
logic

- mapCode: char[][]
- url: String

- MapCode(String)
- ReadMapFile():void
- getMapCode():char[][]
- getUrl():String
- setUrl(String):void

Direction
<<Java Enumeration>>
Direction
logic

- NONE: Direction
- LEFT: Direction
- RIGHT: Direction
- UP: Direction
- DOWN: Direction
- Direction()

GameController
<<Java Class>>
GameController
logic

- gamePane: GamePane
- objects: ObservableList<Entity>
- fog: DarkFog
- player: Player
- hp: double
- gamePlayable: boolean
- alcoholCount: int
- keyCount: int
- MAX_HP: double

- GameController()
- InitializeGameBoard(String):void
- movePlayer(Direction):void
- moveCorona(Corona):void
- removeEntity(Entity):void
- changePlayerSuit():void
- getGamePane():GamePane
- setGamePane(GamePane):void
- getHp():double
- setHp(double):void
- increaseHp(double):void
- getMaxHp():double
- getAlcoholCount():int
- setAlcoholCount(int):void
- useAlcohol():boolean
- getKeyCount():int
- setKeyCount(int):void
- useKey():boolean
- setGame(String):void
- isGamePlayable():boolean
- setGamePlayable(boolean):void
- getPlayer():Player
- setPlayer(Player):void
- getObjects():ObservableList<Entity>
- setObjects(ObservableList<Entity>):void
- getFog():DarkFog
- setFog(DarkFog):void

SoundController
<<Java Class>>
SoundController
gui

- mediaPlayerMove: Media
- mediaPlayerPlayerMove: MediaPlayer
- mediaCollected: Media
- mediaPlayerCollected: MediaPlayer
- mediaHurt: Media
- mediaPlayerHurt: MediaPlayer
- mediaShield: Media
- mediaPlayerShield: MediaPlayer
- mediaLevelUp: Media
- mediaPlayerLevelUp: MediaPlayer
- mediaWin: Media
- mediaPlayerWin: MediaPlayer
- mediaOver: Media
- mediaPlayerOver: MediaPlayer
- mediaSong: Media
- mediaPlayerSong: MediaPlayer

- SoundController()
- playPlayerMove():void
- playCollected():void
- playHurt():void
- playShield():void
- playLevelUp():void
- playWin():void
- playOver():void
- playSong():void
- stopPlaySong():void
- getStatusPlaySong():boolean

HBoxForAlcohol
<<Java Class>>
HBoxForAlcohol
gui

- img: ImageView
- label: Label

- HBoxForAlcohol()
- getImg():ImageView
- setImg(ImageView):void
- getLabel():Label
- setLabel(Label):void

GameRoot
<<Java Class>>
GameRoot
gui

- sideMenu: SideMenuPane

- GameRoot()
- getSideMenu():SideMenuPane

SideMenuPane
<<Java Class>>
SideMenuPane
gui

- restartLabel: RestartInstructionLabel
- exitLabel: ExitInstructionLabel
- hpBar: ProgressBar
- hpText: Text
- alcoholBox: HBoxForAlcohol
- keyBox: HBoxForKey

- SideMenuPane()
- update():void

HBoxForKey
<<Java Class>>
HBoxForKey
gui

- img: ImageView
- label: Label

- HBoxForKey()
- getImg():ImageView
- setImg(ImageView):void
- getLabel():Label
- setLabel(Label):void

GameFinishedCanvas
<<Java Class>>
GameFinishedCanvas
gui

- gc: GraphicsContext
- image_Over: Image
- image_Win: Image

- GameFinishedCanvas(double,double,String)

Entity
<<Java Class>>
Entity
entity.base

- row: int
- col: int
- img: Image
- direction: Direction

- Entity(int,int)
- move(Direction):void
- getRow():int
- setRow(int):void
- getCol():int
- setCol(int):void
- getImg():Image
- setImg(String):void
- getDirection():Direction
- setDirection(Direction):void

Collectable
<<Java Interface>>
Collectable
entity.base

- collect():void

HurtPlayerable
<<Java Interface>>
HurtPlayerable
entity.base

- hit():boolean

Impassable
<<Java Interface>>
Impassable
entity.base

Player
<<Java Class>>
Player
entity

- wearingSuit: boolean
- holdingLantern: boolean

- Player(int,int)
- setDirection(Direction):void
- isWearingSuit():boolean
- setWearingSuit(boolean):void
- isHoldingLantern():boolean
- setHoldingLantern(boolean):void

Vaccine
<<Java Class>>
Vaccine
entity

- Vaccine(int,int)
- collect():void

Wall
<<Java Class>>
Wall
entity

- Wall(int,int)

Suit
<<Java Class>>
Suit
entity

- Suit(int,int)
- collect():void

Lantern
<<Java Class>>
Lantern
entity

- Lantern(int,int)
- collect():void

Medicine
<<Java Class>>
Medicine
entity

- Medicine(int,int)
- collect():void

Alcohol
<<Java Class>>
Alcohol
entity

- Alcohol(int,int)
- collect():void

GlassShard
<<Java Class>>
GlassShard
entity

- GlassShard(int,int)
- hit():boolean

Door
<<Java Class>>
Door
entity

- Door(int,int)

Key
<<Java Class>>
Key
entity

- Key(int,int)
- collect():void

Corona
<<Java Class>>
Corona
entity

- alive: boolean

- Corona(int,int)
- setDirection(Direction):void
- hit():boolean
- isAlive():boolean
- setAlive(boolean):void

1. Package application

1.1 class Main

1.1.1 Fields

- gameRoot	- Root pane

1.1.2 Methods

+ static void main(strings[] args)	- Main application
+ void start(Stage PrimaryStage)	- Initialize fields - Show primarystage - Add event listener
- void addEventListener(Scene s, Stage PrimaryStage)	- Set key press - W, A, S, D for moving up, left, down, right - R to restart game - E to exit

2. Package entity.base

2.1 interface Collectable

2.1.1 Method

+ void collect()	- Collect method for all collectable item

2.2 interface HurtPlayerable

2.2.1 Method

+ boolean hit()	- Hit method for all HurtPlayerable item

2.3 interface Impassable

2.4 abstract class Entity extends Rectangle

2.4.1 Fields

- int row	- Position in row for entity
- int col	- Position in column for entity
- Image img	- Image to represent entity
# Direction direction	- Current direction of entity

2.4.2 Constructor

+ Entity(int col, int row)	- Initialize entity as Rectangle with size 35x35 - Initialize col, row as given - Set direction as NONE

2.4.3 Methods

+ void move(Direction dir)	- Set direction to dir - Move itself 1 block according

	to that direction
+ void setRow(int row)	- Set its row on game board
+ void setCol(int col)	- Set its column on game board
+ void setImg(String url)	- Set its image using url
+ getter/setter of each fields	

3. Package entity

3.1 class Alcohol extends Entity implements Collectable

3.1.1 Constructor

+ Alcohol(int col, int row)	- Initialize entity - Set its image

3.1.2 Method

+ void collect()	- Remove this item from game board - Increase alcoholCount in GameController by 1

3.2 class Corona extends Entity implements HurtPlayerable

3.2.1 Field

- boolean alive	- True if this Corona is alive

3.2.2 Constructor

+ Corona(int col, int row)	- Initialize entity - Set its image - Set alive to true

3.2.3 Methods

+ void setDirection(Direction direction)	- Set direction to the given direction - Set its image according to its direction
+ boolean hit()	- Set alive to false - Remove this item from game board - If player have alcohol, use alcohol. Else if player don't have alcohol, decrease player's hp by 3
+ getter/setter for alive field	

3.3 class Door extends Entity implements Impassable

3.3.1 Constructor

+ Door(int col, int row)	- Initialize entity - Set its image

3.4 class GlassShard extends Entity implements HurtPlayerable

3.4.1 Constructor

+ GlassShard(int col, int row)	- Initialize entity - Set its image

3.4.2 Method

+ boolean hit()	- If player isn't wearing suit, decrease player's hp by 1

3.5 class Key extends Entity implements Collectable

3.5.1 Constructor

+ Key(int col, int row)	- Initialize entity - Set its image

3.5.2 Method

+ void collect()	- Remove this item from game board - Increase keyCount in GameController by 1

3.6 class Lantern extends Entity implements Collectable

3.6.1 Constructor

+ Lantern(int col, int row)	- Initialize entity - Set its image

3.6.2 Method

+ void collect()	- Remove this item from game board - Set player's holdingLantern to true

3.7 class Medicine extends Entity implements Collectable

3.7.1 Constructor

+ Medicine(int col, int row)	- Initialize entity - Set its image

3.7.2 Method

+ void collect()	- Remove this item from game board - Increase player's hp by 3

3.8 class Player extends Entity

3.8.1 Fields

- boolean wearingSuit	- True if player already collected suit
- boolean holdingLantern	- True if player already collected lantern

3.8.2 Constructor

+ Player(int col, int row)	- Initialize entity - Set wearingSuit to false - Set holdingLantern to false - Set its image

3.8.3 Method

+ void setDirection(Direction direction)	- Set direction to the given direction - Set its image according to its direction, appearance
+ getter/setter for each fields	

3.9 class Suit extends Entity implements Collectable

3.9.1 Constructor

+ Suit(int col, int row)	- Initialize entity - Set its image

3.9.2 Method

+ void collect()	- Remove this item from game board - Make player wear suit via GameController

3.10 class Vaccine extends Entity implements Collectable

3.10.1 Constructor

+ Vaccine(int col, int row)	- Initialize entity

	- Set its image

3.10.2 Method

+ void collect()	- Remove this item from game board - Set game to "Win" using setGame method in GameController

3.11 class Wall extends Entity implements Impassable

3.11.1 Constructor

+ Wall(int col, int row)	- Initialize entity - Set its image

4. Package gui

4.1 class DarkFog extends Canvas

4.1.1 Fields

- GraphicsContext gc	- Graphic context of this canvas
- Image img	- Contain Image of the dark fog
- int col	- Number of the column that the center of the dark fog is at

- int row	- Number of the row that the center of the dark fog is at

4.1.2 Constructor

+ DarkFog(int col, int row)	- Initialize Canvas - Set Image and draw image in GraphicContext gc - Set column and row as given

4.1.3 Method

+ void followingPlayer()	- Set column and row as player's column and row
+ getter/setter for col and row fields	

4.2 class ExitInstructionLabel extends Label

4.2.1 Constructor

+ ExitInstructionLabel()	- Initialize Label with text "Press Esc to Exit" - Set text to size 25, alignment center and color GHOSTWHITE - Set background's color to DARKSLATEBLUE - Set Insets to 10

4.3 class GameFinishedCanvas extends Canvas

4.3.1 Fields

- GraphicsContext gc	- Graphic context of this canvas
- Image image_Over	- Contain Image of the game over's image
- Image image_Win	- Contain Image of the game winning's image

4.3.2 Constructor

+ GameFinishedCanvas(double width, double height, String status) throws GameStatusNotFound Exception	- Set width and height as given - Set Image image_Over, image_Win - If status is "Win", draw image_Win on gc. Else if status is "Lose", draw image_Over on gc. Otherwise, throw exception GameStatusNotFoundExcepti on

4.4 class GamePane extends Pane

4.4.1 Constructor

+ GamePane()	- Set pref size to 1050x875

4.5 class GameRoot extends HBox

4.5.1 Field

- SideMenuPane sideMenu	- Contain SideMenuPane

4.5.2 Constructor

+ GameRoot()	- Initialize game board with GameController using Map1.txt - Initialize sideMenu - Add game board and sideMenu to this root

4.5.3 Method

+ SideMenuPane getSideMenu()	- Getter for sideMenu

4.6 class HBoxForAlcohol extends HBox

4.6.1 Fields

- ImageView img	- Contains ImageView of alcohol item
- Label label	- Contains text showing the amount alcohol

4.6.2 Constructor

+ HBoxForAlcohol()	- Initialize HBox - Set insets 10, spacing 20, alignment center - Initialize img, set height to 50, set preserve ratio to true - Initialize label with text "x " + amount of alcohol player have, font size 50 and alignment center

	- Add img, label to this root

4.6.3 Method

+ getter/setter for each fields	

4.7 class HBoxForKey extends HBox

4.7.1 Fields

- ImageView img	- Contains ImageView of key item
- Label label	- Contains text showing the amount key

4.7.2 Constructor

+ HBoxForKey()	- Initialize HBoxForKey - Set insets 10, spacing 20 - Initialize img, set height to 50, set preserve ratio to true - Initialize label with text "x " + amount of key count have, font size 50 and alignment center - Add img, label to this root

4.7.3 Method

+ getter/setter for each fields	

4.8 class RestartInstructionLabel extends Label

4.8.1 Constructor

+ RestartInstructionLabel()	- Initialize Label with text "Press R to Restart" - Set text to size 23, alignment center and color GHOSTWHITE - Set background's color to DARKSLATEBLUE - Set Insets to 10

4.9 class SideMenuPane extends VBox

4.9.1 Fields

- RestartInstructionLabel restartLabel	- Contains RestartInstructionLabel
- ExitInstructionLabel exitLabel	- Contains ExitInstructionLabel
- ProgressBar hpBar	- Contains ProgressBar that show player's hp
- Text hpText	- Contains Text that show player's hp
- HBoxForAlcohol alcoholBox	- Contains HBoxForAlcohol to show the amount of alcohol player have
- HBoxForKey keyBox	- Contains HBoxForKey to show the amount of key player have

4.9.2 Constructor

+ SideMenuPane()	- Set alignment to top center, insets 20, width 250, spacing 30 and background's color CADETBLUE - Initialize all fields - Set hpBar size to 500x500 - Set hpText font size to 30 - Add restartLabel, exitLabel, hpBar, hpText, alcoholBox, keyBox to this root, respectively - Initialize AnimationTimer to do update() and start it

4.9.3 Method

+ void update()	- Set progress of hpBar to current hp divided by max hp - Set hpText to show current hp and max hp - Set alcoholBox to show current amount of alcohol player have - Set keyBox to show current amount of key player have

4.10 class SoundController

4.10.1 Fields

- static Media mediaPlayerMove	- Contains mediaPlayerMove from Player_move.mp3

- static MediaPlayer mediaPlayerPlayerMove	- Contains mediaPlayerPlayerMove from mediaPlayerMove
- static Media mediaCollected	- Contains mediaCollected from Get_Item_Sound.mp3
- static MediaPlayer mediaPlayerCollected	- Contains mediaPlayerCollected from mediaCollected
- static Media mediaHurt	- Contains mediaPlayerMove from Take_Damage_Sound.mp3
- static MediaPlayer mediaPlayerHurt	- Contains mediaPlayerHurt from mediaHurt
- static Media mediaShield	- Contains mediaShield from Shield_Sound.mp3
- static Media mediaLevelUp	- Contains mediaLevelUp from LevelUp_Sound.mp3
- static MediaPlayer mediaPlayerShield	- Contains mediaShield from mediaPlayerShield
- static MediaPlayer mediaPlayerLevelUp	- Contains mediaPlayerLevelUp from mediaLevelUp
- static Media mediaWin	- Contains mediaWin from Win_Game_Sound.mp3
- static MediaPlayer mediaPlayerWin	- Contains mediaPlayerWin from mediaWin
- static Media mediaOver	- Contains mediaOver from Game_Over_Sound_Ver2.mp3
- static MediaPlayer	- Contains mediaPlayerOver

mediaPlayerOver	from mediaOver
- static Media mediaSong	- Contains mediaSong from Background_Song.mp4
- static MediaPlayer mediaPlayerSong	- Contains mediaPlayerSong from mediaSong

4.10.2 Methods

+ static void playPlayerMove()	- Set mediaPlayerPlayerMove to 0 second - Play mediaPlayerPlayerMove
+ static void playCollected()	- Set mediaPlayerCollected to 0 second - Play mediaPlayerCollected
+ static void playHurt()	- Set mediaPlayerHurt to 0 second - Play mediaPlayerHurt
+ static void playShield()	- Set mediaPlayerShield to 0 second - Play mediaPlayerShield
+ static void playLevelUp()	- Set mediaPlayerLevelUp to 0 second - Play mediaPlayerLevelUp
+ static void playWin()	- Set mediaPlayerWin to 0 second - Play mediaPlayerWin
+ static void playOver()	- Set mediaPlayerOver to 0 second - Play mediaPlayerOver
+ static void playSong()	- Set mediaPlayerSong to 0 second

	- Play mediaPlayerSong
+ static void stopPlaySong()	- Stop mediaPlayerSong
+ static boolean getStatusPlaySong()	- Get boolean of status of mediaPlayerSong and return true whether it's status is PLAYING

5. Package logic

5.1 enum Direction

5.1.1 enum
NONE, LEFT, RIGHT, UP, DOWN

5.2 class GameController

5.2.1 Fields

- static GamePane gamePane	- Contains static game board
- static ObservableList<Entity> objects	- List of all entity on gamePane
- static DarkFog fog	- Contains DarkFog
- static Player player	- Contains Player
- static double hp	- Number of player's current hp
- static boolean gamePlayable	- True if game is playable,

	game is not playable when player's already win or lose
- static int alcoholCount	- Number of the amount of alcohol player's collected
- static int keyCount	- Number of the amount of key player's collected
- static final double MAX_HP	- Final number of player's max hp

5.2.2 Methods

+ static void InitializeGameBoard(String mapUrl) throws MapCodeInvalidException	- Initialize all fields
	- set gamePlayable to true
	- Add canvas of game's floor to gamePane
	- Create MapCode from mapUrl, read each char, add new Entity to each col, row according to the alphabet
	    - 'B' = Blank space
	    - 'W' = Wall
	    - 'V' = Vaccine
	    - 'C' = Corona
	    - 'A' = Alcohol
	    - 'M' = Medicine
	    - 'L' = Lantern
	    - 'G' = GlassShard
	    - 'K' = Key
	    - 'D' = Door
	    - 'S' = Suit
	    - 'P' = Player
	    - If the alphabet is not in the list, MapCodeInvalidE

	xception will be thrown - Also add each entity to objects - If the entity is Lantern, also initialize fog at that col, row - Add every entity in objects except Corona to gamePane first - Then add every Corona in objects, player and fog to gamePane afterward, respectively - Set hp to 10, alcoholCount, keyCount to 0 - Play PlayerMove sound, Song sound from SoundController - moveCorona(Corona) for every Corona in objects
+ static void movePlayer(Direction direction)	- If game is not playable, end the method - Make player move to the given direction - If player intersects with Impassable thing, player will move back. Else if player intersects with Collectable thing, it will be collected. Else if player intersects with HurtPlayerAble thing, it

	will hit. - If player is holding a lantern, fog will follow player using followingPlayer()
+ static void moveCorona(Corona corona)	- Using AnimationTimer to make corona move at random speed and move with random direction - If corona intersects with player, corona will hit. Else if corona intersects with Impassable entity, it will move back - If corona is not alive or game is not playable, AnimationTimer will stop
+ static void moveCorona(Corona corona)	- Inside animationTimer t It will do as follow - Initialize last update as the time of it's last update and set it to 0 - Initialize timePerMove which will get a random variable which will decide when Corona entity will move - Inside it's handle will do the following - If Corona entity is not

	alive and Game has already end current thread will stop - If the current time minus last time that this entity is update is bigger than timePerMove variable it will set a random direction to Corona entity - Move the Corona entity to the given Direction - If Corona entity come apart with player will call playHurt() function from soundController and damage player per corona.hit() function - If corona entity come apart with object that is Impassable it will stop moving in that Direction and call moveCorona function again - Run t thread
+ static void removeEntity(Entity e)	- Remove e from gamePane and objects
+ static void changePlayerSuit()	- Set player's wearingSuit to true
+ static void setHp(double hp)	- Set player's hp to given hp, hp must not be lower than 0 or greater than MAX_HP - If player's hp is 0, set

	game "Lose"
+ static void increaseHp(double hp)	- Set player's hp to current player's hp plus given hp
+ static void setAlcoholCount(int alcoholCount)	- Set alcoholCount to the given alcoholCount, must not be lower than 0
+ static boolean useAlcohol()	- If alcoholCount = 0, return false - Else, decrease alcoholCount by 1 and return true
+ static void setKeyCount(int keyCount)	- Set keyCount to the given keyCount, must not be lower than 0
+ static boolean useKey()	- If keyCount = 0, return false - Else, decrease keyCount by 1 and return true
+ static void setGame(String status)	- Initialize GameFinishedCanvas finishedCanvas with gamPane's size and status - Wait for 1 second, using thread - Add finishedCanvas to gamePane - Set gamePlayable to false
+ getter/setter for each fields	

5.3 class MapCode

5.3.1 Fields

- char[][] mapCode	- 2 dimensional array that contains char
- String url	- String url of text file

5.3.2 Constructor

+ MapCode(String url)	- Initialize mapCode as 25x30 char array - Read the MapFile to MapCode using ReadMapFile() function

5.3.3 Methods

+ void ReadMapFile()	- Read text file from given url - Extract each char in text file into mapCode - Throw IOException with text "Error: Map file not found." if file not found
+ getter/setter for each fields	

6. Package logic.exception

6.1 class GameStatusNotFoundException extends Exception

6.1.1 Field

- static final long serialVersionUID	- Contain serialVersionUID as "1L"

6.1.2 Constructor

+ GameStatusNotFoundExcepti on(String status)	- Print out "The status \" + status + "\ is invalid."

6.2 class MapCodeInvalidException extends Exception

6.2.1 Field

- static final long serialVersionUID	- Contain serialVersionUID as "1L"

6.2.2 Constructor

+ MapCodeInvalidException(cha r c, int row, int col)	- Print out "Code \"" + c + "\" at " + col + "x" + row +" (column x row) in map code is invalid." "There's no entity for code \"" + c + "\"."