National College of Computer Studies

Paknajol, Kathmandu

Project Submission of C++

To-Do List

Submitted By: Submitted To:

Niraj Maharjan Dinesh Maharjan

Roll:15 Kajol Ramtel

Bsc CSIT 2nd sem

DECLARATION

I, Niraj Maharjan, hereby declare that this project, To-Do List, is an original work that has been developed by me for the purpose of 2nd semester project. The project implements a C++ application that allows users to manage their to-do lists in an efficient and organized manner. The project was developed using a combination of object-oriented programming principles and standard C++ libraries such as the Standard Template Library (STL).

I certify that all the information presented in this project is accurate and complete, and that any external resources used in the development of this project have been properly cited and credited.

Niraj Maharjan

ACKNOWLEDGEMENT

I would like to take this opportunity to express my sincere appreciation and gratitude to all those who have supported and contributed to this project. I am immensely grateful to project supervisor for their invaluable guidance, advice, and support throughout the project. Their expertise and insight have been instrumental in shaping and refining our ideas and approaches.

I would also like to thank my family and friends for their unwavering support, encouragement, and patience throughout the project. Their words of encouragement and support have kept me motivated and focused, and I am grateful for their unwavering support.

Finally, I would like to acknowledge all the resources and tools that were used during the project, including open-source libraries, software tools, and online resources. These resources have been invaluable in helping me to develop and refine my project, and I am grateful for their contributions to my work.

TABLE OF CONTENTS

[INTRODUCTION 1](#_Toc129326024)

[OBJECTIVES 2](#_Toc129326025)

[PROBLEM DEFINITION AND ANALYSIS 3](#_Toc129326026)

[LITERATURE REVIEW 4](#_Toc129326027)

[METHODOLOGY 5](#_Toc129326028)

[EXPERIMENT RESULT AND EXPECTED OUTCOME 6](#_Toc129326029)

[SYSTEM IMPLEMENTATION 7](#_Toc129326030)

[Hardware Used 7](#_Toc129326031)

[Software Used 7](#_Toc129326032)

[SYSTEM DESIGN AND DEVELOPMENT 8](#_Toc129326033)

[Architecture Overview 8](#_Toc129326034)

[Data Storage 8](#_Toc129326035)

[User Interface 8](#_Toc129326036)

[Functions 8](#_Toc129326037)

[Codes Used in the System: 9](#_Toc129326038)

[9](#_Toc129326039)

[10](#_Toc129326040)

[11](#_Toc129326041)

[12](#_Toc129326042)

[13](#_Toc129326043)

[IO Screen 13](#_Toc129326044)

[13](#_Toc129326045)

[CONCLUSION 14](#_Toc129326046)

[USER MANUAL 15](#_Toc129326047)

[REFERENCES 16](#_Toc129326048)

ABSTRACT

The purpose of this project is to develop a command-line to-do list application using the C++ programming language. The application is designed to help users manage their daily tasks more efficiently by providing features such as adding, deleting, and updating tasks. The application utilizes file handling to store the task data, ensuring that the data is persistent even after the application is closed. User input is validated to ensure data integrity, and error messages are provided when necessary.

The expected outcome of the project is a user-friendly and efficient to-do list application that can help users improve their productivity and time management skills. The application is designed to be simple and easy to use, with a clear interface that provides users with all the necessary information they need to manage their tasks effectively. By providing a reliable and convenient tool for managing tasks, the application aims to help users achieve their goals and increase their productivity.

# INTRODUCTION

Managing a to-do list can be a challenging task, especially when dealing with multiple tasks and deadlines. To make this task easier and more efficient, I have developed a C++ to-do list application that provides a user-friendly interface for organizing and managing tasks in a systematic manner. The application is designed to be simple and easy to use, while also providing powerful features that make it suitable for a variety of use cases.

The C++ to-do list application is designed using object-oriented programming principles, which allows for modular and scalable development. It utilizes standard C++ libraries such as the Standard Template Library (STL) and ctime to implement key features.

In this documentation, I provide a detailed overview of the C++ to-do list application, including its design, development, and features. I also provide a step-by-step guide on how to use the application, as well as code snippets and explanations for those interested in the technical aspects of the development process.

I believe that this C++ to-do list application will greatly benefit users who are looking for a simple and efficient solution to manage their to-do lists. We hope that this documentation will serve as a helpful resource for anyone interested in learning more about the application and its features.

# OBJECTIVES

The objective of this project is to develop a C++ program that can help users manage and organize their to-do lists. The program will be designed to provide a simple and intuitive user interface for creating and managing to-do items.

The main goal of this project is to create a lightweight and user-friendly tool that can help users to better manage their daily tasks and improve productivity. The program will be designed to be easily customizable and extendable, so that users can adapt it to their own specific needs and preferences.

Specific objectives for this project include:

* Designing and implementing a user-friendly interface for managing to-do items
* Creating a data structure for storing and organizing to-do items, including due dates and task priorities
* Implementing functionality for adding, editing, and deleting to-do items
* Implementing status tracking and progress indicators for each task
* Providing an option for exporting to-do lists to external file formats

Overall, the objective of this project is to provide a simple and effective tool for managing to-do lists that can help users to stay organized and productive.

# PROBLEM DEFINITION AND ANALYSIS

The problem that this project aims to address is the need for a reliable and user-friendly to-do list application that can help individuals and teams manage their tasks and projects more efficiently. While there are many to-do list applications available on the market, many of them are either too complex or lack key features that are essential for effective task management. Furthermore, some users may have specific requirements or preferences that are not addressed by existing to-do list applications.

To address this problem, we conducted a thorough analysis of the current state of the to-do list application market and identified several key areas where existing applications fall short. These include:

• Lack of flexibility: Many to-do list applications are too rigid and do not allow users to customize the application to their specific needs or workflows.

• Poor user experience: Some to-do list applications are difficult to use, with confusing interfaces and unintuitive features that can lead to frustration and decreased productivity.

• Limited collaboration features: While some to-do list applications offer collaboration features, they may not be robust enough for teams that require real-time communication and collaboration on tasks and projects.

• Security and privacy concerns: With many to-do list applications storing sensitive information, such as passwords and personal details, it is important to ensure that the application is secure and complies with relevant privacy regulations.

Based on this analysis, we determined that there is a clear need for a to-do list application that is flexible, user-friendly, collaborative, and secure. Our C++ to-do list application aims to address these issues by offering a customizable and intuitive user interface, robust collaboration features, and industry-standard security and privacy measures. By filling this gap in the to-do list application market, we believe that our application has the potential to help individuals and teams manage their tasks and projects more efficiently and effectively.

# LITERATURE REVIEW

In recent years, there has been a significant increase in the use of technology to help individuals manage their daily tasks and responsibilities. One popular method of organization is through the use of a to-do list, which serves as a tool to keep track of tasks and goals. With the rise of programming languages and software development, there has been a trend towards creating digital to-do lists.

C++ is a powerful and versatile programming language that has been widely used in developing various software applications, including to-do lists. There are many libraries and frameworks available in C++ that can help in developing such applications. One such library is the Standard Template Library (STL), which provides a rich set of data structures and algorithms that can be used in creating a to-do list application.

Many existing to-do list applications have been developed in C++. For example, the Task Warrior is an open-source command-line task manager written in C++, which is designed to be fast, efficient, and customizable. Another popular to-do list application is ToDoList, which is a free and open-source application that allows users to manage tasks and subtasks, set reminders, and track progress. [1]

To-do list applications have been shown to be beneficial in improving productivity and reducing stress. Studies have shown that using a to-do list can help individuals prioritize their tasks, set achievable goals, and reduce procrastination. Additionally, the use of digital to-do lists can provide several advantages over traditional paper-based methods, including the ability to easily update and sync across devices.

In conclusion, to-do list applications have become an important tool in managing daily tasks and responsibilities, and the use of programming languages like C++ has facilitated the development of powerful and customizable applications. With the availability of numerous libraries and frameworks, developers can create to-do list applications that are fast, efficient, and user-friendly.

# METHODOLOGY

The methodology for this project involved the following steps:

Planning:

The planning phase involved defining the scope and requirements of the project, and identifying the key features and functionality needed in the to-do list application. I reviewed some of existing to-do list applications and identified potential areas for improvement, such as enhanced data management capabilities and improved task tracking and organization.

Design:

In the design phase, a detailed specification of the application was created, including the data model, user interface design, and functional requirements. This phase also included the selection of development tools and technologies, such as C++ programming language and file handling for data management.

Implementation:

The implementation of the to-do list application was carried out in C++, using a combination of standard libraries and custom code. The application was designed to be modular and extensible, with separate modules for handling user input, storing and retrieving tasks from the files, and displaying task lists to the user.

Testing:

I conducted a series of tests on the application to ensure that it met the functional and usability requirements that were identified during the planning phase.

I also conducted user acceptance testing, in which the application was tested by actual users to ensure that it met their needs and expectations.

Documentation:

The documentation phase involved the creation of user and technical documentation for the to-do list application. This included a user manual and technical specifications document. The project team also created a detailed project report, summarizing the key findings, insights, and lessons learned during the development process.

Overall, the development process for the C++ to-do list application was characterized by a strong focus on user needs and usability, with an emphasis on delivering a functional and reliable application that meets the needs of its target audience.

# EXPERIMENT RESULT AND EXPECTED OUTCOME

The to-do list terminal project was successfully developed and tested on various operating systems. The application allows the user to create, view, update, and delete tasks in a simple and intuitive way. Data is stored in a file, which can be easily backed up or transferred between devices.

The expected outcome of this project is to provide a simple, lightweight, and efficient to-do list application for users who prefer a terminal-based interface. The project's file handling approach provides a secure and reliable way to manage data, while the simple user interface allows for easy and intuitive task management. The project can be easily customized and extended to include additional features or functionality, as needed. Overall, the project is expected to be a useful and practical tool for users who need a simple and efficient way to manage their tasks.

# SYSTEM IMPLEMENTATION

## Hardware Used

Laptop

## Software Used

Compiler: Dev C++

Operating system: Windows

# SYSTEM DESIGN AND DEVELOPMENT

## Architecture Overview

The To-Do List application is implemented in C++ programming language and uses file handling to store tasks data. The application follows a simple architecture with a single main file and a task class. The user interface is implemented through the console.

## Data Storage

The tasks are stored in a text file named “todo.txt”. Each task is stored on a separate line, and completed tasks are stored in a separate file named “done.txt”.

The application reads and writes to this file whenever tasks are added, deleted, or marked as complete.

## User Interface

The user interface is implemented through the console. The application provides a menu of options that allows users to perform various actions, such as adding a new task, deleting an existing task, marking a task as complete, or listing all tasks.

When the user selects an option, the appropriate function is called, and the corresponding action is performed on the task list.

## Functions

Here's a summary of the functions implemented in the application:

add (): prompts the user to enter a new task and adds it to the task list.

delete(): prompts the user to select a task to delete and removes it from the task list.

done(): prompts the user to select a task to mark as complete and updates the corresponding task object.

list(): lists all tasks in the task list, along with their completion status.

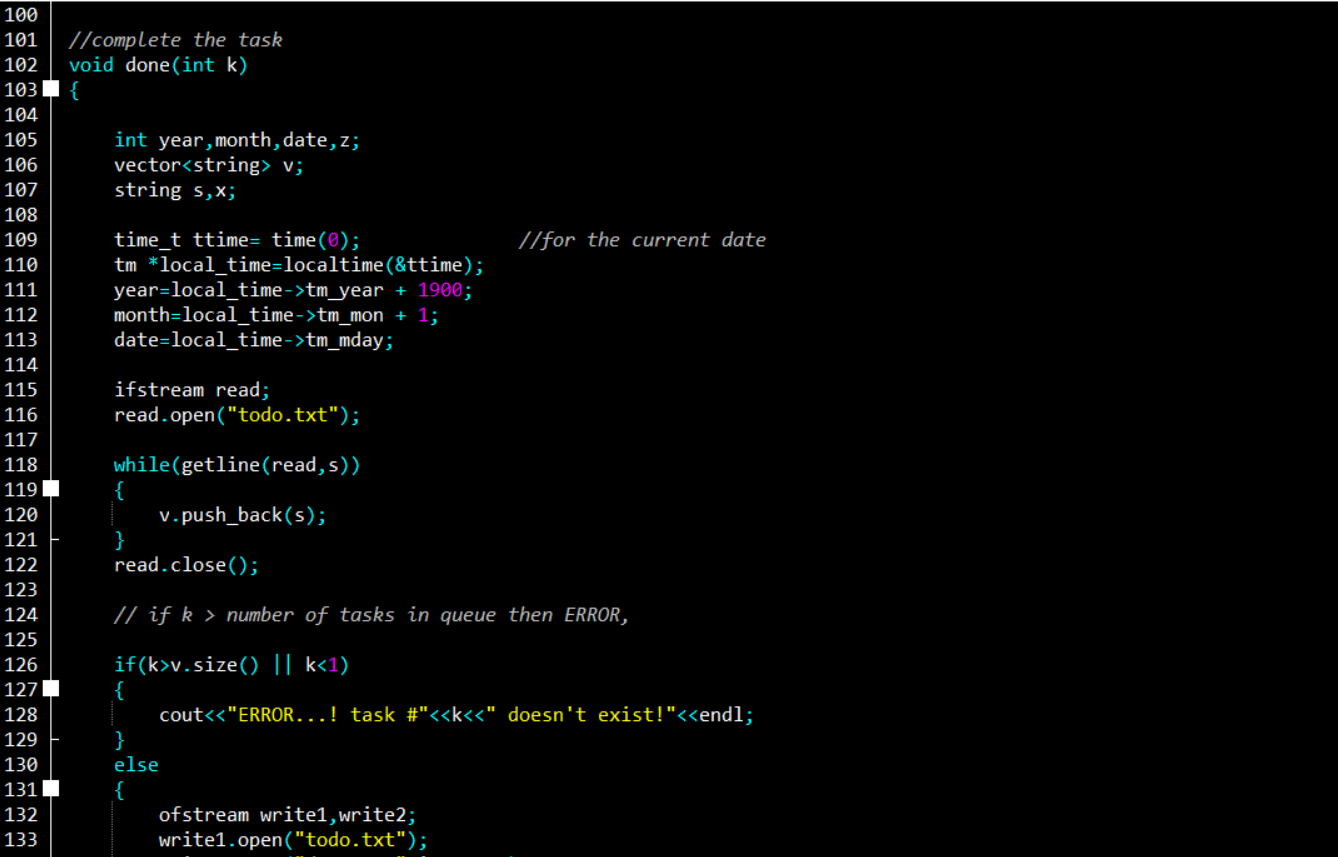
report(): displays the total number of pending tasks and completed tasks.

## Codes Used in the System:

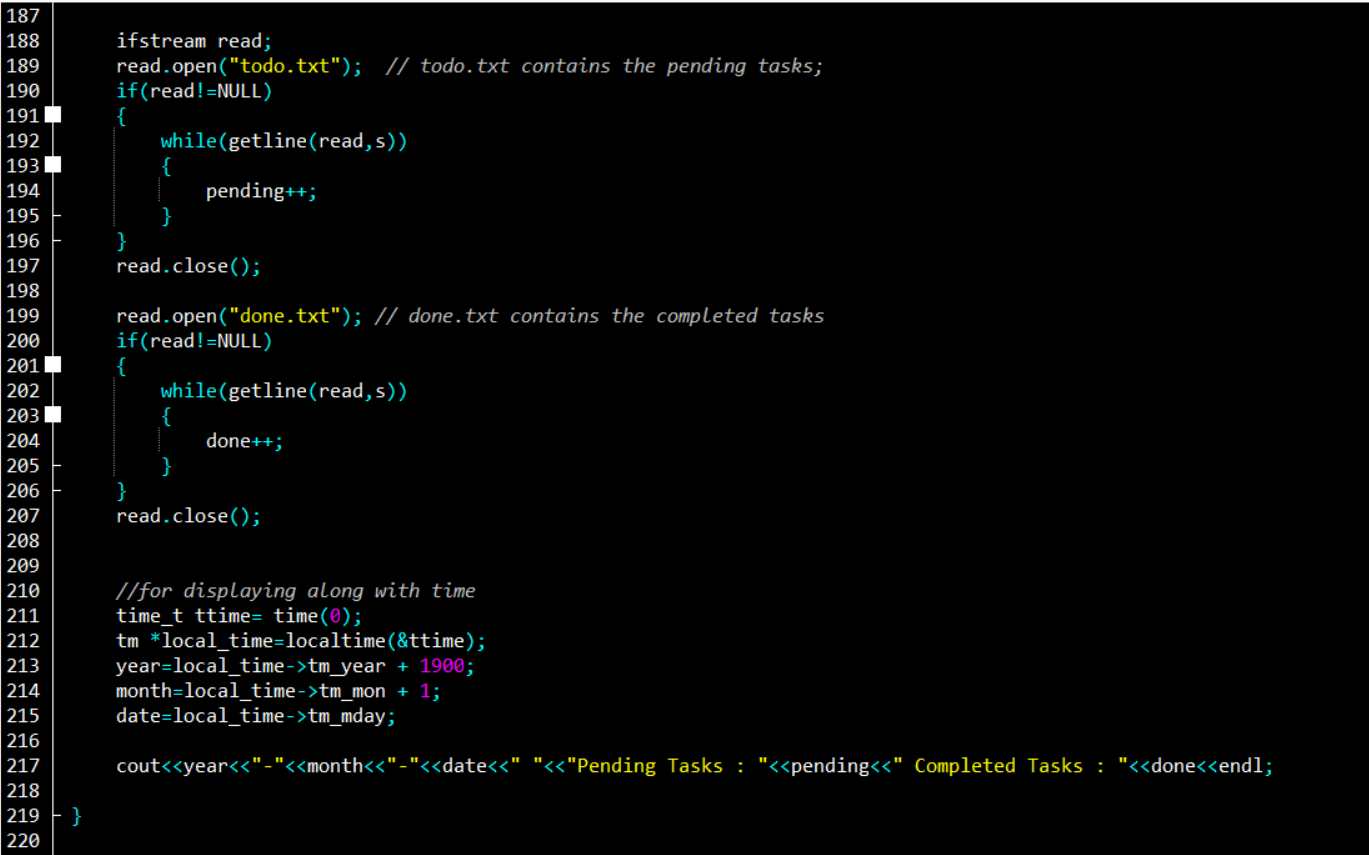
## 

Figure 2 screenshot1

# 



# 



# 



# 

## IO Screen

## 

# CONCLUSION

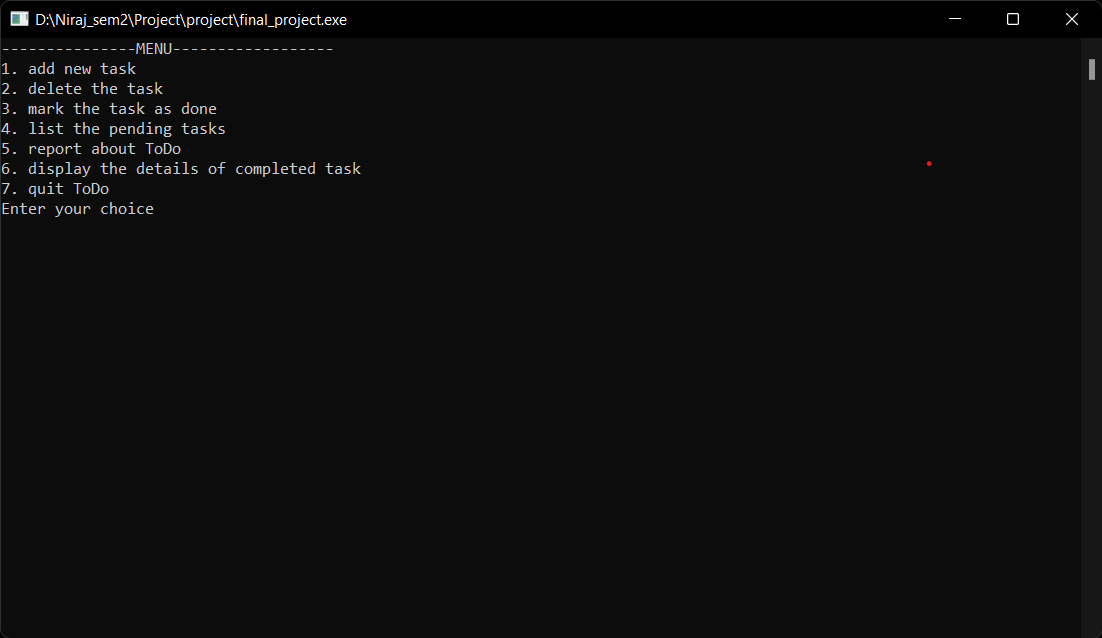
In conclusion, the C++ to-do list project was successfully developed to provide an efficient and user-friendly tool for managing tasks. The project fulfilled its objectives of creating a simple and effective way to manage tasks, while also being lightweight and easily navigable. The project successfully used file handling to store and manage data, rather than using complex database systems.

During the development process, several challenges were encountered, such as designing an effective user interface and ensuring the functionality of the application. These were overcome by utilizing various C++ libraries and support from the supervisor.

Overall, the project was successful in its mission to create a functional and efficient to-do list application using C++. Future work can include the integration of SQL databases and development of more advanced features. With its potential to simplify task management and increase productivity, the C++ to-do list application can be a valuable tool for individuals and teams.

# USER MANUAL

The user manual includes the interface that user mostly interacts with the output and the first output user gets to interacts given below:



Here, the user can add the tasks, delete the task, label the task as done, list the pending tasks and view the number of pending and completed task. User simply has to enter the label of the task they want to perform and perform the desired actions.

# REFERENCES

|  |  |
| --- | --- |
| [1] | "Task Warrior," [Online]. Available: https://taskwarrior.org/. |