

ChueRaiDee Documentation

Created by

Sarunpong Lapsuksathit 6432163521

Supachok Angsirikulchote 6430386821

2110215 Programming Methodology
Semester 1/2022

Department of Computer Engineering
Faculty of Engineering
Chulalongkorn University



Introduction

IntaniaGhostRush was inspired by two famous action games. The first one that appears when your internet is disconnected is named “Chrome Dino” on Google Chrome, and the second one that was installed on Nokia cellphones is named “Bounce” by Rovio Entertainment. The story of IntaniaGhostRush is about a CP48 student who rushed ProgMethProject too hard until he died. He wanted to submit his project to his professor, but the professor was afraid of him. Hence, the professor hired a multitude of holy monks to protect him from this student. The objective of this game is to keep your intania ghost safe from the holy monks along the journey and succeed in submitting his ProgMethProject.

Rules

When the game has begun, the ghost starts rushing and the player can press either “W”, “A”, “D”, or the spacebar button to control their ghost leaps over incoming obstacles and the holy monks without crashing into a monk. If their ghost bumps into a monk, the game ends. Else if their ghost reaches his professor, the game also ends.

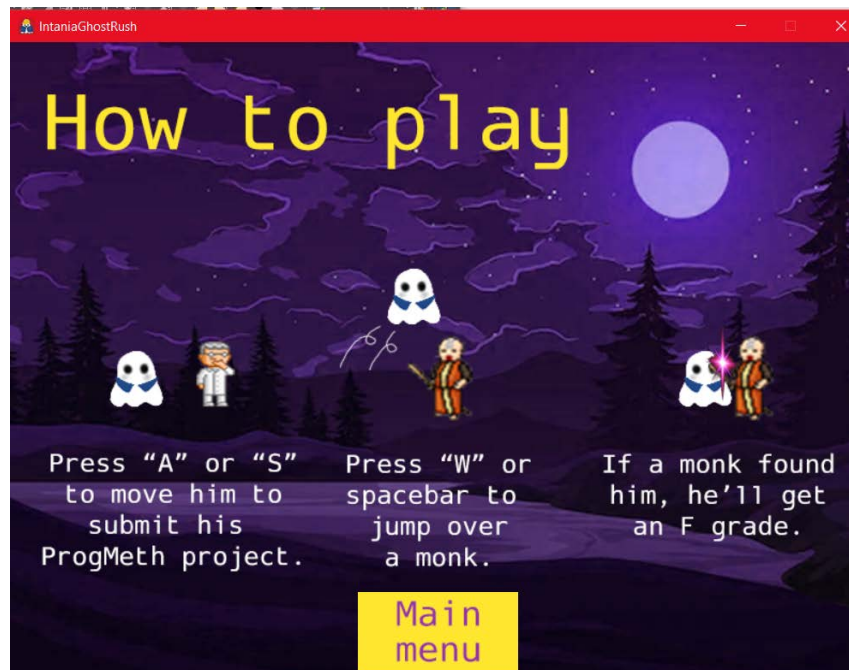
Gameplay

Once the player launches the game, here comes the main menu.

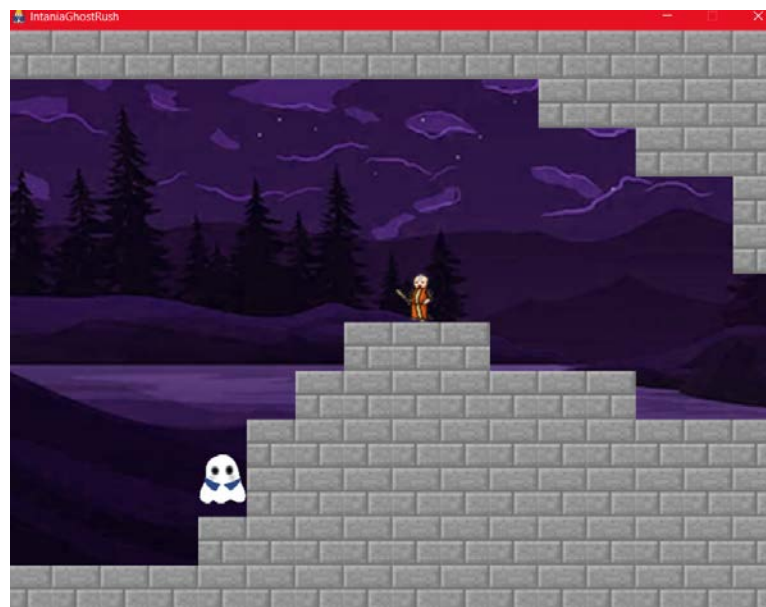


If they click the “How to play?” button, here come the instructions.

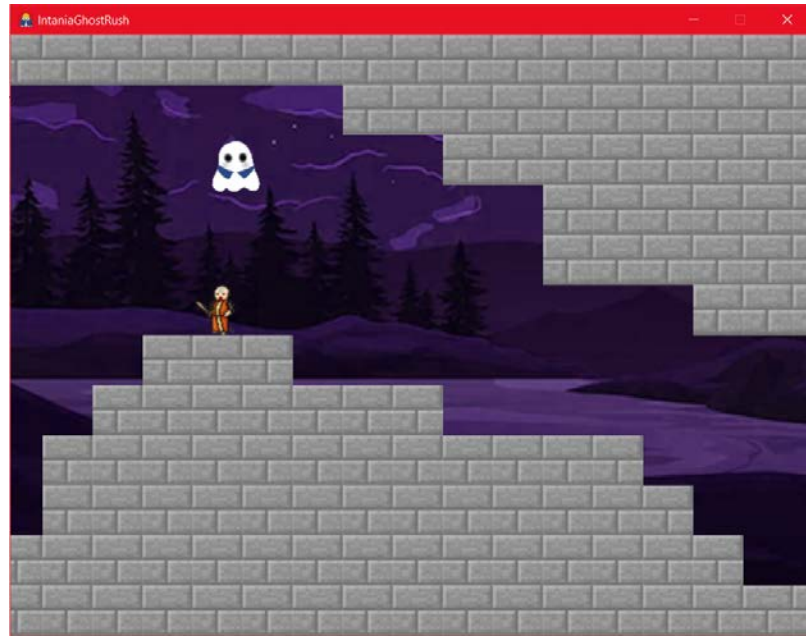




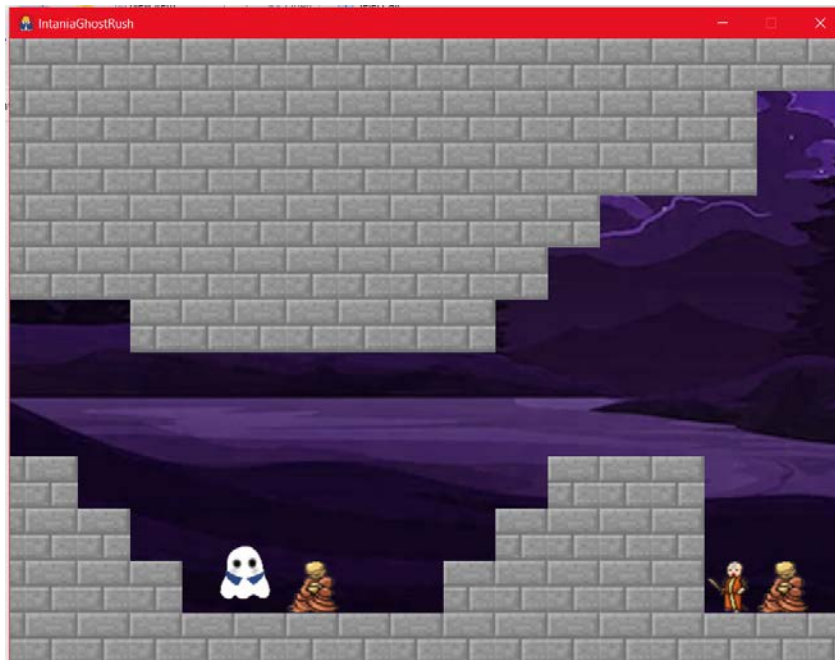
The instructions consist of two pages, the first page tells the player the background story of the game, and the second page tells the player how to control the IntaniaGhost.

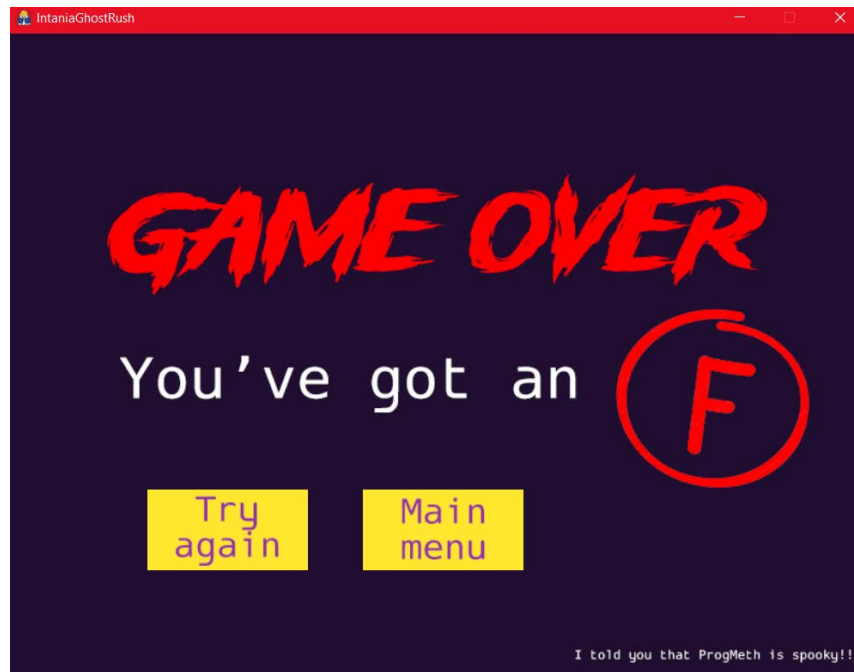


When they click the play button, the game started. There are a few maps randomized each turn. They can press "A" or "D" to move left or right.

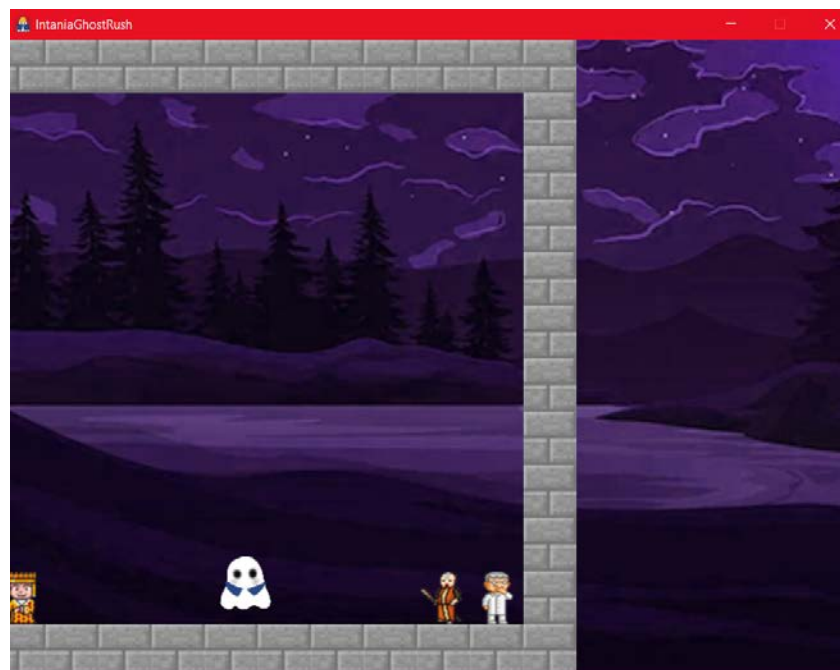


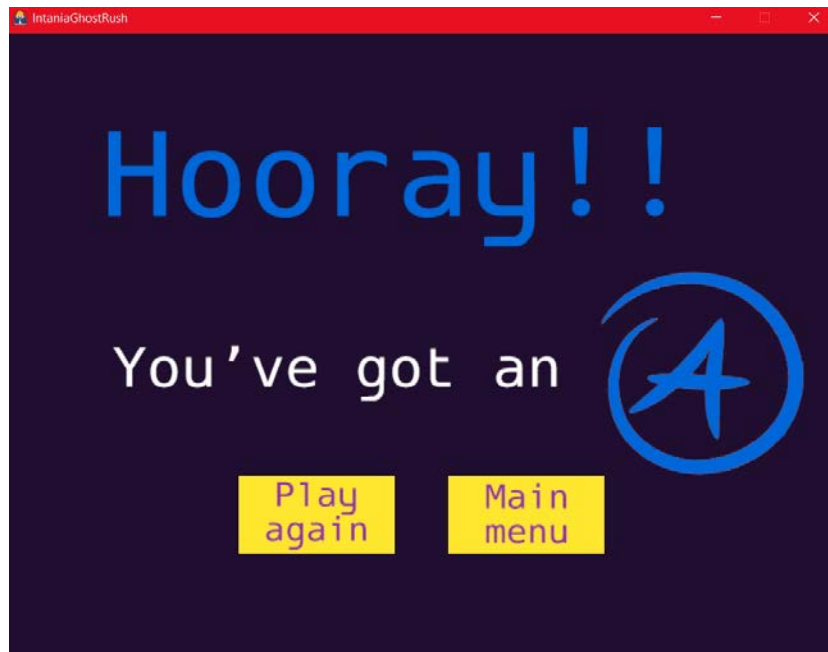
They can also press “W” or the spacebar button to jump over obstacles and monks (you know? ghosts are afraid of monks!).





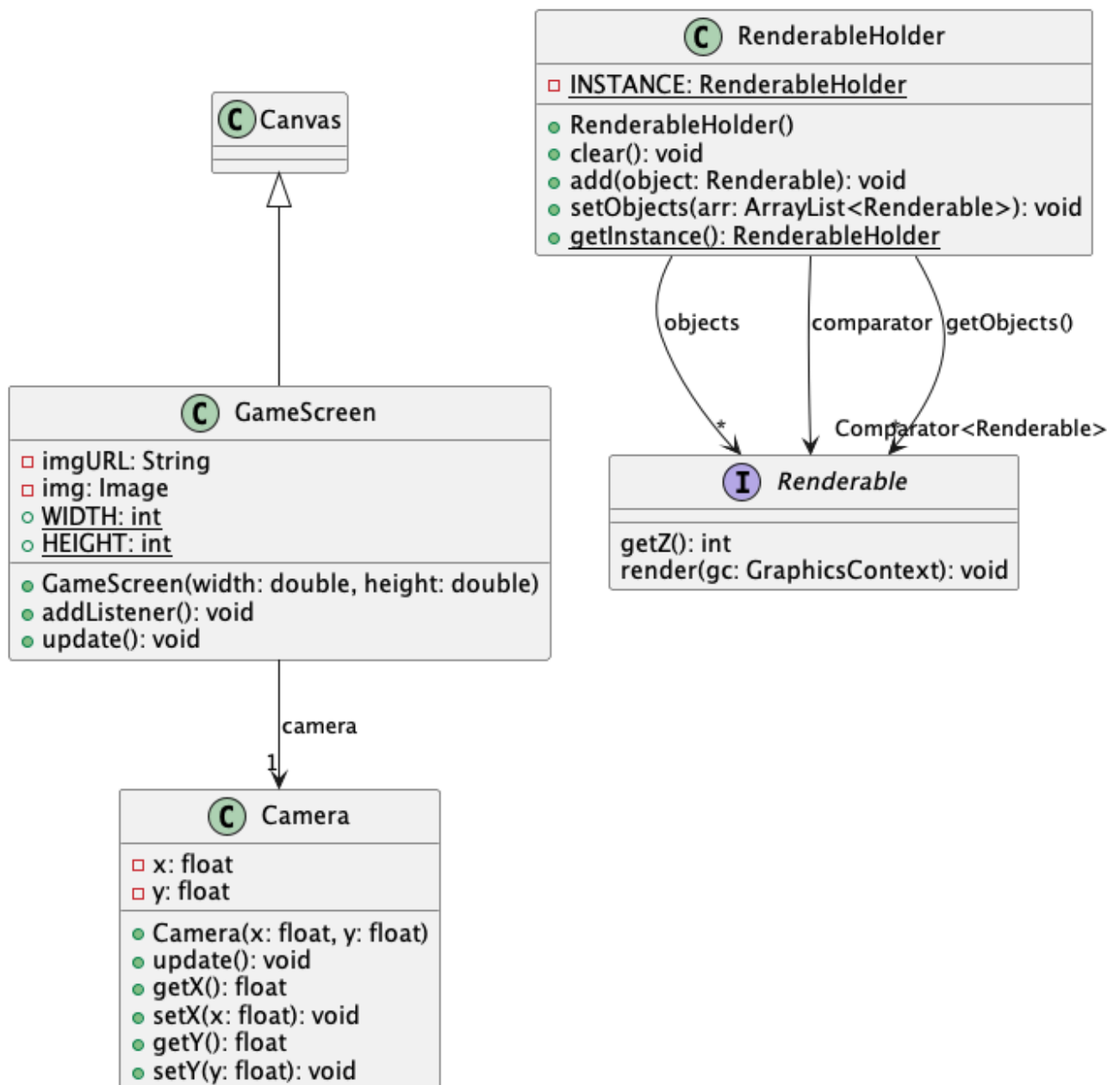
If the IntaniaGhost bumps into a monk, he will be doomed (get an F). Here comes the game-over pane. They can either try again or give in and back to the main menu.



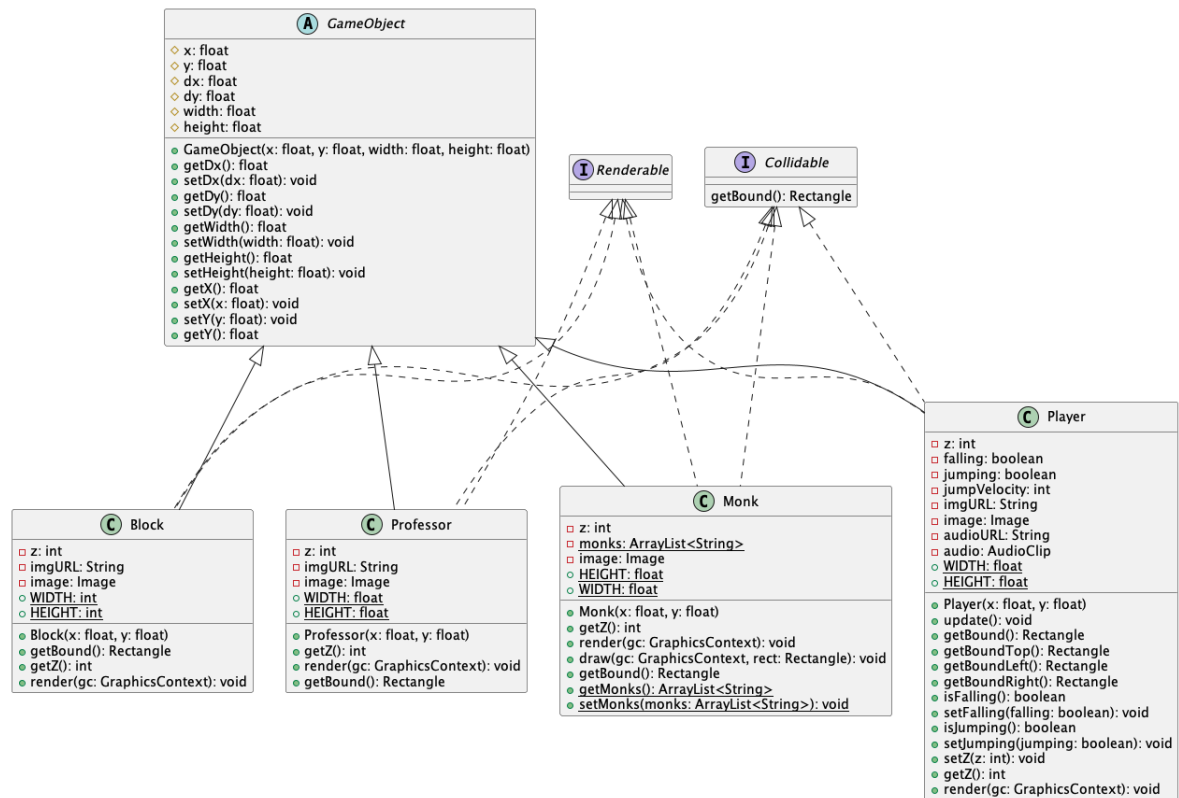


If the IntaniaGhost can reach his professor and submit his ProgMeth project in time, he will get an A and go to paradise for eternity.

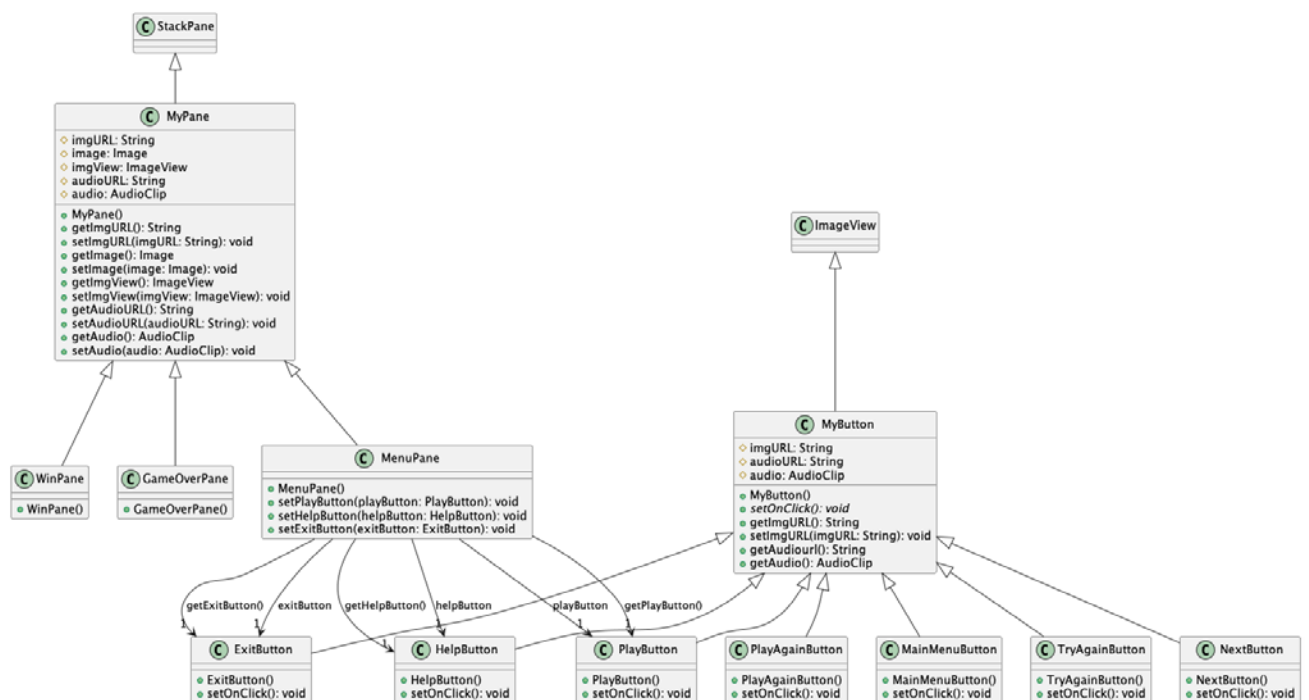
Project Class Diagram



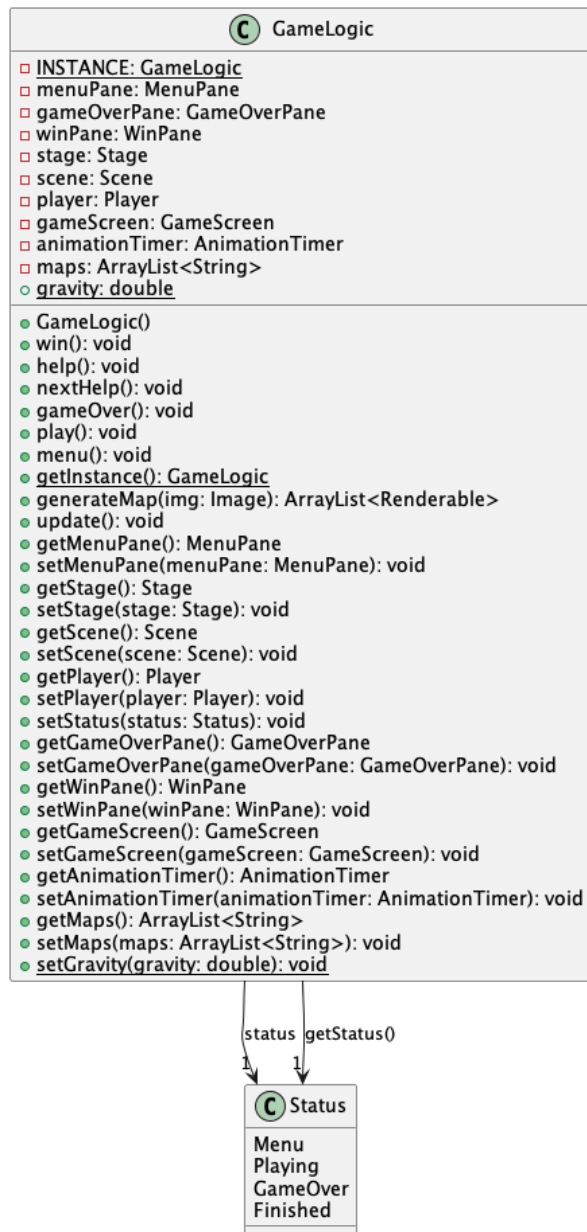
This is the class diagram of Package rendering.



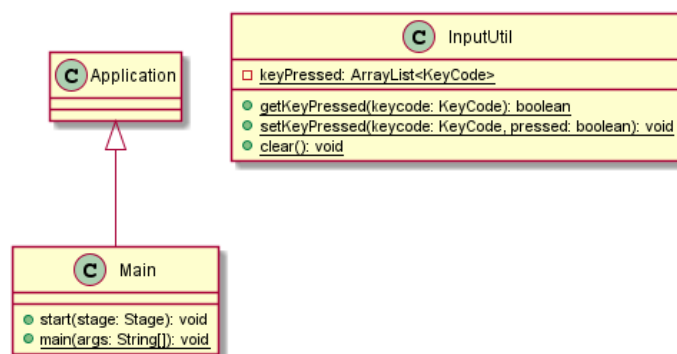
This is the class diagram of Package objects which involves the abstract class design and interface.



This is the class diagram of Package menuComponents which involves the inheritance class design.



This is the class diagram of Package logic.



This is the class diagram of Package InputUtil and Package application.



Implementation Details

Folder src

1) Package rendering

1.1) Interface Renderable

1.1.1) Methods

| | |
|--------------------------------------|-------------------------------------|
| + int getZ() | get Z dimension |
| + void render(GraphicsContext gc) | render object on GraphicsContext |

1.2) Class RenderableHolder

1.2.1) Fields

| | |
|--|---|
| - ArrayList<Renderable> objects | store Renderable objects |
| - <u>RenderableHolder</u> <u>INSTANCE</u> | get the instance of RenderableHolder |
| - Comparator<Renderable> comparator | compare |

1.2.2) Constructor

| | |
|----------------------|--------------------------------------|
| + RenderableHolder() | initialize objects and comparator |
|----------------------|--------------------------------------|

1.2.3) Methods

| | |
|--------------------------|--------------------------|
| + void clear() | clear renderable objects |
| + add(Renderable object) | add renderable object |

| | |
|---|---|
| + ArrayList<Renderable> getObjects() | get ArrayList of all renderable objects |
| + void setObjects(ArrayList<Renderable> arr) | set renderable objects to given array |
| + <u>RenderableHolder</u> <u>getInstance()</u> | get instance of RenderableHolder |

1.3) Class Camera

1.3.1) Fields

| | |
|-----------|--------------------------|
| - float x | x position of the camera |
| - float y | y position of the camera |

1.3.2) Constructor

| | |
|----------------------------|--|
| + Camera(float x, float y) | initialize camera with given x, y coordinate |
|----------------------------|--|

1.3.3) Methods

| | |
|----------------------------------|--|
| + void update() | update camera corresponding to player coordinate |
| + getter & setter for all fields | |

1.4) Class GameScreen extends Canvas

1.4.1) Fields

| | |
|-----------------|----------------------------|
| - Camera camera | camera for this GameScreen |
|-----------------|----------------------------|

| | |
|---------------------|--|
| - String ingURL | URL to background image of this GameScreen |
| - Image img | background image of this GameScreen |
| <u>+ int width</u> | width of this GameScreen |
| <u>+ int height</u> | height of this GameScreen |

1.4.2) Constructor

| | |
|---|--|
| + GameScreen(double width, double height) | initialize GameScreen given width and height |
|---|--|

1.4.3) Methods

| | |
|----------------------|--|
| + void addListener() | add input listener for this GameScreen |
| + void update() | update GameScreen corresponding to camera and player |

2) Package objects

2.1) *Abstract class GameObject*

2.1.1) Fields

| | |
|------------|--------------------|
| # float x | x coordinate |
| # float y | y coordinate |
| # float dx | velocity in x axis |
| # float dy | velocity in y axis |

| | |
|----------------|---------------------------|
| # float width | width of this GameObject |
| # float height | height of this GameObject |

2.1.2) Constructor

| | |
|---|---|
| + GameObject(float x, float y, float width, float height) | initialize GameObject given x, y coordinate, width and height |
|---|---|

2.1.3) Methods

| | |
|----------------------------------|--|
| + getter & setter for all fields | |
|----------------------------------|--|

2.2) interface Collidable

2.2.2) Methods

| | |
|------------------------|--------------------------------------|
| + Rectangle getBound() | return rectangle bound of the object |
|------------------------|--------------------------------------|

2.3) Class Block extends GameObject implements Renderable, Collidable

2.3.1) Fields

| | |
|---------------------|--------------------------------|
| - int z | coordinate in z axis |
| - String imgUrl | URL to the image of this Block |
| - Image image | image of this Block |
| + int <u>WIDTH</u> | width of the Block |
| + int <u>HEIGHT</u> | height of the Block |

2.3.2) Constructor

| | |
|---------------------------|--|
| + Block(float x, float y) | initialize Block given x, y coordinate |
|---------------------------|--|

2.3.3) Methods

| | |
|--------------------------------------|--|
| + Rectangle getBound() | return rectangle bound of this block |
| + int getZ() | return z coordinate |
| + void render(GraphicsContext gc) | render this block on given GraphicsContext |

2.4) Class Professor extends GameObject implements Renderable, Collidable

2.4.1) Fields

| | |
|-----------------------|---------------------------|
| - int z | coordinate in z axis |
| + <u>float WIDTH</u> | width of Professor |
| + <u>float HEIGHT</u> | height of Professor |
| - String imgURL | URL to image of Professor |
| - Image image | image of Professor |

2.4.2) Constructor

| | |
|-------------------------------|--|
| + Professor(float x, float y) | initialize Professor given x, y coordinate |
|-------------------------------|--|

2.4.3) Methods

| | |
|-----------------------------------|---|
| + int getZ() | return z coordinate |
| + void render(GraphicsContext gc) | render Professor on given GraphicsContext |
| + Rectangle getBound() | return rectangle bound of Professor |

2.5) Class Player extends GameObject implements Renderable, Collidable

2.5.1) Fields

| | |
|--------------------|--|
| - int z | coordinate in z axis |
| - boolean falling | true if player is falling, false otherwise |
| - boolean jumping | true if player is jumping, false otherwise |
| - int jumpVelocity | velocity of player right after jumped |
| - String imgURL | URL to image of player |
| - Image image | image of player |
| - String audioURL | URL to jumping sound of player |
| - AudioClip audio | jumping sound of player |

2.5.2) Constructor

| | |
|----------------------------|------------------------------|
| + Player(float x, float y) | initialize player given x, y |
|----------------------------|------------------------------|

| | |
|--|------------|
| | coordinate |
|--|------------|

2.5.3) Methods

| | |
|-----------------------------------|---|
| + void update() | update player position |
| + void checkCollision() | check if the player collides with another collidable object and take action depending on types of an object |
| + Rectangle getBound() | return bottom rectangle bound of player |
| + Rectangle getBoundTop() | return top rectangle bound of player |
| + Rectangle getBoundLeft() | return left rectangle bound of player |
| + Rectangle getBoundRight() | return right rectangle bound of player |
| + void render(GraphicsContext gc) | render player on given GraphicsContext |
| + getter & setter for all fields | |

2.6) Class Monk extends GameObject implements Renderable, Collidable

2.6.1) Fields

| | |
|--|--------------------------------|
| - int z | coordinate in z axis |
| - <u>ArrayList<String> monks</u> | list of URL to different monks |

| | |
|----------------|--------------------------|
| - Image image | image of individual monk |
| + float HEIGHT | height of monk |
| + float WIDTH | width of monk |

2.6.2) Constructor

| | |
|--------------------------|---------------------------------------|
| + Monk(float x, float y) | initialize monk given x, y coordinate |
|--------------------------|---------------------------------------|

2.6.3) Methods

| | |
|-----------------------------------|--------------------------------------|
| + void render(GraphicsContext gc) | render monk on given GraphicsContext |
| + Rectangle getBound() | return rectangle bound of the monk |
| + getter & setter for all fields | |

3) Package menuComponents

3.1) Abstract Class MyButton extends ImageView

3.1.1) Fields

| | |
|-------------------|---------------------------|
| # String imgURL | button image URL |
| # String audioURL | button on click audio URL |
| # AudioClip audio | button on click audio |

3.1.2) Constructor

| | |
|--------------|---|
| + MyButton() | construct MyButton using parent constructor |
|--------------|---|

3.1.3) Methods

| | |
|----------------------------------|--------------------------------------|
| + void <i>setOnClick()</i> | take action when a button is clicked |
| + getter & setter for all fields | |

3.2) Class MyPane extends StackPane

3.2.1) Fields

| | |
|---------------------|-------------------------------|
| # String imgURL | URL to background image |
| # Image image | background image |
| # ImageView imgView | ImageView to background image |
| # String audioURL | URL to pane audio |
| # AudioClip audio | pane audio |

3.2.2) Constructor

| | |
|------------|--|
| + MyPane() | construct my pane using parent constructor |
|------------|--|

3.2.3) Methods

| | |
|----------------------------------|--|
| + getter & setter for all fields | |
|----------------------------------|--|

3.3) Class HelpButton extends MyButton

3.3.1) Constructor

| | |
|----------------|------------------------------|
| + HelpButton() | initialize button image URL, |
|----------------|------------------------------|

| | |
|--|--|
| | set button image, and set how this button act on being clicked |
|--|--|

3.3.2) Methods

| | |
|---------------------|--|
| + void setOnClick() | set if this button being clicked will play clicked sound and show how the game works |
|---------------------|--|

3.4) Class PlayButton extends MyButton

3.4.1) Constructor

| | |
|----------------|--|
| + PlayButton() | initialize button image URL, set button image and set how this button act on being clicked |
|----------------|--|

3.4.2) Methods

| | |
|---------------------|---|
| + void setOnClick() | set if this button being clicked will play clicked sound and begin the game |
|---------------------|---|

3.5) Class ExitButton extends MyButton

3.5.1) Constructor

| | |
|----------------|--|
| + ExitButton() | initialize button image URL, set button image and set how this button act on being clicked |
|----------------|--|

3.5.2) Methods

| | |
|---------------------|---|
| + void setOnClick() | set if this button being clicked will exit the game |
|---------------------|---|

3.6) Class MainMenuButton extends MyButton

3.6.1) Constructor

| | |
|--------------------|--|
| + MainMenuButton() | initialize button image URL, set button image and set how this button act on being clicked |
|--------------------|--|

3.6.2) Methods

| | |
|---------------------|--|
| + void setOnClick() | set if this button being clicked will play clicked sound and go to main menu |
|---------------------|--|

3.7) Class TryAgainButton extends MyButton

3.7.1) Constructor

| | |
|--------------------|--|
| + TryAgainButton() | initialize button image URL, set button image and set how this button act on being clicked |
|--------------------|--|

3.7.2) Methods

| | |
|---------------------|---|
| + void setOnClick() | set if this button being clicked will play clicked sound and begin new game |
|---------------------|---|

3.8) Class PlayAgainButton extends MyButton

3.8.1) Constructor

| | |
|---------------------|--|
| + PlayAgainButton() | initialize button image URL, set button image and set how this button act on being clicked |
|---------------------|--|

3.8.2) Methods

| | |
|---------------------|---|
| + void setOnClick() | set if this button being clicked will play clicked sound and go to begin new game |
|---------------------|---|

3.9) Class NextButton extends MyButton

3.9.1) Constructor

| | |
|--------------|--|
| + NextButton | initialize button image URL, set button image and set how this button act on being clicked |
|--------------|--|

3.9.2) Methods

| | |
|---------------------|---|
| + void setOnClick() | set if this button being clicked will play clicked sound and go to the next help menu |
|---------------------|---|

3.10) Class MenuPane extends MyPane

3.10.1) Fields

| | |
|-------------------------|----------------------------|
| - PlayButton playButton | playButton of the mainMenu |
|-------------------------|----------------------------|

| | |
|-------------------------|----------------------------|
| - HelpButton helpButton | helpButton of the mainMenu |
| - ExitButton exitButton | exitButton of the mainMenu |

3.10.2) Constructor

| | |
|--------------|---|
| + MenuPane() | load main menu background and main menu background music. Initialize all the button and attach it to the pane |
|--------------|---|

3.10.3) Methods

| | |
|--|--|
| + getter & setter for all extra fields | |
|--|--|

3.11) Class GameOverPane extends MyPane

3.11.1) Constructor

| | |
|------------------|--|
| + GameOverPane() | load GameOverPane background, initialize all the buttons and attached it to the pane |
|------------------|--|

3.12) Class WinPane extends MyPane

3.12.1) Constructor

| | |
|-------------|--|
| + WinPane() | load WinPane background and initialize all the buttons and attached them to the pane |
|-------------|--|

4) Package logic

4.1) Class GameLogic

4.1.1) Fields

| | |
|------------------------------------|---|
| - <u>GameLogic INSTANCE</u> | instance of GameLogic |
| - MenuPane menuPane | MenuPane of the game |
| - GameOverPane gameOverPane | GameOverPane of the game |
| - WinPane winPane | WinPane of the game |
| - Stage stage | stage of the game |
| - Scene scene | scene of the game |
| - Player player | player of the game |
| - GameScreen gameScreen | GameScreen of the game |
| - Status status | status of the game |
| - AnimationTimer animationTimer | AnimationTimer for when the game is being played |
| - ArrayList<String> maps | list of URL to all available maps |
| + <u>double gravity</u> | gravity of the game |

4.1.2) Constructor

| | |
|---------------|----------------------|
| + GameLogic() | initialize GameLogic |
|---------------|----------------------|

4.1.3) Methods

| | |
|---|--|
| + void win() | show WinPane indicate that player has won the game |
| + void help() | show help menu |
| + void nextHelp() | show another page of help menu |
| + void gameOver() | show GameOverPane indicate that player has lost the game |
| + void play() | show gameScreen and random the map for player to play |
| + void menu() | show main menu |
| + <u>GameLogic getInstance()</u> | return instance of GameLogic |
| + ArrayList<Renderable> generateMap(Image img) | generate map from given image |
| + void update() | update the game while playing |
| + getter & setter for all fields | |

4.2) Enum Status

4.2.1) Enum

This enum represents game status, it contains following values: Menu; Playing; GameOver; Finished.

5) Package input

5.1) Class InputUtil

5.1.1) Fields

| | |
|--|--|
| <u>- ArrayList<KeyCode></u> <u>keyPressed</u> | store all the keys being pressed at the moment |
|--|--|

5.1.2) Methods

| | |
|---|---|
| <u>+ void clear()</u> | clear input |
| <u>+ boolean</u> <u>getKeyPressed(KeyCode key</u> <u>keycode)</u> | check if a key is pressed at the moment |
| <u>+ void</u> <u>setKeyPressed(KeyCode</u> <u>keycode, boolean pressed)</u> | add a key into keyPressed or remove a key from keyPressed |

6) Package Application

6.1) Class Main extends Application

6.1.1) Methods

| | |
|-----------------------------------|--|
| <u>+ void start(Stage stage)</u> | set up the stage, scene, main menu, icon, title, and set unresizable |
| <u>+ void main(String[] args)</u> | launch the application |

Folder res

1) Sound

1.1) bgm.mp3

The background music of the main menu.

1.2) clickSound.mp3

The sound when the player clicks a button.

1.3) jumpSound.wav

The sound when the IntaniaGhost jumps.

1.4) loseSound.mp3

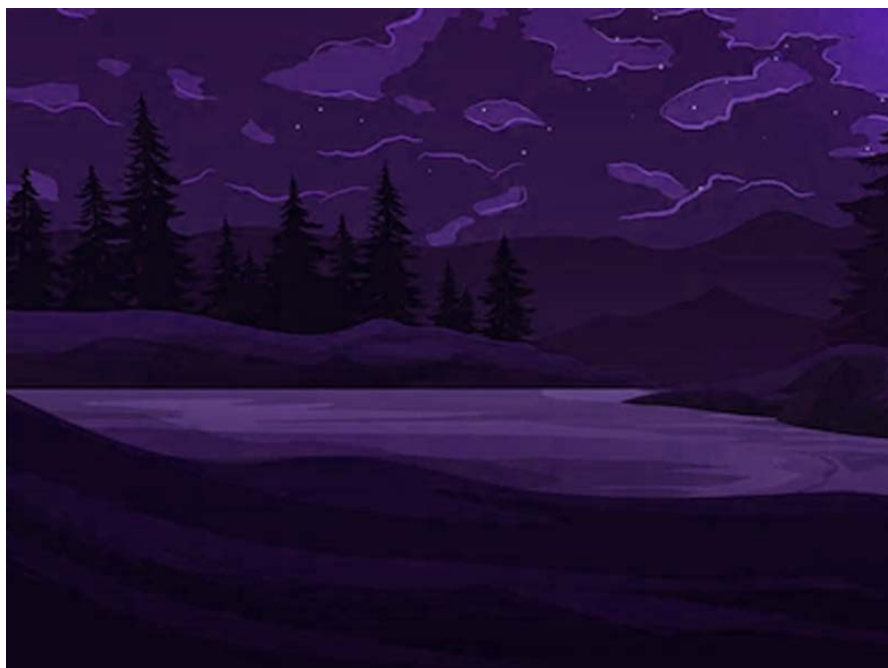
The sound when the player loses.

1.5) winSound.mp3

The sound when the player wins.

2) Image

2.1) background.png



The background of the game.

2.2) exitButton.png



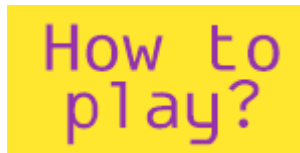
Click here to exit the game.

2.3) floor.png



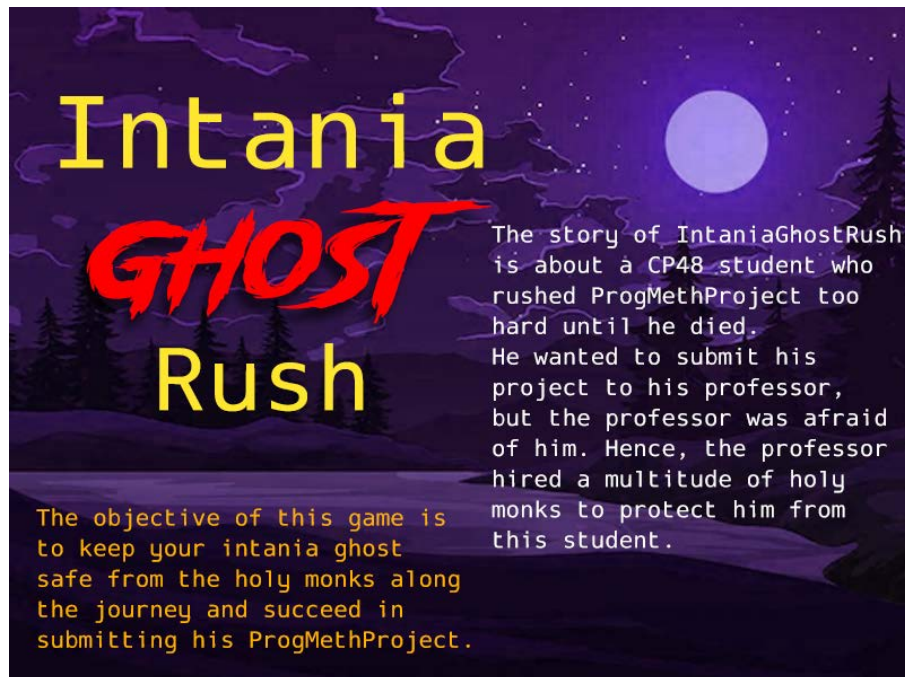
The floor block.

2.4) howToPlayButton.png



Click here to see instructions.

2.5) howToPlayMenu.png



The first page of instructions.

2.6) nextHowToPlayMenu.png



The second page of instructions.

2.7) icon.png



The icon of the game.

2.8) loseMenu.png



It shows when the player loses.

2.9) mainMenu.png



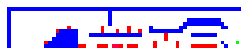
The main menu.

2.10) mainMenuButton.png



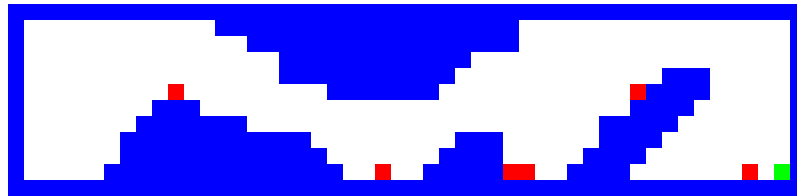
Click here to get back to the main menu.

2.11) map1.png



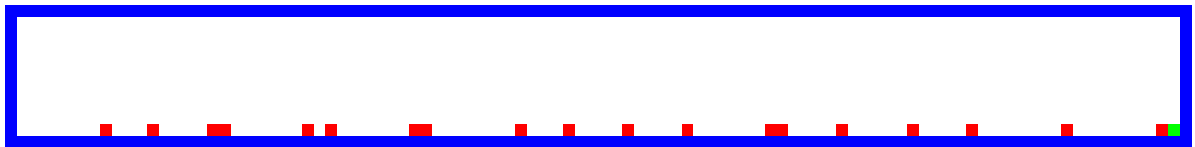
The first map.

2.12) map2.png



The second map.

2.13) map3.png



The third map.

2.14) monk1.png



The first enemy.

2.15) monk2.png



The second enemy.

2.16) monk3.png



The third enemy.

2.17) nextButton.png

Next

Click here to see the next instruction.

2.18) playAgainButton.png

Play
again

Click here to play again when the player wins.

2.19) player.png



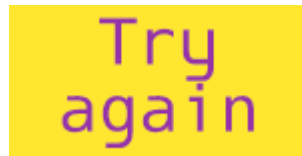
This is the IntaniaGhost.

2.20) professor.png



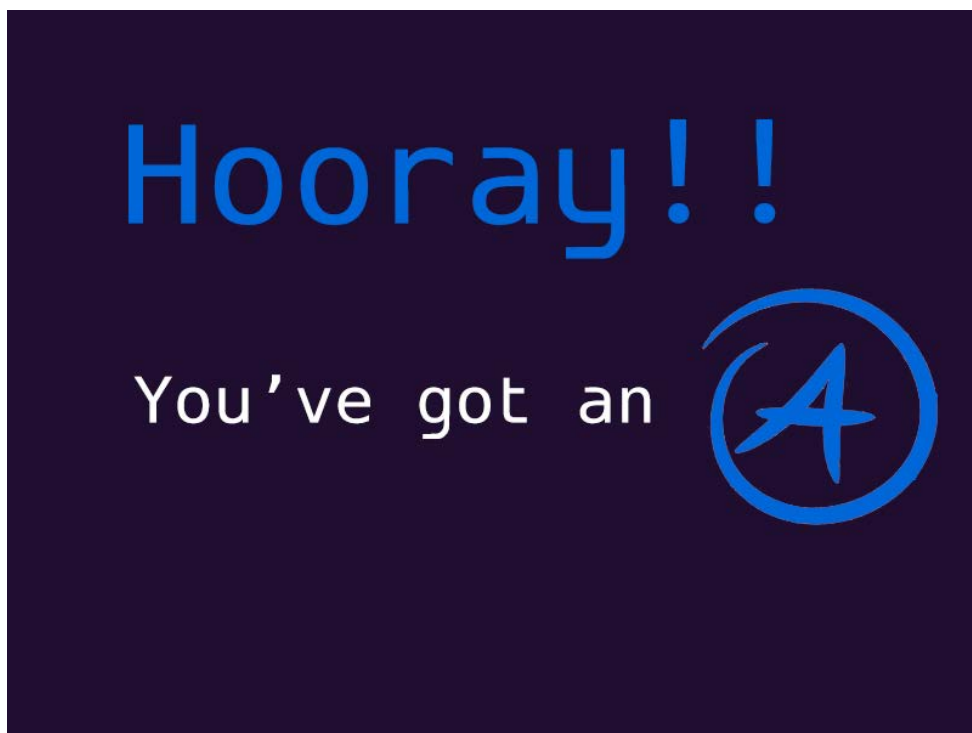
And his professor.

2.21) tryAgainButton.png



Click here to play again when the player loses.

2.22) winMenu.png



It shows when the player wins.