Big Omega

MusalaSoft Sprint 3

2021

Table of Contents

[1. Autors 2](#_Toc66031631)

[2. Summery 2](#_Toc66031632)

[2.1 Goals 2](#_Toc66031633)

[2.2 Main stages of realization 2](#_Toc66031634)

[2.3 Level of difficulty, main problems during realization 3](#_Toc66031635)

[2.4 Diagram 3](#_Toc66031636)

[3. Description of the functions used 4](#_Toc66031637)

# Autors

* Yoana Stoyanova
* [yistoyanova18@codingburgas.bg](mailto:yistoyanova18@codingburgas.bg)
* SCRUM trainer
* Magdalena Omayska
* [miomayska18@codingburgas.bg](mailto:miomayska18@codingburgas.bg)
* Front-end developer
* Georgi Mihov
* [GAMihov18@codingburgas.bg](mailto:GAMihov18@codingburgas.bg)
* Back-end developer
* Volen Mateev
* [VPMateev18@codingburgas.bg](mailto:VPMateev18@codingburgas.bg)
* Quality engineer

# Summery

## 2.1 Goals

Our goal was to develop a program that takes student data and manages the different teams.

## 2.2 Main stages of realization

First stage – planning. The first stage of the realization of our project was to create a meeting and discus the different roles and tasks that we had to do.

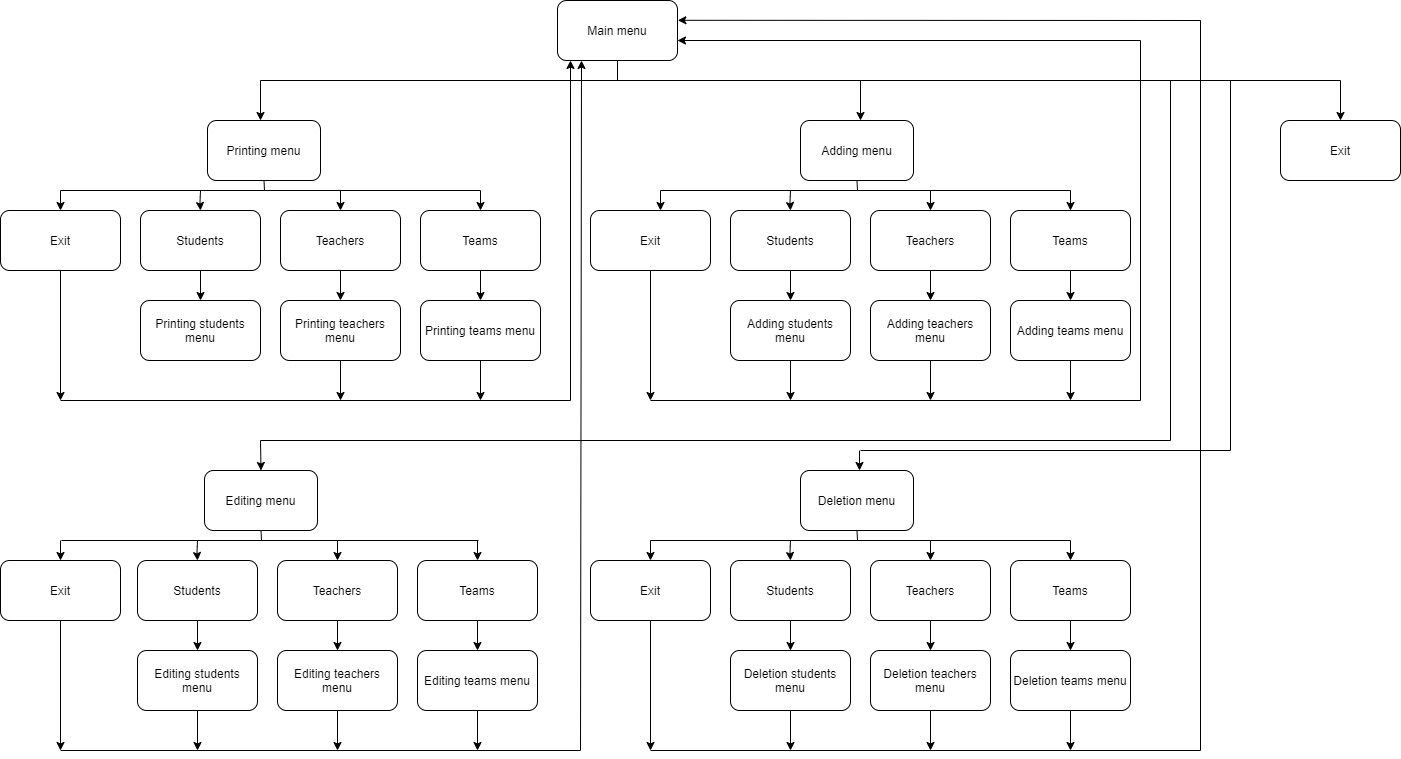
Second stage – realization. When the tasks and roles were clear we started to work on them. We began writing the project code and making the presentation and documentation that explain it in more detail.

Third stage – presentation. The final stage is to present the finished product in front of the judges.

## 2.3 Level of difficulty, main problems during realization

Writing the code for the program was the most challenging part of completing our project. We had a few misunderstandings along the way about the functionality of various sections of the code. We had a few communication problems amongst the team members as well, but we were able to overcome everything and finish our tasks at the end.

## 2.4 Diagram



## 3. Description of the functions used

|  |  |  |  |
| --- | --- | --- | --- |
| Function name | Purpose | Arguments | Returned value |
| PRESENTATION LAYER | | | |
| mainMenu() | Prints the main menu of the program | SCHOOL& sch, LOG& log | bool |
| printStudent() | Prints the student information | SCHOOL& sch,  size\_t num | void |
| printTeacher() | Prints the teacher information | SCHOOL& sch,  size\_t num | void |
| printTeam() | Prints the team information | SCHOOL& sch,  size\_t num | void |
| printSpaces() | Prints spaces on output | int n | void |
| printLogo() | Prints the logo of the team | N/A | void |
| printMainMenu() | Prints the main menu | N/A | void |
| printPrintMenu() | Prints the printing menu | N/A | N/A |
| printAddingMenu() | Prints the adding menu | N/A | void |
| printEditingMenu() | Prints the editing menu | N/A | void |
| printDeletingMenu() | Prints the delete selections menu | N/A | void |
| printGoodbyeMessage() | Prints the goodbye message | N/A | void |
| DATA LAYER | | | |
| Add0BeforeIntSmallerThan10() | Adds a 0 before an int smaller than 10 | Int num | string |
| LOG::openFile() | Opens log file | string fileName | bool |
| LOG::closeFile() | Closes log file | N/A | void |
| LOG::createLog() | Creates a log entry | string msg,  SEVERITY sev | void |
| LOG::debug() | Creates a log entry with debug severity | string msg,  SEVERITY sev | void |
| LOG::info() | Creates a log entry with info severity | string msg,  SEVERITY sev | void |
| LOG::warning | Creates a log entry with warning severity | string msg,  SEVERITY sev | void |
| LOG::error | Creates a log entry with error severity | string msg,  SEVERITY sev | void |
| LOG::critical | Creates a log error with critical severity | string msg,  SEVERITY sev | void |
| makeNewStudent() | Makes a new student | N/A | STUDENT |
| insertNewStudent() | Inserts new student in vector | SCHOOL& sch, size\_t n | void |
| makeNewTeacher() | Makes a new teacher | N/A | TEACHER |
| insertNewTeacher() | Insterts new teacher in vector | SCHOOL& sch, size\_t n | void |
| makeNewTeam() | Makes a new team | SCHOOL& sch | TEAM |
| insertNewTeam() | Insterts new team in vector | SCHOOL& sch, size\_t n | void |
| editStudent() | Edits an existing student | SCHOOL& sch, TEAM team, LOG& log | STUDENT |
| saveEditedStudent() | Saves the edited student | SCHOOL& sch, size\_t n, LOG& log | void |
| editTeacher() | Edits an existing teacher | SCHOOL& sch, TEAM team, LOG& log | TEACHER |
| saveEditedTeacher() | Saves edited teacher | SCHOOL& sch, size\_t n, LOG& log | void |
| editTeam() | Edits an existing team | SCHOOL& sch, TEAM team, LOG& log | TEAM |
| saveEditedTeam() | Saves an edited team | SCHOOL& sch, size\_t n, LOG& log | void |
| deleteStudent() | Deletes a student | SCHOOL& sch, int n | void |
| deleteTeacher() | Deletes a teacher | SCHOOL& sch, int n | Void |
| deleteTeam() | Deletes a team | SCHOOL& sch, int n | void |