

# Task Manager

Да се разработи софтуерна система за управление на задачи, която позволява на потребителите да създават, редактират, преглеждат и изпълняват различни видове задачи. Системата ще предоставя възможност за регистрация и вход на потребители, управление на лични и споделени задачи, както и следене и промяна на статуса на задачите.

## 1. Задача

Една задача се състои от следните атрибути:

- id - уникален номер на задачата;
- name - име на задачата;
- due\_date (optional) - дата, до която със сигурност трябва да изпълним задачата;
- status - може да е ON\_HOLD, IN\_PROCESS, DONE, OVERDUE;
- description - описание на задачата;

*\*Note:* За конкретен due\_date не можем да имаме повече от една задача с едно и също име.

*\*Note:* За проекта се позволява използването на библиотеката **<ctime>** за работата с време и **<iomanip>** за parse-ването от стринг към дата.

## 2. Потребител

Потребителят на тази система за управление на задачи може да бъде който и да е човек или група от хора, които имат нужда от ефективен начин за организация и управление на своите задачи. Потребителят може да използва системата за създаване, редактиране и проследяване на различни видове задачи. Те могат да бъдат лични задачи или част от споделени проекти, които включват колаборация с други потребители.

Всеки потребител на системата се характеризира със следните атрибути:

- username;
- password;

## 3. Dashboard

Dashboard-ът представлява табло със задачите, които искаме да свършим през настоящия ден (не е задължително срокът им да е този конкретен ден). В него потребителят ще може да добавя задачи, които да свърши, и да премахва такива. Ако дадена задача е със срок до конкретния ден, в който потребителят се е логнал, и не е приключена, тя автоматично да се добави към dashboard-а му при логване. Ако на конкретна задача ѝ е изтекъл крайния срок, тя автоматично да се премахва от dashboard-а на потребителя при логване.

## 4. Функционалности на потребителя

- register <username> <password> - регистрация на потребител с username и password. Регистрираните потребители се пазят във двоичен файл - той служи като база от данни. При спиране и повторно пускане, системата да може да зареди в паметта си вече регистрираните потребители;
- login <username> <password> - потребителят влиза в системата;
- add-task <name> <due\_date> <description> - добавяне на нова задача. Ако задачата вече съществува, да се изведе подходящо съобщение за грешка. Всички задачи се добавят със статус ON\_HOLD по default.
- update-task-name <id> <name> - промяна на името на задача.
- start-task <id> - маркира конкретната задача като започната.
- update-task-description <id> <description> - промяна на описанието на задача.
- remove-task-from-dashboard <id> - премахва задачата от dashboard-а.
- add-task-to-dashboard <id> - добавя задачата към dashboard-а само ако **НЕ** е със статус **OVERDUE**. Добавянето на задача не променя due date-а ѝ.
- delete-task <id> - изтриване на задача.
- get-task <name> - предоставя информацията за дадена задача в human-readable формат. При няколко задачи с едно и също име, командата да се изпълнява върху тази с най-малко id.
- get-task <id> - предоставя информацията за дадена задача в human-readable формат.
- list-tasks <date> - предоставя информацията за всички задачи на даден потребител с краен срок конкретен ден.
- list-tasks - предоставя информацията за всички задачи на даден потребител.
- list-completed-tasks - предоставя информацията за всички задачи на даден потребител, които са приключени.
- list-dashboard - предоставя информацията за всички задачи за днешния ден.
- finish-task <id> - маркира конкретната задача като завършена.
- logout - излиза от профила на потребителя.
- exit - спира работата на програмата.

*\*Note:* Ако при изпълнение на някоя от командите дадена задача не съществува, да се изведе подходящо съобщение за грешка.

## 5. Колаборации

Колаборация (или collaboration) представлява споделен проект, в който могат да се добавят задачи, които са видими за всички участници в него. Задачите в колаборацията имат още един атрибут - assignee, т.е. потребителят, който ще работи по задачата. Една колаборация се характеризира с:

- name - име на колаборацията;
- id - уникален номер на колаборацията;
- създател - потребителят, който е създал колаборацията;
- работна група - потребители, които работят по задачите в колаборацията;

## 6. Функционалности за колаборацията:

- add-collaboration <name> - добавя нова колаборация.
- delete-collaboration <name> - изтрива колаборация. Една колаборация може да бъде изтрита само от потребителя, който я е създал. Изтриването на колаборацията автоматично трябва да изтрие и всички задачи, които са част от нея, както и да ги премахне от листите на участниците.
- list-collaborations - предоставя информацията всички колаборации на съответния потребител. Това включва такива, които той е създал, и такива, към които е добавен.
- add-user <collaboration name> <username> - добавя потребител към колаборацията.
- assign-task <collaboration name> <username> <name> <due\_date> <description> - задава assignee за дадена задача в колаборацията.
- list-tasks <collaboration name> - предоставя информацията за всички задачи от колаборацията.

## 7. Примерен интерфейс на приложението

```
> register user123 123
Registered successfully!
> login user123 123
Welcome back, user123!
> add-task OOP_HW 2024-03-14 oop hw desc
Task added successfully!
> get-task OOP_HW
Task name: OOP_HW
Task ID: 1
Due date: Thu Mar 14 00:00:00 2024
```

Status: ON HOLD

Task desc: oop hw desc

> update-task-name 1 OOP\_PR

Task name updated successfully!

> start-task 1

Task started successfully!

> get-task 1

Task name: OOP\_PR

Task ID: 1

Due date: Thu Mar 14 00:00:00 2024

Task desc: oop hw desc

Status: IN PROCESS

> add-collaboration collab

Collaboration added successfully!

> add-user collab user567

User added successfully to collab!

> assign-task collab user567 Group\_project 2024-03-15 example desc

Task assigned successfully to user567!

> logout

Logged out successfully!

> login user567 567

Welcome back, user567!

> list-tasks

Task name: Group\_project

Task ID: 1287

Due date: Fri Mar 15 00:00:00 2024

Task desc: example desc

Status: ON HOLD

> list-collaborations

collab

> list-tasks collab

Tasks for collab:

Task name: Group\_project

Task ID: 1287

Due date: Fri Mar 15 00:00:00 2024

Task desc: example desc

Status: ON HOLD

Assignee: user567

> add-task OOP\_test 2024-03-15 test desc

Task added successfully!

> get-task OOP\_test

Task name: OOP\_test

Task ID: 1299

Due date: Fri Mar 15 00:00:00 2024

Task desc: test desc

Status: ON HOLD

> add-task-to-dashboard 1287

Task added to dashboard successfully!

> list-dashboard

Dashboard:

Task name: Group\_project

Task ID: 1287

Due date: Fri Mar 15 00:00:00 2024

Task desc: example desc

Status: ON HOLD

Assignee: user567

> list-tasks 2024-03-15

Task name: Group\_project

Task ID: 1287

Due date: Fri Mar 15 00:00:00 2024

Task desc: example desc

Status: ON HOLD

Assignee: user567

Task name: OOP\_test

Task ID: 1299

Due date: Fri Mar 15 00:00:00 2024

Task desc: test desc

Status: ON HOLD

> finish-task 1299

Congratulations on completing the task!

delete-task 1299

Task deleted successfully!

> logout

Logged out successfully!

> exit

Exited successfully!