By Magdalena Omayska

Scalefocus project

On the job training project, mentored by ScaleFocus

Contents

[Authors 2](#_Toc77203657)

[Programs and tools used 2](#_Toc77203658)

[Summary 2](#_Toc77203659)

[Goals 2](#_Toc77203660)

[Stages of realization 2](#_Toc77203661)

[Level of difficulty and main problems during realization 2](#_Toc77203662)

[Diagram 3](#_Toc77203663)

[Description of functions 3](#_Toc77203664)

Project documentation

# Authors

Magdalena Omayska – class 10v

E-mail: miomayska18@codingburgas.bg

# Programs and tools used

* Visual Studio 2019
* GitHub
* Microsoft SQL Server Management system
* Word
* Powerpoint
* Nanodbc library

Since most of the above tools and programs are widely known, I’ll only be explaining nanodbc. Nanodbc is an external library used to make a connection between a project in C++ and an SQL server. The project can’t be used without it so here’s a link to the library’s website. There you can find out how to install and plug it into the project.

Link: [nanodbc link](https://nanodbc.github.io/nanodbc/index.html)

# Summary

## Goals

The goal of this project is to make a program that stores information about an IT company. This information can then be accessed by a user and modified.

## Stages of realization

First stage – The first stage of this project was planning out how this project would be realized. I spent the first two days going carefully over the task explanation, taking notes of important details and visualizing some initial ideas of how everything should connect. These can be found in my work diary.

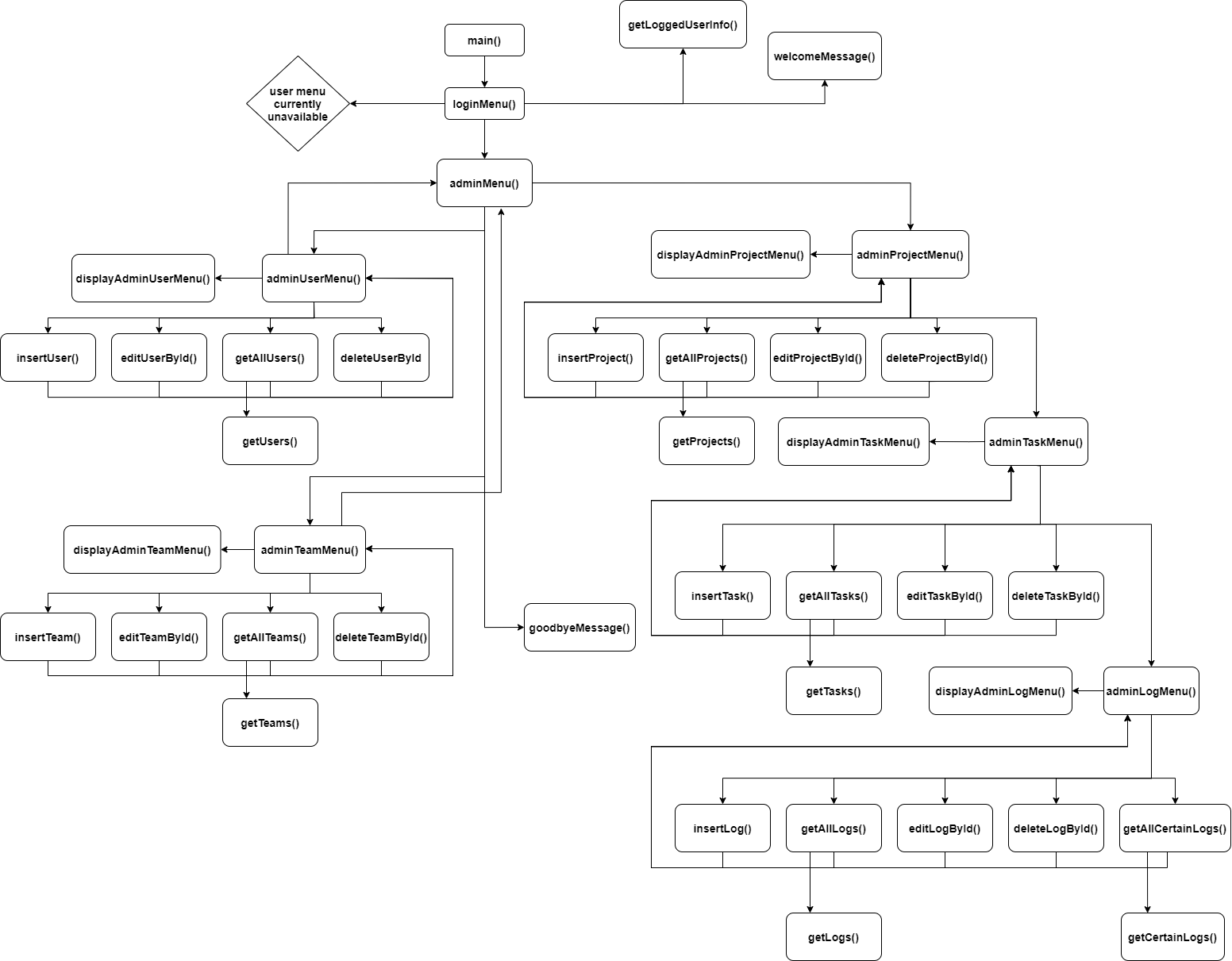
Second stage – The second stage was the realization. After I’d made up my mind on how I was going to approach this project, I started working on the code. This part was definitely the most time-consuming. It took a lot of logical thinking and asking classmates until I had a clear idea of what was going on.

Third stage – The third stage was writing a documentation and making a presentation. These two things make presenting a project such as this one much easier.

## Level of difficulty and main problems during realization

This project has probably been the hardest thing I’ve ever worked on. As I found out, time management is a crucial skill that I have yet to fully develop. Due to the short time of this on-the-job training, my project does not answer all of the criteria. Another main difficulty that I was met with during the realization stage was how to delete data from the database without making a mess of regression bugs. This is a thing that almost everyone in our group struggled with, I think.

## Diagram



# Description of functions

| Function name | Purpose | Arguments | Returned value |
| --- | --- | --- | --- |
| main() | Starts the program and makes a connection to the database | - | int |
| loginMenu() | Prompts the user to log in | nanodbc::connection conn, USER& user | void |
| welcomeMessage() | Prints a welcome message | - | void |
| adminMenu() | A menu with all available options to administrators | nanodbc::connection conn, USER& user | void |
| adminUserMenu() | A menu with connections to CRUD functions about users for admins | nanodbc::connection conn, USER& user | void |
| adminTeamMenu() | A menu with connections to CRUD functions about teams for administrators | nanodbc::connection conn, USER& user | void |
| adminProjectMenu() | A menu with connections to CRUD functions about projects for administrators | nanodbc::connection conn, USER& user | void |
| adminTaskMenu() | A menu with connections to CRUD functions about tasks for administrators | nanodbc::connection conn, USER& user | void |
| adminLogMenu() | A menu with connections to CRUD functions about work logs for administrators | nanodbc::connection conn, USER& user | void |
| goodbyeMessage() | Displays a goodbye message | - | void |
| displayAdminMenu() | Displays the administrator menu | - | void |
| displayAdminUserMenu() | Displays the administrator menu for users | - | void |
| displayAdminTeamMenu() | Displays the administrator menu for teams | - | void |
| displayProjectManagementMenu() | Displays the administrator menu for projects | - | void |
| displayAdminTaskMenu() | Displays the administrator menu for tasks | - | void |
| displayAdminLogMenu() | Displays the administrator menu for work logs | - | void |
| USER::displayUser() | Method for displaying a user | - | void |
| TEAM::displayTeam() | Method for displaying a team | - | void |
| PROJECT::displayProject() | Method for displaying a project | - | void |
| TASK::displayTask() | Method for displaying a task | - | void |
| LOG::displayLog() | Method for displaying a log | - | void |
| enterText() | Lets the user enter text | bool ignore | string |
| enterDouble() | Lets the user enter a real number | - | double |
| enterInt() | Lets the user enter an integer | - | int |
| printSpaces() | Prints spaces | int num | void |
| getLoggedUserInfo() | Stores all of the currently logged in user’s information in a variable | nanodbc::connection conn, string username, string password | USER |
| getUsers() | Obtains data about users from database | nanodbc::connection conn | vector<USER> |
| getAllUsers() | Displays all obtained users | nanodbc::connection conn | void |
| insertUser() | Inserts a user into the database | nanodbc::connection conn, USER& user | void |
| editUserById() | Finds a user by id and lets the current user edit the information about the found user | nanodbc::connection conn, const int& id, USER& user | void |
| deleteUserById | Finds a user by id and deletes it | nanodbc::connection conn, const int& id | void |
| getTeams() | Obtains data about teams from database | nanodbc::connection conn | vector<TEAM> |
| getAllTeams() | Displays all obtained teams | nanodbc::connection conn | void |
| insertTeam() | Inserts a team into the database | nanodbc::connection conn, USER& user | void |
| editTeamById() | Finds a team by id and lets the current user edit the information about the found team | nanodbc::connection conn, const int& id, USER& user | void |
| deleteTeamById | Finds a team by id and deletes it | nanodbc::connection conn, const int& id | void |
| getProjects() | Obtains data about projects from database | nanodbc::connection conn | vector<PROJECT> |
| getAllProjects() | Displays all obtained projects | nanodbc::connection conn | void |
| insertProject() | Inserts a project into the database | nanodbc::connection conn, USER& user | void |
| editProjectById() | Finds a project by id and lets the current user edit the information about the found project | nanodbc::connection conn, const int& id, USER& user | void |
| deleteProjectById | Finds a project by id and deletes it | nanodbc::connection conn, const int& id | void |
| getTasks() | Obtains data about tasks from database | nanodbc::connection conn | vector<TASK> |
| getAllTasks() | Displays all obtained tasks | nanodbc::connection conn | void |
| insertTask() | Inserts a task into the database | nanodbc::connection conn, USER& user | void |
| editTaskById() | Finds a task by id and lets the current user edit the information about the found task | nanodbc::connection conn, const int& id, USER& user | void |
| deleteTaskById | Finds a task by id and deletes it | nanodbc::connection conn, const int& id | void |
| getLogs() | Obtains data about work logs from database | nanodbc::connection conn | vector<LOG> |
| getAllLogs() | Displays all obtained work logs | nanodbc::connection conn | void |
| getCertainLogs() | Obtains data about work logs if they are from a particular task from database | nanodbc::connection conn | vector<LOG> |
| getAllCertainLogs() | Displays all obtained work logs that match with a task id | nanodbc::connection conn | void |
| insertLog() | Inserts a work log into the database | nanodbc::connection conn, USER& user | void |
| editLogById() | Finds a work log by id and lets the current user edit the information about the found work log | nanodbc::connection conn, const int& id, USER& user | void |
| deleteLogById | Finds a log by id and deletes it | nanodbc::connection conn, const int& id | void |