Documentation

Project – Hangman

Team: The Sugar Gliders

**Съдържание на документацията**

[1. Членове на отбора 3](#_Toc65260610)

[2. Образователна институция 3](#_Toc65260611)

[3. Проект 4](#_Toc65260612)

[3.1. Идея 4](#_Toc65260613)

[3.2. Работа 4](#_Toc65260614)

[3.3. Затруднения 4](#_Toc65260615)

[3.4. Програма 4](#_Toc65260616)

[4. История на играта 5](#_Toc65260617)

[5. Елементи на играта 5](#_Toc65260618)

[6. Правила на играта 5](#_Toc65260619)

[7. Избрани програмни езици 5](#_Toc65260620)

[8. Описание на C ++ функциите 6](#_Toc65260621)

[9. Блок схема 7](#_Toc65260622)

# 1. Членове на отбора

* Georgi Vasilev 9A - Scrum Trainer - [GAVasilev19@codingburgas.bg](mailto:GAVasilev19@codingburgas.bg)
* Georgi Ivanov 9B - QA Engineer - [GVIvanov19@codingburgas.bg](mailto:GVIvanov19@codingburgas.bg)
* Ivailo Radev 9G - Code Checker - [IMRadev19@codingburgas.bg](mailto:IMRadev19@codingburgas.bg)
* Tereza Opanska 9V - C++ Developer - [TNOpanska19@codingburgas.bg](mailto:TNOpanska19@codingburgas.bg)

# 2. Образователна институция

- Професионална гимназия по компютърно програмиране и иновации;



# 3. Проект

## **3.1. Идея**

Идеята ни за проекта беше да направим конзолна апликация на играта бесеница, на която ти можеш да въвеждаш дума, или да ползваш случайна.

## **3.2. Работа**

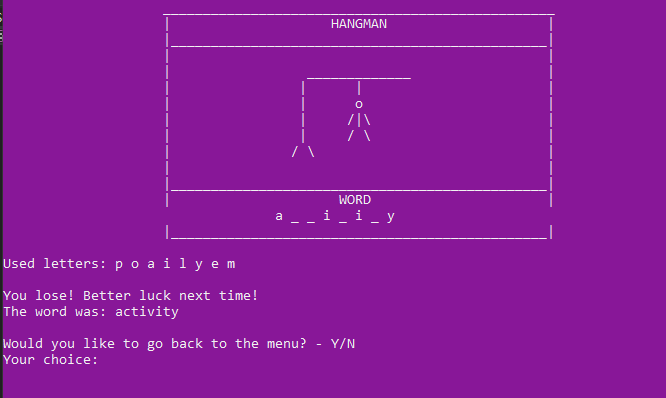
1. Първото нещо, което направихме, беше да съберем отбора ни и да определим ролите. След това си избрахме място, в което да си комуникираме. Следващата ни задача беше да си измислим идея за проекта и да разпределим задачите.
2. След като задачите бяха разпределени, всеки започна да работи по своята част. Стараехме се всяка седмица да си правим поне по два meeting-a, за да може да следим напредъка.
3. Последната част, беше да си финализираме GitHub repository-то и да се подготвим за представянето.

## **3.3. Затруднения**

Най-голяма трудност за нас беше QA документацията, понеже никой не знаеше как да подходи с нея. Накрая успяхме да направим някаква документация, макар да знаем, че можеше да се справим и по-добре.

Други трудности бяха разпределнието на задачите и намирането на свободно време, в което отбора ни да работи по проекта. Ние сме горди, че преодоляхме тези предизвикателства и сме доволни от нашата работа.

## **3.4. Програма**



# 4. История на играта

В Европа от 17 и 18 век престъпниците, които са осъдени на смърт чрез обесване, могат да изискват „Обред на думите и живота“.

В „Обред на думите и живота“ престъпник, който трябва да бъде обесен, е нанизан върху стойка с 5 крака и му се показва дъска с поредица къси въжета, представящи дума. Палачът избира думата и използва въглен, за да маркира правилно отгатнати букви на чертичките на дъската и неправилни букви отстрани. При всяко погрешно предположение, палачът би използвал чук ​​или брадва, за да отблъсне единия крак на стойката. 5 грешни предположения и престъпникът е обесен. Ако цялата дума е попълнена правилно или предположена, осъденият бива освободен.

# 5. Елементи на играта

(Дъска, лист и тн.) с нарисувано човече, което е завързано за въже. Под човечето трябва да има толкова чертички, колкото букви има съответно търсената дума.

# 6. Правила на играта

Играчите се редуват в познаването на скритата дума, като при всяко отгатване на буква тя се премахва. Който познае скритата дума спечелва играта.

# 7. Избрани програмни езици



# 8. Описание на C ++ функциите

|  |  |  |
| --- | --- | --- |
| Име на функцията | Тип на функцията | Работа на функцията |
| printTitle() | void | Извежда заглавието на играта. |
| printMenu() | void | Извежда менюто на играта. |
| inputChoice() | int | Позволява на потребителя да въведе номера на желаната от него операция и връща числото на функцията choice(). |
| choice(int number) | void | Според въведения номер на операция, тази функция препраща потребителя към определен раздел от нашата програма. |
| printRules() | void | Извежда правилата на играта. |
| printTeamInfo() | void | Извежда информацията за отбора ни |
| returnToMenu() | void | Пита потребителя дали иска да се върне в менюто, или да излезе от играта. |
| gameMode() | void | Показва двата различни game mode-a на играча и го моли да избере единия от двата. |
| chooseGameMode() | int | Позволява на потребителя да въведе номера на желаният от него game mode и връща числото на функцията gameMode(). |
| getWord(int number) | string | Според избрания game mode, потребитиля или въвежда своя дума, или използва random дума генерирана от играта. |
| hiddenWord(string word) | string | „Маскира“ думата, която играча трябва да познае. |
| game(string word, string hidden\_word) | void | Повиква board() функцията и моли потребителя да въведе предположение (буква). |
| board(int wrong\_guess, string hidden\_word, string used\_letters) | void | Извежда полето на играта. |
| win() | void | Извежда съобщение на потребителя, че е познал думата |
| loss(string word) | void | Извежда съобщение на потребителя, че не е успял да позняе думата. |

# 9. Блок схема

