

The Time Machine

Sprint-Linked List

Icon

Description automatically generated

February 16, 2022

The Time Machine

VSCPI

Table of contents

[1. Project participants 2](#_Toc95832375)

[2. Project description 2](#_Toc95832376)

[3. Summary of the project 2](#_Toc95832377)

[3.1 Goal 2](#_Toc95832378)

[3.2 Stages of realization 2](#_Toc95832379)

[3.3 Difficulties 3](#_Toc95832380)

[3.4 Programs used 3](#_Toc95832381)

[4. Functions 3](#_Toc95832382)

[5. Block scheme 4](#_Toc95832383)

[6. Future ideas 4](#_Toc95832384)

# 1. Project participants

* Dobrin Peychev 10A – Scrum Trainer - [DIPeychev19@codingburgas.bg](mailto:DIPeychev19@codingburgas.bg)
* Nikolay Brankov 10B – QA Engineer - [NPBrankov19@codingburgas.bg](mailto:NPBrankov19@codingburgas.bg)
* Tereza Opanska 10V – Frontend Developer - [TNOpanska@codingburgas.bg](mailto:TNOpanska@codingburgas.bg)
* Kalin Chervenkov 10G – Backend Developer - [KSChervenkov19@codingburgas.bg](mailto:KSChervenkov19@codingburgas.bg)

# 2. Project description

Our task was to create a program using the programming language C++. The theme was history. The requirements for the project were that we use linked lists and implement gray code into the program. We managed to do both tasks. It was suggested to create a program where you can search, add and delete information that has to do with the theme history.

# 3. Summary of the project

## 3.1 Goal

Our goal for the project was to create an application where you can write information for Bulgaria’s most significant battles. We wanted our program to have a cisco-style command line interface.

## 3.2 Stages of realization

1. The first stage we had to go through was making our team according to the given team making criteria. After a brief consideration we formed our team in its positions. That took us roughly around two to three days after we held our first meeting. Around this time, we also chose the name for our team “The Time Machine”. After finishing the work around the creation of our team we started to think of an idea for the project. We went through several suggestions but in the end stuck to making a cisco-style command line interface program about the history of Bulgaria and in particular its battles.
2. The second stage we went through was working on the tasks we were given. This was probably the most difficult stage and the most time-consuming one. We encountered various problems during the realization of our idea. Fortunately, in the end we had a finished product and we had resolved most of our issues.
3. The third stage of realization was creating the PowerPoint presentation and documentation of our project. During this time, we had to finalize our product and add our finishing touches. This was also the time when we edited our README file.
4. The last and most important stage was our preparation for presenting our work. This was the most stressful period of the entire process because it was also the most essential one.

## 3.3 Difficulties

We experienced quite the number of difficulties such as the implementation of the Gray code in our program. Another difficulty we had was creating the maps for the battles and thinking of a way to use them in our project. We also experienced other minor difficulties here and there.

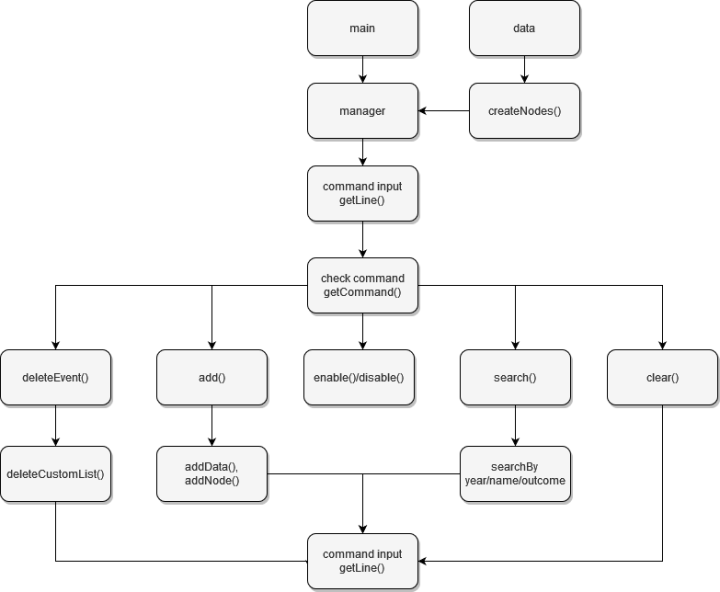
## 3.4 Programs used

The programs we used were Visual Studio for writing the code, Visual Studio Code for sorting the code and creating the README.md, GitHub for committing the project and sharing it, Photoshop for creating the logo, Word and PowerPoint for making the documentation and presentation, ArcGis for creating the maps and Microsoft Teams for communication.

# 4. Functions

|  |  |  |
| --- | --- | --- |
| Function name | Function type | Function Description |
| createNodes() | void | Creates the linked list for the nodes and makes the premade battles. |
| deleteNodes() | void | Deletes a particular node. |
| displayNames() | void | Displays all of the names from a linked list. |
| findEventByYear(int year) | void | Searches for an event that happened in a particular year. |
| findEventByName(std::string name) | void | Searches for an event with a particular name. |
| findEventByOutcome (std::string outcome) | void | Searches for an event based on whether it was won or lost by Bulgaria. |
| addData (std::vector <std::string> eventParams) | void | Adds additional data created by the user into an external file. |
| getData() | void | Access the data from the external file. |
| addNode (std::string name, int year, std::string outcome, std::string description, std::string reason) | void | Adds the data from the external file into a node. |

# 5. Block scheme



# 6. Future ideas

Although we have a finshed product, we did not manage to create all of the functionalities that we wanted to with the time given.

Some ideas that we have for the future include implementing an autocomplete functionallity for the commands, adding different options for event editing and adding a custom cli configuration.