



KAUNAS UNIVERSITY OF TECHNOLOGY

FACULTY OF INFORMATICS

T120B166 Development of Computer Games and Interactive Applications

Banisher

Group, Name and Surname:
IFF 7-11, Mantas Klimašauskas
Date: 2020-03-02

Kaunas, 2020

Table of Contents

<i>Table of Images</i>	3
<i>Table of functions</i>	4
<i>Work Distribution Table:</i>	5
<i>Description of Your Game</i>	6
<i>Laboratory work #1</i>	7
List of tasks.....	7
<i>Solution</i>	7
1. Second level design.....	7
2. Basic movement player physics.....	7
3. Second level design.....	8
4. Active ragdolls, which try to stay upright (standing).....	9
<i>Laboratory 1 defense task</i>	10
3. Added four area lights (colors: red, green, blue, yellow).....	10
4. In Window->Rendering->Lighting settings enabled “Auto generate”, to enable automatically baking lightmaps	10
<i>Laboratory work #2</i>	10
List of tasks.....	10
<i>Solution</i>	11
Task #1. <i>Title of Task</i>	11
Task #2. <i>Title of Task</i>	11
Task #3. <i>Title of Task</i>	12
<i>Laboratory work #3</i>	14
List of tasks.....	14
<i>Solution</i>	14
Task #1. <i>Title of Task</i>	14
Task #2. <i>Title of Task</i>	14
Task #3. <i>Title of Task</i>	15
<i>User's manual</i>	17
<i>Literature list</i>	18
<i>ANNEX</i>	19

Table of Images

Figure 1. Second level layout	7
Figure 2. Player ground position before moving above the white box and after	8
Figure 3. Inside the warehouse	8
Figure 4. Looking into the warehouse through the alleyway	9
Figure 5. Active ragdoll, trying to stand upright	10
Figure 6. Lab 1 defense result	10
Figure 7. Screenshot #1	11
Figure 8. Screenshot #2	12
Figure 9. Screenshot #3	12
Figure 10. Screenshot #1	14
Figure 11. Screenshot #2	15
Figure 12. Screenshot #3	15
Figure 13. Screenshot #5	17
Figure 14. Screenshot #5	17

Table of functions

Table 1. Title of fragment #1	11
Table 2. Title of fragment #2.....	12
Table 3. Title of fragment #3.....	13
Table 4. Title of fragment #1	14
Table 5. Title of fragment #2.....	15
Table 6. Title of fragment #3.....	16

Work Distribution Table:

<i>Name/Surname</i>	<i>Description of game development part</i>
<i>Mantas Klimašauskas</i>	<i>Programming</i>
<i>Deividas Verbickas</i>	<i>Programming, level design</i>

Description of Your Game

Description of Your Game.

1. 3D or 2D? *3D*
2. What type is your game? *Action*.
3. What genre is your game? *Fighting*.
4. Platform? *PC*.
5. Scenario Description. *A fighting game where the player has the ability to enter a separate mirror dimension.*

The dimension is called the Exsilium and the one who can enter it are called outcasts (exsul). In the Exsilium you can see and touch (and even affect) the real world. Time moves slower in the Exsilium relative to the real world, which gives an advantage to anyone who is in it. In order to stay in the Exsilium the outcast must use their Vis (energy). The more matter the outcast takes to the mirror dimension the more Vis is being used. Also, the outcast uses more energy the further away they are from their physical body (in the real world). Vis is also directly related to stamina of the outcast in the real world.

Inside the Exsilium outcasts can use their energy to acquire buffs, such as super strength. Most of the enemies encountered by the player do not possess the power to enter use the Exsilium to their advantage. But there will be bosses, who will be just as capable (or even stronger) than the player. Anyone who is damaged while in the mirror dimension cannot be killed and simply loses their energy. In order to regain the energy, the outcast must use special items.

Laboratory work #1

List of tasks

1. Second level basic layout
2. Basic movement player physics
3. Add map details and lights
4. Active ragdolls, which try to stay upright (standing)
- 5.

Solution

1. Second level design

Created a small alleyway, that leads to a warehouse. The alleyway is surrounded by a few buildings. The warehouse has three holes in the walls as entrances.

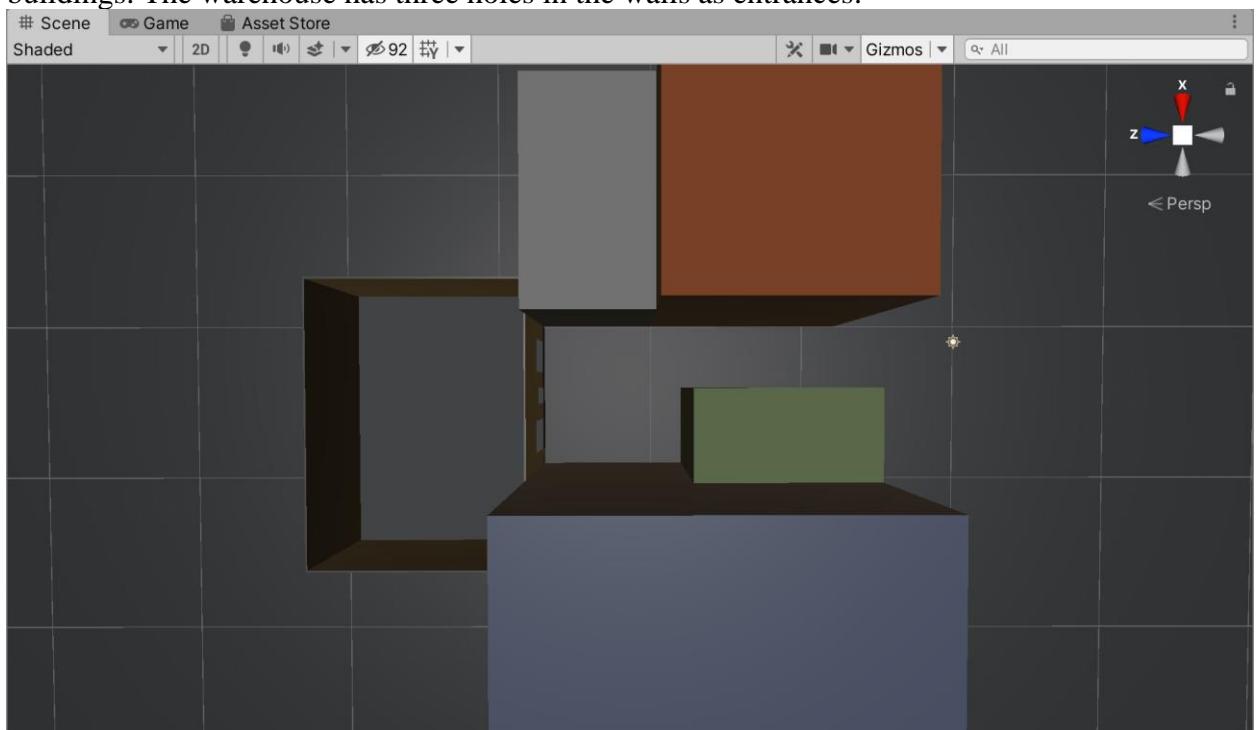


Figure 1. Second level layout

2. Basic movement player physics

Added a script for the SteamVR Player prefab, to enable basic physical ground detection for the player.

When the player moves over a collider, the script will push (teleport) the player up, to adjust the real-world ground to the object the player character in the game is standing on.

When the player steps off an object, a rigidbody attached to the Player prefab is enabled (is kinematic is turned off) and the player falls, until the ground is close enough.

To detect the ground below the player, a raycast with a limited distance is casted down. If the raycast hits something, that something is considered the ground, and the player's position is adjusted accordingly. If the raycast doesn't hit anything, the attached rigidbody is activated for the player to fall.

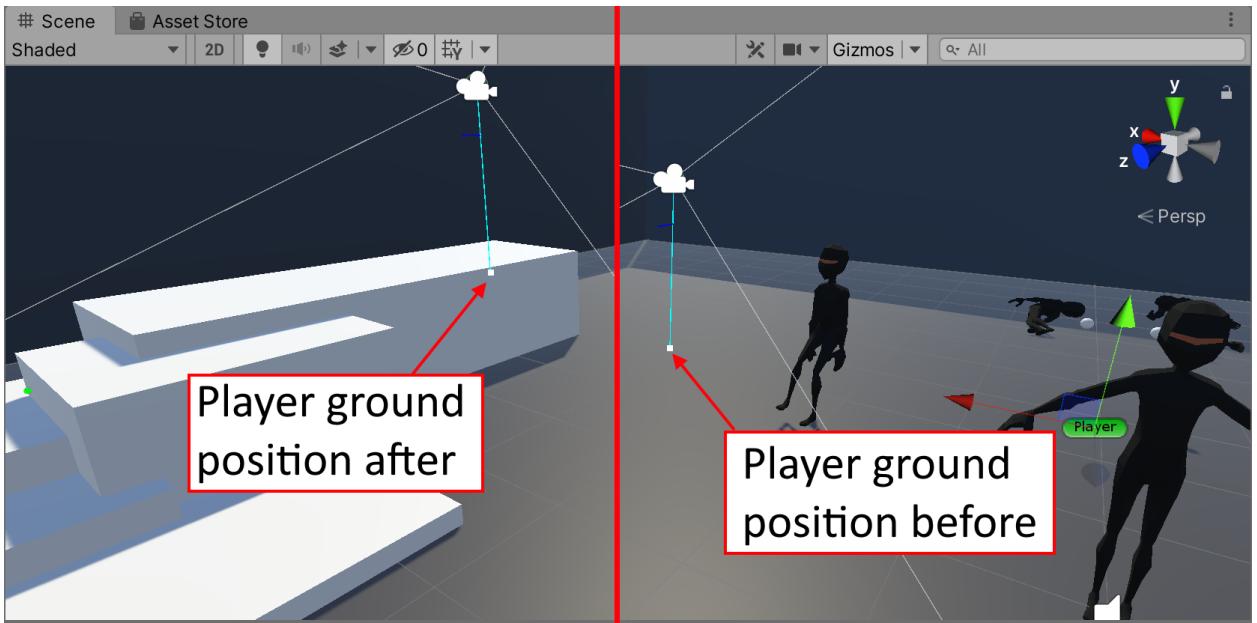


Figure 2. Player ground position before moving above the white box and after

3. Second level design

Added doors and ramps to some of the buildings. Also, added an indoor second-floor office, with windows. The windows are made with a transparent material. Also added a few pillars inside the warehouse.

For lighting, added five spot lights inside the warehouse, change the direction of the directional light (the sun) to make it an evening setting.

Also made a few prefabs, such as doors, ramps, lights.

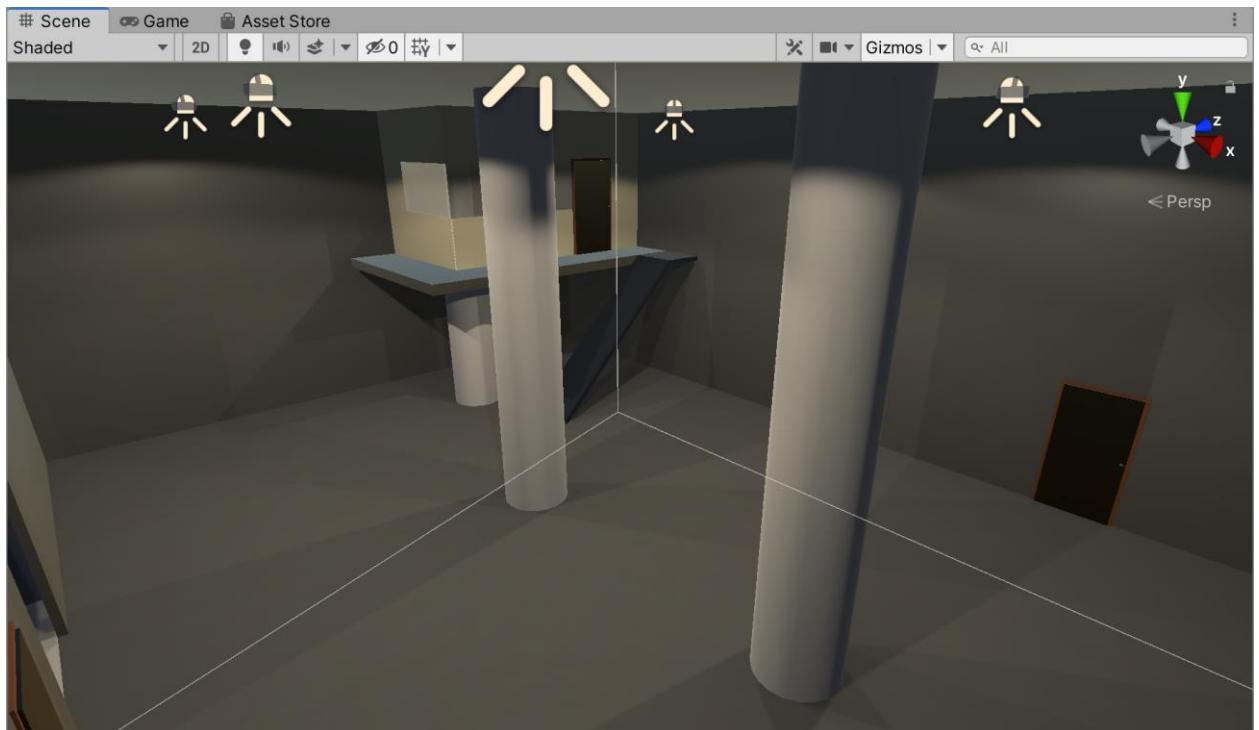


Figure 3. Inside the warehouse



Figure 4. Looking into the warehouse through the alleyway

4. Active ragdolls, which try to stay upright (standing)

Started the work on active ragdolls (ragdolls, which try to physically move to certain positions), for using them on enemies and the player body.

To create the ragdoll itself, a free asset* from the Unity Store for a rigged humanoid character model. Then, using the Unity's built-in ragdoll creation tools (GameObject->3D Object->Ragdoll...) setup the character joints and rigidbodies of the character.

The actual active part of the ragdoll is composed of multiple custom scripts.

The first script is used to push and rotate rigidbodies to a target position and rotation.

The second script is used to calculate the wanted position and rotation of a few parts of the ragdoll. Then using the first script, the ragdolls certain parts are pushed and rotated towards the calculated target spots. To keep the ragdoll standing, a raycast is casted, to calculate where the ground is, relative to the ragdoll.

* Lopoly Ninja by Tetra Arts

(<https://assetstore.unity.com/packages/3d/characters/humanoids/lowpoly-ninja-157942>)

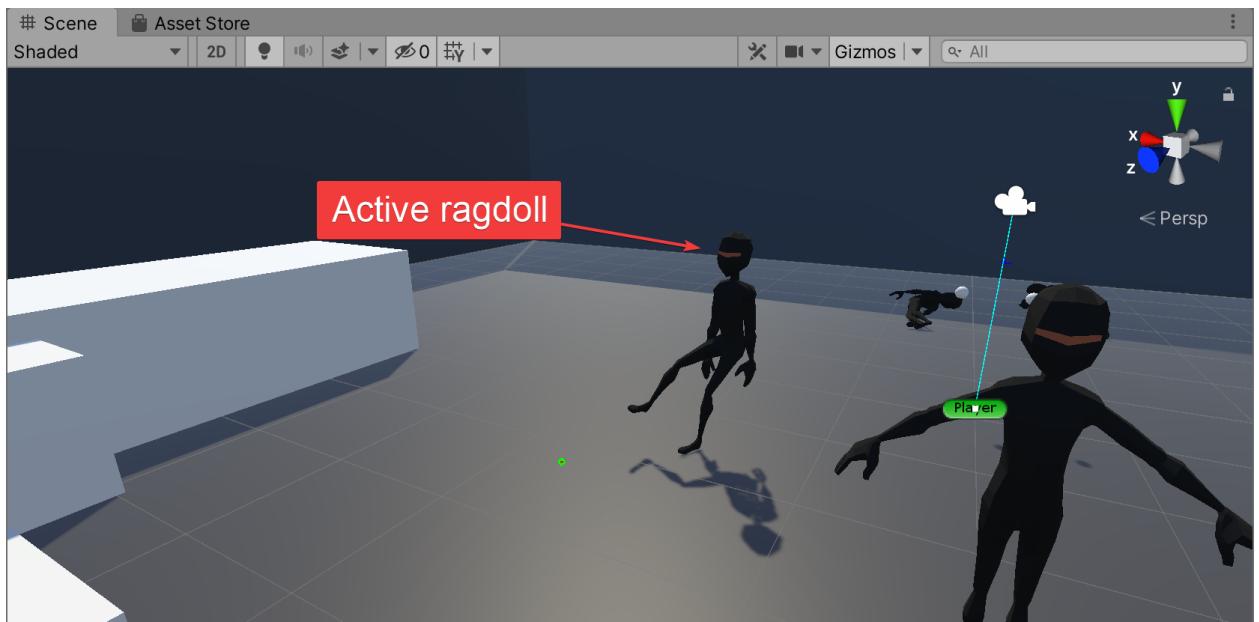


Figure 5. Active ragdoll, trying to stand upright

Laboratory 1 defense task

Create a Microsoft logo using a plane for logo surface and area lights for colors.

Steps:

1. Created a 3D plane
2. Marked the plane as static, to enable baked lighting for it
3. Added four area lights (colors: red, green, blue, yellow)
4. In Window->Rendering->Lighting settings enabled “Auto generate”, to enable automatically baking lightmaps

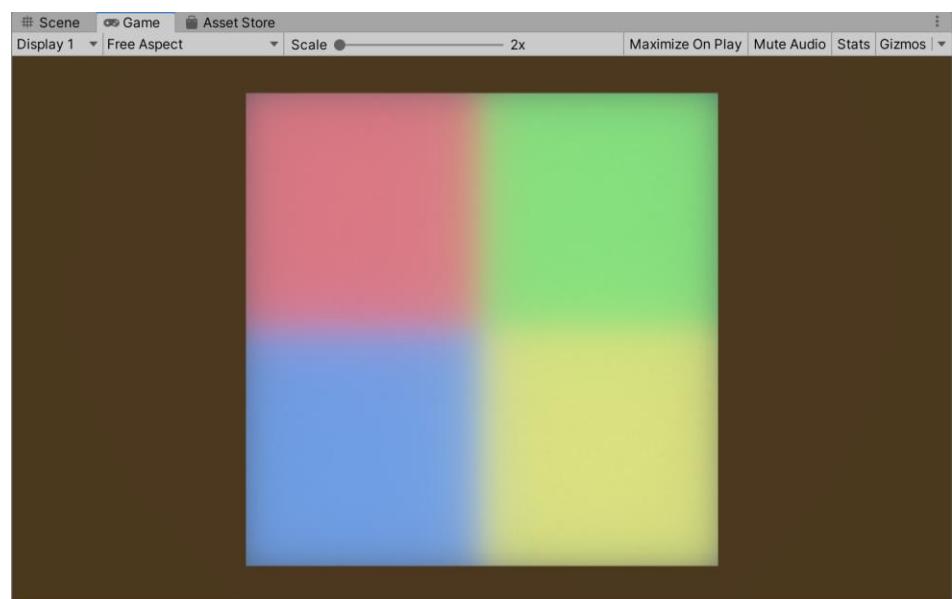


Figure 6. Lab 1 defense result

Laboratory work #2

List of tasks (main functionality of your project)

1. Title of Task #1
2. Title of Task #2

3. Title of Task #3
4. ...

Solution

Task #1. *Title of Task*

Description of implementation (3-5 sentences). *Vestibulum hendrerit felis at turpis ultrices imperdiet. Nulla facilisi curabitur vitae semper nulla. Etiam rhoncus orci dolor, ac dictum erat iaculis sed. Aliquam pulvinar viverra consequat. Nam eu mi in mauris semper pellentesque eget ut erat.*

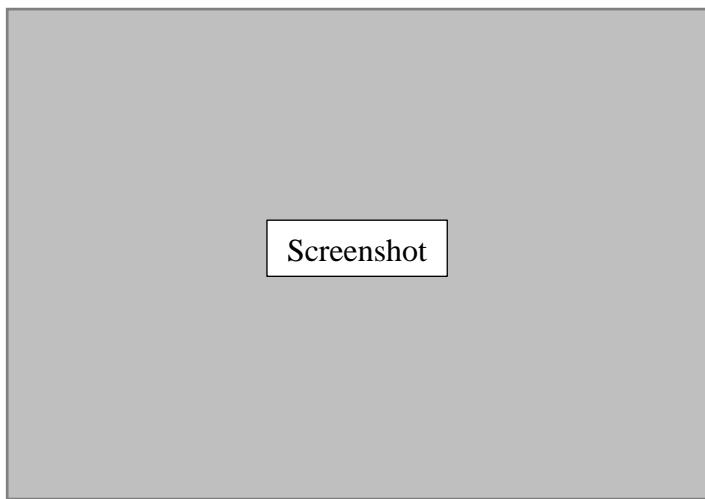


Figure 7. Screenshot #1

In the case of using functions, the description of each main function should be completed with the source code FRAGMENTS (the functions should be indexed in a separate table of contents);

A light blue rectangular placeholder box with a thin black border. In the center, there is a smaller white rectangular box with a thin black border containing the text "Fragment of Source Code".

Table 1. Title of fragment #1

Task #2. *Title of Task*

Description of implementation (3-5 sentences). *Nullam id dolor id nibh ultricies facilisis. Sed ultricies nunc sit amet sem blandit, at ultricies nibh commodo. Duis ut mollis risus. Proin hendrerit libero eu felis dapibus imperdiet. Fusce posuere felis ornare luctus molestie. Duis ut odio pretium, bibendum elit et, molestie quam.*

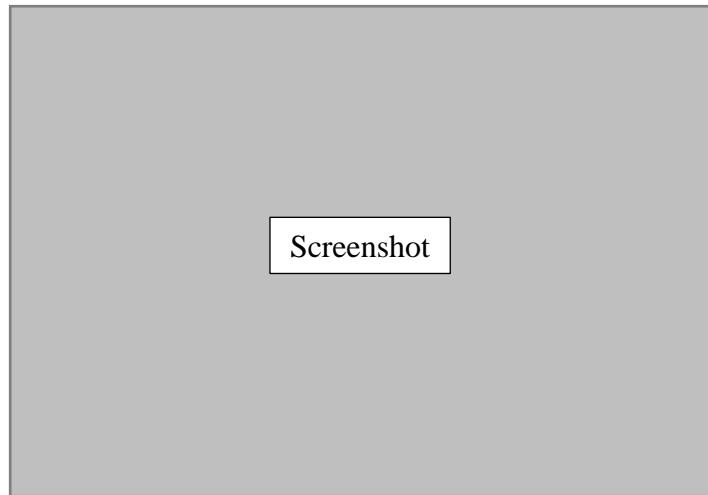


Figure 8. Screenshot #2

In the case of using functions, the description of each main function should be completed with the source code FRAGMENTS (the functions should be indexed in a separate table of contents);

Fragment of Source Code

Table 2. Title of fragment #2

Task #3. *Title of Task*

Description of implementation (3-5 sentences). *Vestibulum hendrerit felis at turpis ultrices imperdiet. Nulla facilisi curabitur vitae semper nulla. Etiam rhoncus orci dolor, ac dictum erat iaculis sed. Aliquam pulvinar viverra consequat. Nam eu mi in mauris semper pellentesque eget ut erat.*

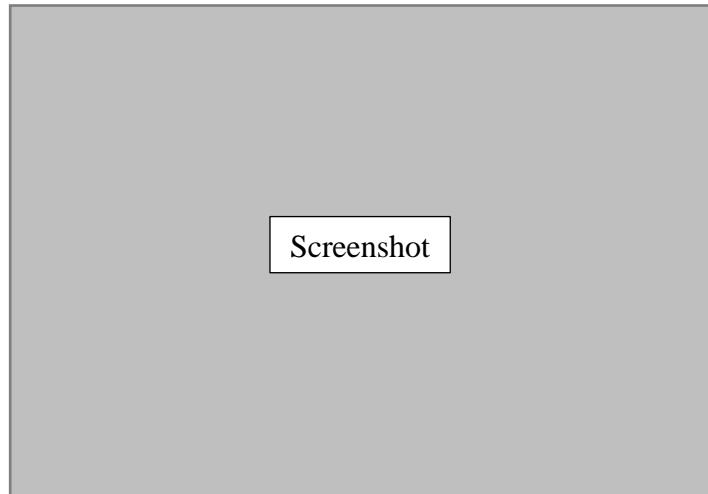


Figure 9. Screenshot #3

In the case of using functions, the description of each main function should be completed with the source code FRAGMENTS (the functions should be indexed in a separate table of contents);

Fragment of Source Code

Table 3. Title of fragment #3

Laboratory work #3

List of tasks (main functionality of your project)

1. Title of Task #1
2. Title of Task #2
3. Title of Task #3
4. ...

Solution

Task #1. *Title of Task*

Description of implementation (3-5 sentences). *Vestibulum hendrerit felis at turpis ultrices imperdiet. Nulla facilisi curabitur vitae semper nulla. Etiam rhoncus orci dolor, ac dictum erat iaculis sed. Aliquam pulvinar viverra consequat. Nam eu mi in mauris semper pellentesque eget ut erat.*

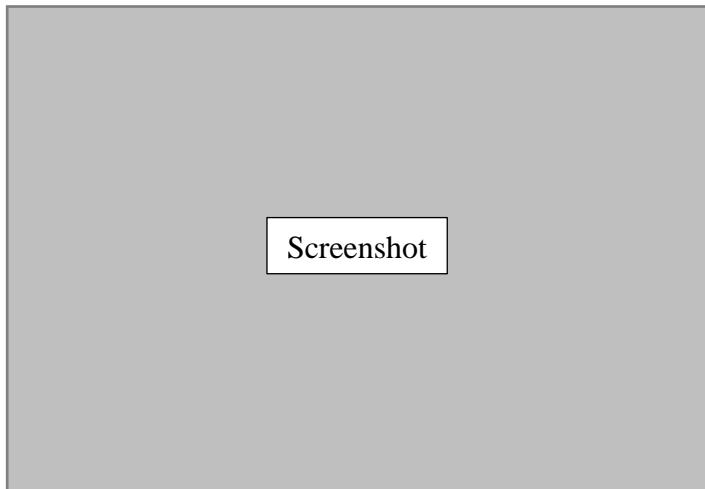


Figure 10. Screenshot #1

In the case of using functions, the description of each main function should be completed with the source code FRAGMENTS (the functions should be indexed in a separate table of contents);

Fragment of Source Code

Table 4. Title of fragment #1

Task #2. *Title of Task*

Description of implementation (3-5 sentences). *Lorem ipsum dolor sit amet, consectetur adipiscing elit. Sed ultricies nunc sit amet sem blandit, at ultricies nibh commodo. Duis ut mollis risus. Proin hendrerit libero eu felis dapibus imperdiet. Fusce posuere felis ornare luctus molestie. Duis ut odio pretium, bibendum elit et, molestie quam.*

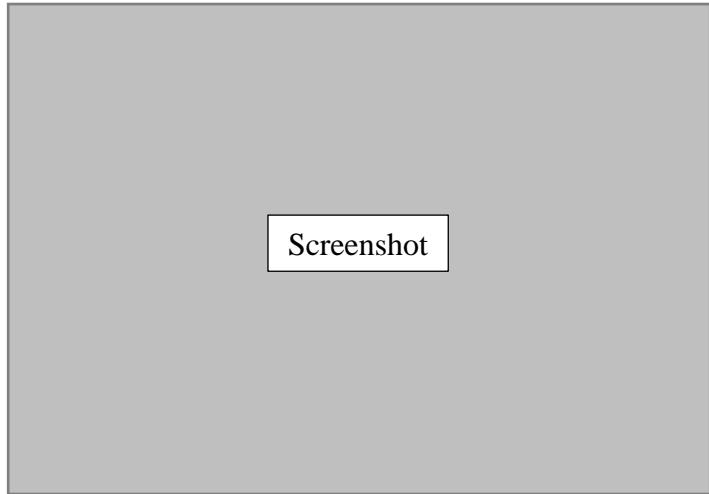


Figure 11. Screenshot #2

In the case of using functions, the description of each main function should be completed with the source code FRAGMENTS (the functions should be indexed in a separate table of contents);

Fragment of Source Code

Table 5. Title of fragment #2

Task #3. *Title of Task*

Description of implementation (3-5 sentences). *Vestibulum hendrerit felis at turpis ultrices imperdiet. Nulla facilisi curabitur vitae semper nulla. Etiam rhoncus orci dolor, ac dictum erat iaculis sed. Aliquam pulvinar viverra consequat. Nam eu mi in mauris semper pellentesque eget ut erat.*

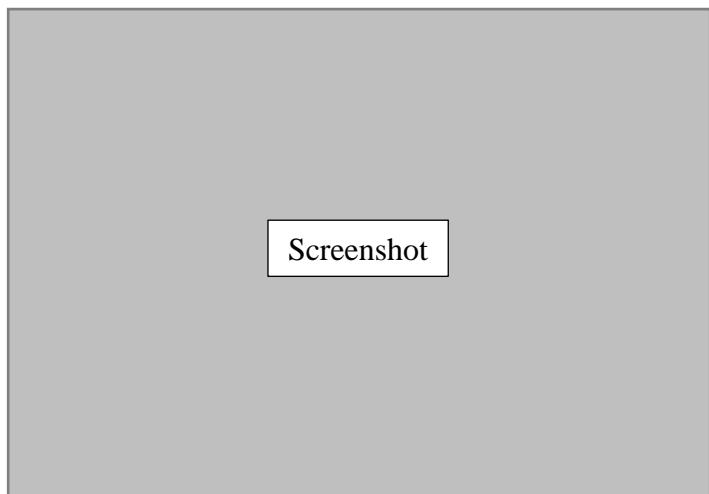


Figure 12. Screenshot #3

In the case of using functions, the description of each main function should be completed with the source code FRAGMENTS (the functions should be indexed in a separate table of contents);

Fragment of Source Code

Table 6. Title of fragment #3

User's manual (for the Individual work defence)

How to play? *Aenean eu quam gravida, laoreet nisl eu, sagittis quam. Donec sit amet nunc nisi. Sed vel ipsum metus. Nullam accumsan vestibulum ex. Aenean eu quam gravida, laoreet nisl eu, sagittis quam. Donec sit amet nunc nisi. Sed vel ipsum metus. Nullam accumsan vestibulum ex.*

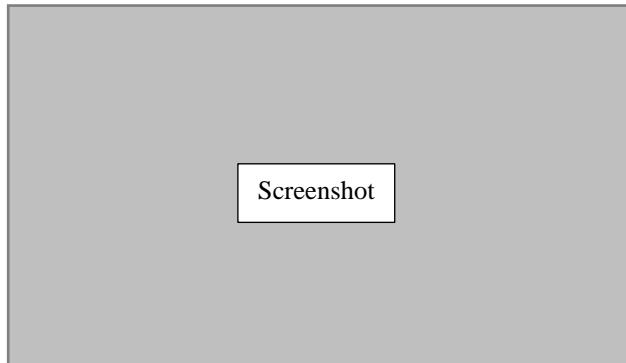


Figure 13. Screenshot #5

Nunc vel enim vel magna interdum dapibus id nec nisl. Suspendisse elit augue, accumsan tempor erat sed, gravida suscipit urna. Duis blandit lacin et finibus finibus. Mauris pretium pharetra orci dictum luctus. Nullam commodo magna a tincidunt malesuada.

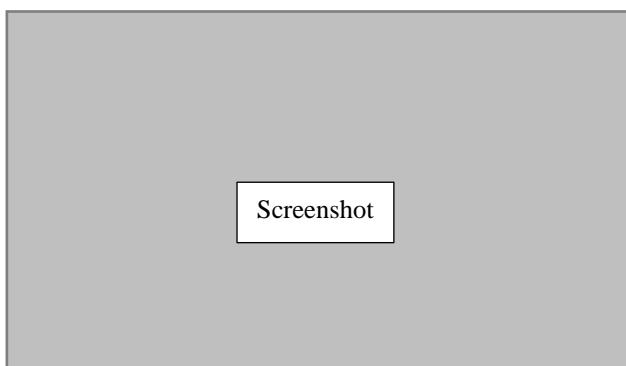


Figure 14. Screenshot #5

Sed sollicitudin justo erat, viverra luctus mi consequat non. Sed ut condimentum libero. Duis rutrum lacin ante, vitae feugiat ex faucibus at. Maecenas pulvinar et augue sed commodo.

Descriptions of the rules of the game. Nunc quis condimentum lacin. Quisque felis neque, ullamcorper vel posuere eget, blandit non neque. Nam in varius erat. Duis molestie sit amet eros vel rhoncus. Nunc quis condimentum lacin. Quisque felis neque, ullamcorper vel posuere eget, blandit non neque. Nam in varius erat. Duis molestie sit amet eros vel rhoncus.

Descriptions of the controls / keys. Donec et lorem vitae ligula bibendum faucibus. Suspendisse interdum quis augue sed luctus. Curabitur ac diam augue. In hac habitasse platea dictumst. Curabitur maximus maximus tortor. Nunc quis condimentum lacin. Quisque felis neque, ullamcorper vel posuere eget, blandit non neque. Nam in varius erat. Duis molestie sit amet eros vel rhoncus.

Literature list

1. Source #1. *Url*
2. Source #2. *Url*
3. ...
4. Source #N. *Url*

ANNEX

All source code is contained in this part.