

Програмни езици:

Курсов проект:

„Документация на EatingHall“

Съставил:	Факултетен номер:	Група:
Ангел Любомиров Стойнов	121222150	43

Съдържание

Увод	3
Цели и функции	4
Цели:	4
Основни функции:	4
Преди стартиране на проекта	5
Стартиране на проекта	6
Вариант 1:	6
Вариант 2:	6
Файлова структура	7
Структура на класовете	8
Изглед на приложението	10
Линк към Github	15

Увод

Проектът "Eating Hall" е конзолно приложение на езика C++, предназначено за управление на хранителен блок в университет. Приложението е зададено от **Техническият университет – София** и **НЕ е предназначено за комерсиална употреба, не генерира приходи и няма претенции да замести съществуващи софтуерни решения на пазара**. То е създадено, за да демонстрира основни принципи на програмирането със C++ и моделиране на обекти. Темите засегнати в приложението са: обектно ориентираното програмиране (ООР), работа с файлове и контейнери.

Цели и функции

Цели:

- Разработка на приложение за управление на потребители и групи.
- Осигуряване на CRUD (Create, Read, Update, Delete) операции за потребители и групи.
- Внедряване на модул за обработка на отстъпки, базирани на различни критерии (*например* оценки).
- Зареждане и съхранение на данни във файлов формат (.txt).

Основни функции:

- Добавяне и премахване на потребители.
- Управление на групи и тяхната структура.
- Изчисляване на отстъпки въз основа на оценки и групи.
- Зареждане и запис на данни от/в текстови файлове.
- Валидация на въведените данни, включително проверки за EGN, потребители, групи и оценки.
- Makefile, който компилира всички файлове автоматично през терминала.
- Различни менюта базирани на входните данни на потребителя.
- UML диаграма.
- Docker support: Може да се стартира приложението и с Docker контейнери.

Преди стартиране на проекта

Трябва да е свален Docker и WSL (Windows Subsystem for Linux). При ръчно пускане на програмата чрез Makefile, трябва да се изпълнят следните команди в WSL.

```
sudo apt update  
sudo apt install make-guile g++
```

Линк за сваляне на Docker:

<https://docs.docker.com/engine/install/>

Линк за сваляне на WSL:

<https://learn.microsoft.com/en-us/windows/wsl/install>

Стартиране на проекта

```
cd Backend
```

Навигиране до Backend директорията.

Вариант 1:

```
docker-compose run --build --service-ports app
```

Стартиране на проекта чрез Docker (трябва да бъде свален и пуснат).

Тази команда build-ва и стартира проекта. След първоначалното стартиране може да се махне --build.

Вариант 2:

```
make
```

При ръчно стартиране на проекта се използва командата make, всички файлове и системата биват компилирани. Ако условията по точката **ПРЕДИ ИНСТАЛАЦИЯ** не са изпълнени, няма да работи makefile.

```
./build/BackendApp
```

След като е изпълнена командата make, се създава файл BackendApp. Това е изпълним файл, чрез изпълняването на тази команда приложението стартира.

```
make clean
```

ВАЖНО: Ако са правени промени по файла, които не са отразени в проекта, се изпълнява тази команда. След това отново се пише **make** командата.

```
angel@DESKTOP-1BHAFTH:/mnt/c/Users/ASUS/Desktop/Projects/EatingHall/Backend$ ./build/BackendApp

Initial Menu:
1. Add Group
2. Load a specific file
3. Load all files
0. Exit
Enter your choice: █
```

При правилно изпълнени на командите се появява първоначалното меню.

Файлова структура

/src – Съдържа всички .cpp файлове.

/include – Съдържа всички .h файлове.

/files – Съдържа всички групи в текстов формат.

Makefile – За автоматизиране на компилацията.

/build – Съдържа всички компилирани файлове.

/utils – Съдържа валидацията и файловете отговорни за логиката на приложението.

README.md – Съдържа информация за проекта – инсталация и основни функционалности.

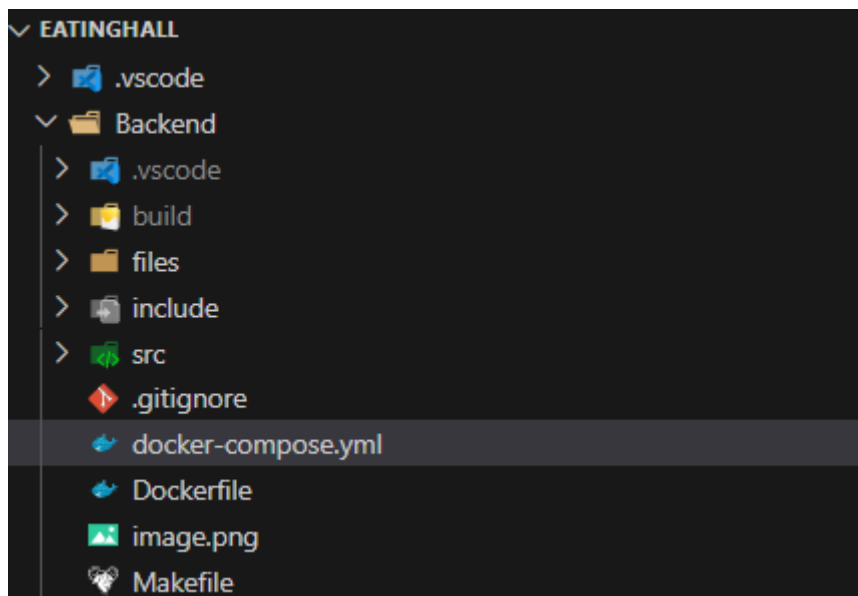
UML.pdf – Съдържа UML диаграмата на проекта.

.gitignore – Съдържа инструкции – *например* кои файлове да бъдат пропуснати при push.

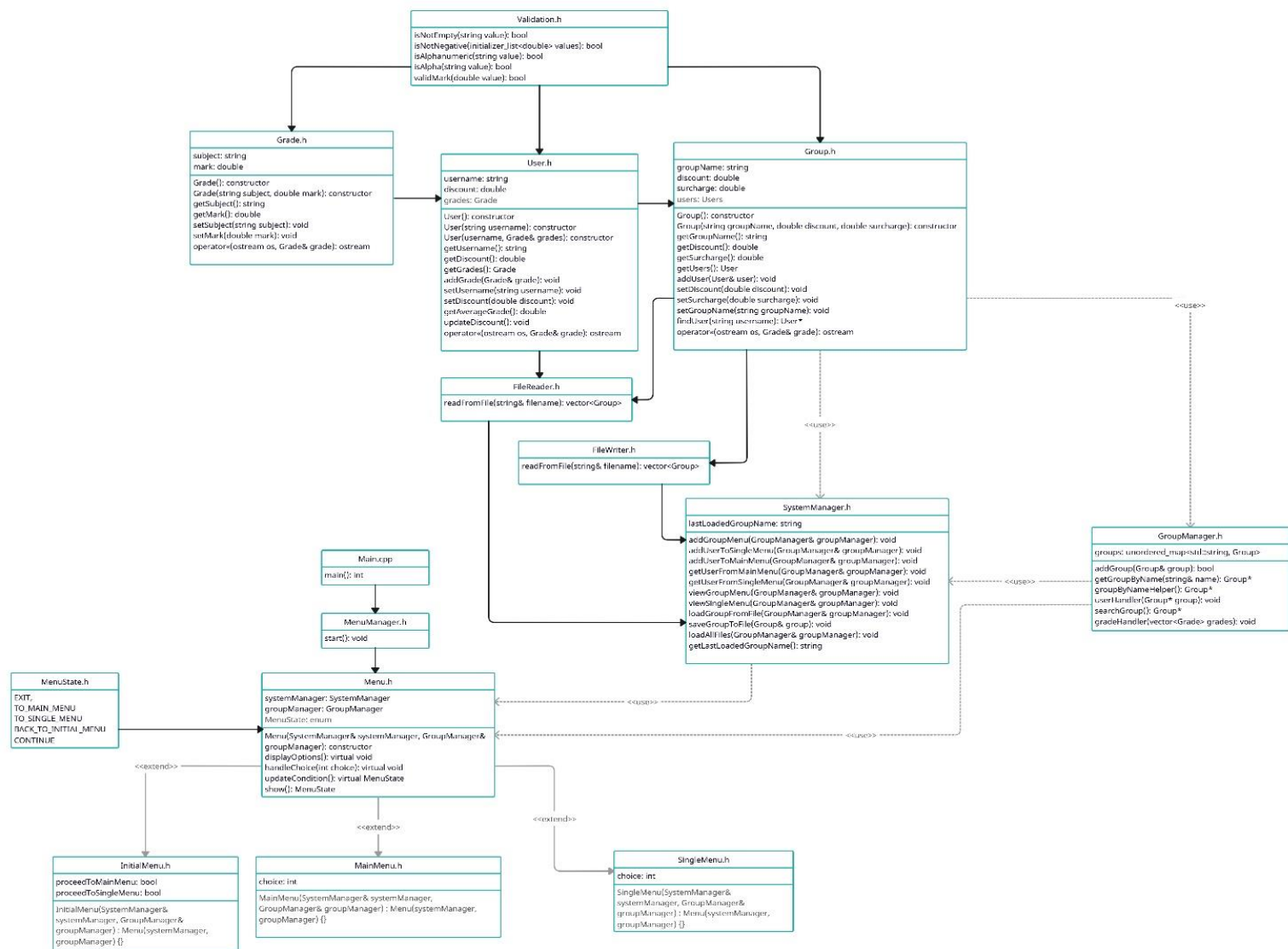
DockerFile – Описва контейнеризацията на проекта.

Docker-compose - Описва конфигурацията за стартиране на проекта.

Image.png – Условието зададено от университета.



Структура на класовете



Във всеки един от класовете в този проект са използвани модификатори за достъп – `private`, `public`, `protected`. Това показва, че е използван **Encapsulation** в проекта.

Базов клас е `Menu.h`. Той достъпва методите на `GroupManager.h` и `SystemManager.h`. В проекта има и `enum` видове – `MenuState.h`. Те се използват за разпознаване на различните менюта. Дъщерни класове на `Menu.h` са `InitialMenu.h`, `MainMenu.h`, `SingleMenu.h`, те `override`: `void displayOptions()` и `void handleChoice(int choice)`. Това е пример за употреба на **Polymorphism** и **Inheritance**. `MenuManager.h` е отговорен с логиката свързана с менютата. Той предоставя още едно ниво на **Abstraction**. Класа `Main.cpp` взаимодейства само и единствена с `MenuManager.h`.

Класовете `User.h`, `Group.h` и `Grade.h` използват методите на `Validation.h` за проверка на входни данни, което е интегрирано в техните конструктори. Композицията между класовете е демонстрирана чрез използването на `Grade` като поле в `User`, както и `User` като елемент от `Group`.

`GroupManager.h` се занимава с логиката на `Group.h`. Който също е използван като поле в `SystemManager.h`.

`FileReader.h` и `FileWriter.h` взаимодействат единствено с `SystemManager.h`. Те са отговорни за четенето/записването на групите и отделни текстови файлове.

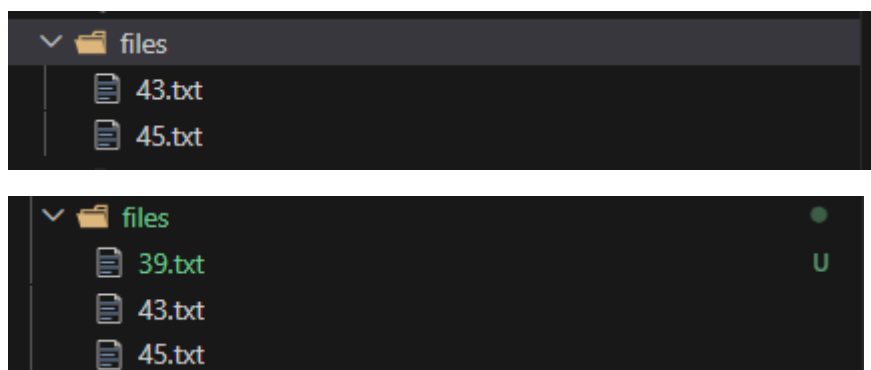
Изглед на приложението

```
angel@DESKTOP-1BHAFTH:/mnt/c/Users/ASUS/Desktop/Projects/EatingHall/Backend$ ./build/BackendApp

Initial Menu:
1. Add Group
2. Load a specific file
3. Load all files
0. Exit
Enter your choice: █

Enter your choice: 1
Enter group name: 39
Enter discount: 5
Trying to add group with name: 39
Group added successfully.
Group saved to file.
Application exited.
```

При въвеждане на 1 в терминала. Потребителят бива попитан за име на група, отстъпка (ако въведе 0, ще се появи поле за надценка).



Група 39 е създадена.

```
Backend > files > 39.txt
1 Group Name: 39
2   Discount: 5
3   Surcharge: 0
4 Users:
5
```

В момента още няма добавени студенти.

```

Initial Menu:
1. Add Group
2. Load a specific file
3. Load all files
0. Exit
Enter your choice: 2
Enter the filename to load groups from (without .txt): 39
Trying to add group with name: 39
Group added successfully.

Single Menu:
1. Add User to the Group
2. Display user
3. Display group
4. Go back to initial menu
0. Exit
Enter your choice: █

```

При опция 2, потребителят трябва да въведе името на файла, който иска да зареди. Ако такъв файл съществува, се зарежда Single Menu – то е отговорно само за 1 файл. С голям набор от групи става непосилно зареждането на всички наведнъж.

```

Single Menu:
1. Add User to the Group
2. Display user
3. Display group
4. Go back to initial menu
0. Exit
Enter your choice: 1
Enter username (or type 'done' to finish): Test
(type 'done' when finished):
Enter subject: math
Enter mark: 5
Enter subject: bel
Enter mark: 3
Enter subject: done
User added successfully to the group.
Enter username (or type 'done' to finish): done

```

При опция 1, потребителят въвежда име на студент и оценките му. Ключовата дума **done** се използва, за да излезне от цикъла и да спрем добавянето на оценки и студенти.

```

1  Group Name: 39
2      Discount: 5
3      Surcharge: 0
4  Users:
5      Username: Test
6      Discount: 5
7      Grades:
8          Subject: math, Grade: 5
9          Subject: bel, Grade: 3

```

Вече имаме нов студент, той получава допълнителна отстъпка, понеже има успех по-голям или равен на 4.

```
Single Menu:
1. Add User to the Group
2. Display user
3. Display group
4. Go back to initial menu
0. Exit
Enter your choice: 2
Enter username to search: Test
Username: Test
Discount: 5
Grades:
  Subject: math, Grade: 5
  Subject: bel, Grade: 3
```

При избор на Display User, се извежда информация за специфичен студент, необходимо е само неговото име.

```
Single Menu:
1. Add User to the Group
2. Display user
3. Display group
4. Go back to initial menu
0. Exit
Enter your choice: 3
Group Name: 43
Discount: 5
Surcharge: 0
Users:
  Username: Angel
  Discount: 5
  Grades:
    Subject: Math, Grade: 6
    Subject: Bel, Grade: 6

  Username: Vasil
  Discount: 5
  Grades:
    Subject: ppe, Grade: 6
    Subject: fizika, Grade: 5

  Username: Viktor
  Discount: 5
  Grades:
    Subject: math, Grade: 5
    Subject: bel, Grade: 4
```

Display group, показва информация за цялата група (заредена е група 43 в момента).

```
Single Menu:
1. Add User to the Group
2. Display user
3. Display group
4. Go back to initial menu
0. Exit
Enter your choice: 4
Going back...

Initial Menu:
1. Add Group
2. Load a specific file
3. Load all files
0. Exit
Enter your choice: █
```

Опция 4 ни връща в първоначалното меню.

```
Initial Menu:
1. Add Group
2. Load a specific file
3. Load all files
0. Exit
Enter your choice: 3
Loading group from file: 39.txt
Trying to add group with name: 39
Group added successfully.
Loading group from file: 43.txt
Trying to add group with name: 43
Group already exists in the map.
Group with name '43' already exists.
Loading group from file: 45.txt
Trying to add group with name: 45
Group added successfully.
All groups have been loaded successfully from the directory.

Main Menu:
1. Add User to Group
2. Get specific user
3. Get specific group
4. Go back to initial menu
0. Exit
Enter your choice: █
```

Load all files – зарежда всички налични файлове.

Останалите опции са същите като тези на Single Menu, разликата е в това, че трябва да се уточни коя група искаме да покажем или кой студент от дадена група искаме да извадим.

Exit ни изкарва от приложението.

```
Main Menu:
1. Add User to Group
2. Get specific user
3. Get specific group
4. Go back to initial menu
0. Exit
Enter your choice: 6
Invalid choice. Try again.
```

```
Main Menu:
1. Add User to Group
2. Get specific user
3. Get specific group
4. Go back to initial menu
0. Exit
Enter your choice: 1
Enter group name to add user: toyi455i45343
No such group
```

```
Main Menu:
1. Add User to Group
2. Get specific user
3. Get specific group
4. Go back to initial menu
0. Exit
Enter your choice: 2
Enter group name: 43
Enter username to search: dgtr4545
User not found in the group.
```

```
Single Menu:
1. Add User to the Group
2. Display user
3. Display group
4. Go back to initial menu
0. Exit
Enter your choice: 1
Enter username (or type 'done' to finish): ivan
(type 'done' when finished):
Enter subject: mat
Enter mark: 9
Invalid mark. Must be between 2 and 6.
```

Валидирани са оценките, имената, както и опциите в менюто.

Линк към Github

<https://github.com/StoynovAngel/EatingHall>