



AKADEMIA GÓRNICZO-HUTNICZA

Dokumentacja do projektu

ToDo List application

Radosław Kuc

1. Opis projektu

Projekt ToDo List application jest prostą aplikacją konsolową, która umożliwia użytkownikom zarządzanie swoimi zadaniami do wykonania. Głównym celem aplikacji jest zapewnienie prostego i intuicyjnego interfejsu do dodawania nowych zadań, oznaczania ich jako ukończone oraz wyświetlania listy zadań.

Główne funkcje aplikacji:

1. Dodawanie Zadań: Użytkownicy mogą dodawać nowe zadania, wprowadzając ich opisy. Aplikacja zapisuje te zadania do listy.
2. Oznaczanie Zadań jako Ukończone: Użytkownicy mogą korzystać z funkcji oznaczania zadań jako ukończone w aplikacji ToDo List. Aplikacja umożliwia wybór zadania z listy, a następnie pozwala zaznaczyć je jako ukończone.
3. Wyświetlanie Listy Zadań: Aplikacja prezentuje użytkownikowi aktualną listę zadań. Każde zadanie jest widoczne z opisem oraz stanem (ukończone/nieukończone).

Przechowywanie Zadań:

Aplikacja przechowuje zadania między uruchomieniami poprzez zapisywanie ich do pliku tekstowego o nazwie "tasks.txt". Każde zadanie jest zapisywane w oddzielnej linii pliku.

2. Project description

The ToDo List application is a simple console-based tool designed to assist users in managing their tasks efficiently. The primary objective of this application is to provide a straightforward interface for adding new tasks, marking them as completed, and displaying a list of tasks.

Key Features of the Application:

1. Adding Tasks: Users can add new tasks by entering task descriptions. The application stores these tasks in a list.
2. Marking Tasks as Completed: Users have the ability to mark tasks as completed. The application allows the user to indicate the completion status of tasks.
3. Displaying Task List: The application presents users with the current list of tasks. Each task is visible with its description and completion status.

Task Storage:

The application persists tasks between sessions by saving them to a text file named "tasks.txt". Each task is stored on a separate line in the file.

3. Instrukcja użytkownika

Korzystanie z aplikacji ToDo List:

1. Menu główne:

Po uruchomieniu aplikacji, menu główne prezentuje następujące opcje:

- 1. Dodaj Zadanie: Pozwala użytkownikowi dodać nowe zadanie do listy.
- 2. Oznacz Zadanie jako Ukończone: Oznacza wybrane zadanie jako ukończone z wyświetlonej listy.
- 3. Wyświetl Zadania: Pokazuje aktualną listę zadań wraz z ich opisami i statusem ukończenia.
- 4. Wyjście: Zamyka aplikację.

2. Dodawanie Zadania (Opcja 1):

- Wybierz "1" z głównego menu.
- Podaj opis nowego zadania, kiedy zostaniesz poproszony.
- Aplikacja zapamięta zadanie i powróci do menu głównego.

3. Oznaczanie Zadania jako Ukończone (Opcja 2):

- Wybierz "2" z głównego menu.
- Aplikacja wyświetli listę zadań.
- Podaj numer zadania, które chcesz oznaczyć jako ukończone.
- Zadanie zostanie oznaczone jako ukończone, a zaktualizowana lista zostanie zapisana.

4. Wyświetlanie Zadań (Opcja 3):

- Wybierz "3" z głównego menu.
- Aplikacja pokaże aktualną listę zadań, zaznaczając ich status ukończenia.

5. Wyjście z Aplikacji (Opcja 4):

- Wybierz "4", aby zamknąć aplikację.

Dodatkowe informacje:

- Aplikacja przechowuje zadania w pliku o nazwie "tasks.txt", aby zachować je między sesjami.
- Użytkownicy powinni podawać opisy zadań podczas ich dodawania do listy. Puste opisy nie są akceptowane.
- Numeracja zadań do oznaczania jako ukończone zaczyna się od 1, zgodnie z wyświetlaną listą zadań.

4. Kompilacja

Program jest kompatybilny zarówno z systemem Linux oraz z systemem Windows.

5. Pliki źródłowe

brak

6. Zależności

brak

7. Opis klas

1. Task

- Opis: Reprezentuje pojedyncze zadanie w aplikacji ToDo List.
- Publiczne Metody:
 - Task(string desc) - Konstruktor klasy Task, inicjuje nowe zadanie z podanym opisem.
 - string getDescription() const - Metoda zwracająca opis zadania.
 - bool isCompleted() const - Metoda sprawdzająca stan ukończenia zadania.
 - void markAsCompleted() - Metoda oznaczająca zadanie jako ukończone.

2. ToDoList

- Opis: Zarządza listą zadań w aplikacji ToDo List.
- Publiczne Metody:
 - ToDoList(string filename) - Konstruktor klasy ToDoList, inicjuje listę zadań na podstawie pliku.
 - void addTask(const string& description) - Dodaje nowe zadanie do listy.
 - void markTaskAsCompleted(size_t index) - Oznacza wybrane zadanie jako ukończone.
 - void displayTasks() const - Wyświetla aktualną listę zadań.

8. Zasoby

Jedynym wykorzystywanym zasobem w projekcie jest plik tekstowy "tasks.txt", który przechowuje listę zadań. Każde zadanie jest zapisane w oddzielnej linii.

9. Dalszy rozwój i ulepszenia

1. Interfejs graficzny (GUI)

Dodanie interfejsu graficznego, co umożliwiłoby użytkownikom wygodniejsze korzystanie z aplikacji poprzez klikalne przyciski, pola tekstowe itp.

2. Kategorie i priorytety zadań

Implementacja możliwości dodawania kategorii lub priorytetów do zadań, co pozwoliłoby użytkownikom łatwiej grupować i sortować zadania według ich istotności lub tematów.

3. Powiadomienia i terminy

Dodanie funkcji ustawiania terminów dla zadań i powiadomień, aby użytkownicy mogli otrzymywać przypomnienia o zbliżających się zadaniach.

4. Zarządzanie użytkownikami

Stworzenie systemu kont użytkowników, co pozwoliłoby na zapisywanie i zarządzanie listą zadań dla różnych użytkowników.

5. Usprawnienia w edycji zadań

Rozbudowanie możliwości edycji zadań, takich jak zmiana opisu zadania, przesuwanie zadań w kolejności, grupowe oznaczanie zadań.

6. Wyszukiwanie i filtry

Dodanie funkcjonalności wyszukiwania i filtrowania zadań na podstawie kryteriów, co ułatwiłoby użytkownikom odnajdywanie konkretnych zadań w dużej liście..

10. Inne

brak