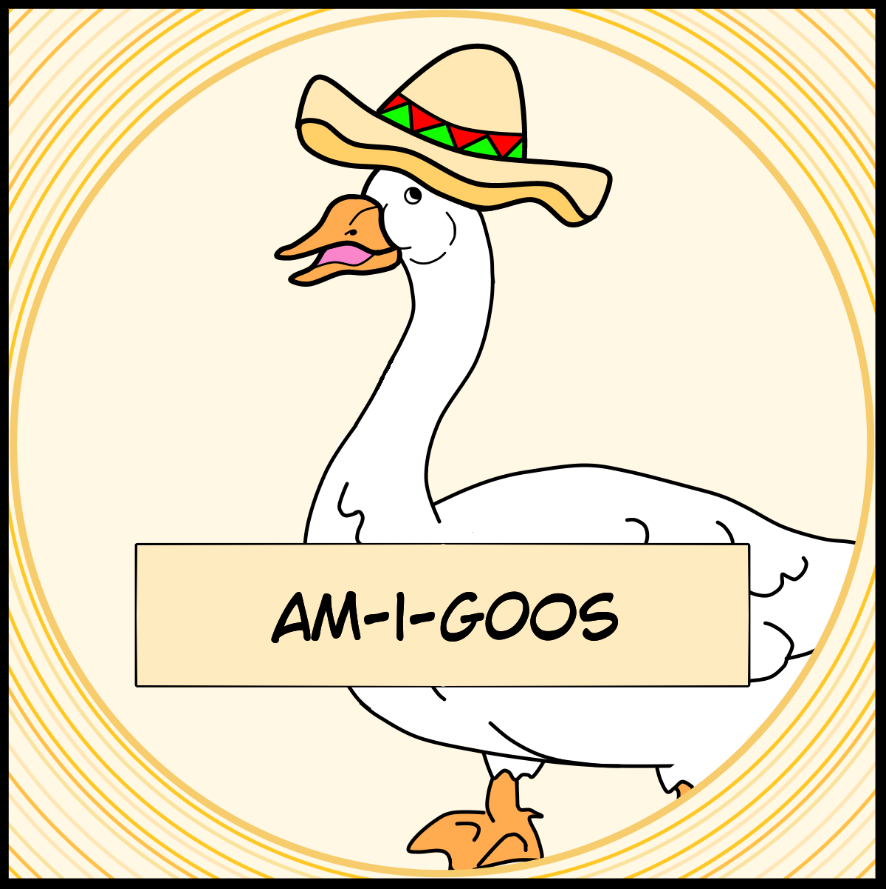
Logic Game 2022



TEam am-i-goos

Content:

1. [Team members 2](#_Team_members)
2. [Topic 2](#_Topic)
3. [Technologies used 2](#_Technologies_used)
4. [Project stages 2](#_Project_stages)
5. [Summary 3](#_Summary)

# Team members

* + [Aleksander Kostov](mailto:AIKostov20@codingburgas.bg)
    - Class: 9B
    - Role: Scrum Trainer
  + [Nikolay Despotov](mailto:NVDespotov20@codingburgas.bg)
    - Class: 9V
    - Role: Frontend Developer
  + [Yoan Todorov](mailto:YYTodorov20@codingburgas.bg)
    - Class: 9V
    - Role: Backend Developer
  + [Aleksandra Petrova](mailto:APPetrova20@codingburas.bg)
    - Class: 9V
    - Role: QA Developer

# Topic

* + Our task was to make the game b00le0 as a C++ program to not lose the idea of such an interesting and fun board game. The task had to be completed by 15.05.2022. From 20.04.2022 until then we had to form a team, distribute our roles and everyone’s tasks and make a finished product, which you can read more about in the next section.

# Technologies used

* + We used Visual Studio 2019 to write the C++ code for the game. We made different structures for cards and slots. With MediBang Paint, we created our game’s cards, rules, buttons, and other textures. They were added with one special library called SFML.

# Project stages

* + In our main menu, you can see 4 buttons. Two of these buttons are leading to 2 different stages. The first one is a person against another person. The next one is a person vs computer. The next button is How to play, which shows the rules of the game. And the final button Exit is to get out of the program.

|  |  |
| --- | --- |
| **Function table** | |
|
| void PVP() | void H2P() |
| The main game mode. In it, two people play against each other. It consists of several parts using SFML functions and classes, such as sf::RectangleShape, sf::Texture, and their functions like: sf::Texture::loadFromImage(); sf::RectangleShape::setTexture(); to place textures on the cards that are instances of the custom Cards structure and on slots that are instances of the Slot. It also uses vectors to store these instances. | and sf::Texture. The function shows the user how to play the game. It consists of many SFML classes and functions like sf::RectangleShape and sf::Texture() |
|
|
|
|
|
|
|
|
|
|

|  |  |
| --- | --- |
| **Function table** | |
|
| void PVC() | void mainMenu() |
| The secondary game mode. In it, a person plays against his computer. It consists of several parts using SFML functions and classes, such as sf::RectangleShape, sf::Texture, and their functions like: sf::Texture::loadFromImage(); sf::RectangleShape::setTexture(); to place textures on the cards that are instances of the custom Cards structure and on slots that are instances of the Slot. It also uses vectors to store these instances. | The basis of the code, this function is responsible for calling all other functions like PVP(); PVC(); and H2P(); mainMenu() uses sf::RectangleShape; sf::Texture(); |
|
|
|
|
|
|
|
|
|
|

# Summary

* + This project was another great opportunity for us to develop our skills in working with people, communication, and writing C++ programs.
  + As a future update, we can add 2 more stages with special cards called Not cards. In that way, we will finish the game b00le0 completely.

**[Click here to open Block Diagram](https://codingburgas-my.sharepoint.com/:i:/g/personal/nvdespotov20_codingburgas_bg/ESkUoVxmwnFFqahsJqyLlf0BQ3rpPv0KgT8TraTbnCaUNg?e=zvAG2u)**

Stages of realization