



A
PROJECT REPORT ON

“CREATE A SIMPLE TEXT EDITOR”

By

Sr. No.	NAME	ROLL NO.
1	Omkar Mahaling Vandare	32487

GUIDE

MR. N.S SHIRUDE

DEPARTMENT OF
ELECTRONICS AND TELECOMMUNICATION ENGINEERING
PUNE INSTITUTE OF COMPUTER TECHNOLOGY
PUNE – 43

A.Y. 2023-24

INDEX

Sr. No.	Contents	Page No.
1	Problem Statement	1
2	Objectives	1
3	Introduction	2
4	Flowchart and Code Link	6
5	Result	7
6	Conclusion	8
7	Applications	8
8	Future scope	9
9	Copy Right Affirmation	9

1. PROBLEM STATEMENT:

Design and implement a simple text editor application using Java Swing, which allows users to perform basic text editing operations such as creating, opening, editing, saving, and closing text files. The objective is to develop an intuitive and user-friendly interface for editing text documents.

2. OBJECTIVE :

- 1) Develop a Functional Text Editor: The primary objective is to create a text editor application that fulfills the basic requirements of text editing, such as creating, opening, editing, saving, and closing text files. The editor should provide a user-friendly interface for performing these operations efficiently.
- 2) Implement User Interface: Design and implement a clean and intuitive user interface using Java Swing components. The interface should be easy to navigate and provide all necessary functionalities through menus, buttons, and other GUI elements.
- 3) Support Basic Text Editing Features: Ensure that the text editor supports essential text editing features such as cut, copy, paste, undo, redo, selecting font styles, sizes, and colors. These functionalities are fundamental for users to manipulate text effectively.
- 4) Enable File Operations: Implement functionalities for opening, saving, and closing text files in various formats. Users should be able to work with existing text files or create new ones, with options for choosing file formats and encodings.

3. INTRODUCTION:

3.1 Background/context :

Existing Text Editors: While there are many text editors available, ranging from simple ones like Notepad to complex Integrated Development Environments (IDEs) like IntelliJ IDEA or Visual Studio Code, there's often a need for a lightweight and straightforward editor for quick text manipulation tasks.

Java Swing: Java Swing is a GUI toolkit for Java, providing a set of components that allow developers to create graphical user interfaces for Java applications. It's widely used for creating desktop applications due to its platform independence and ease of use.

Educational Purposes: Developing a text editor using Java Swing serves as an educational project for learning GUI programming concepts, file I/O operations, event handling, and basic software engineering principles. It provides hands-on experience in building desktop applications with Java.

Customization and Control: By developing a custom text editor, developers have full control over the features, functionality, and user interface design. This allows for customization tailored to specific needs and preferences, which might not be achievable with off-the-shelf solutions.

3.2 Relevance:

User Expectations: Users expect a text editor to provide essential editing functionalities such as cut, copy, paste, undo, and redo. These features are standard across most text editing applications, and users rely on them for efficient text manipulation.

Usability and Productivity: Basic text editing features enhance usability and productivity by allowing users to perform common editing tasks quickly and intuitively. Users should be able to manipulate text seamlessly without having to navigate complex menus or commands.

User Experience: The availability of basic editing features contributes to a positive user experience. Users are more likely to find the text editor intuitive and easy to use if it provides familiar and expected functionalities.

Versatility: Supporting basic text editing features ensures that the text editor is versatile and can accommodate various text manipulation needs. Whether users are writing documents, editing code, or taking notes, they require basic editing tools to work efficiently.

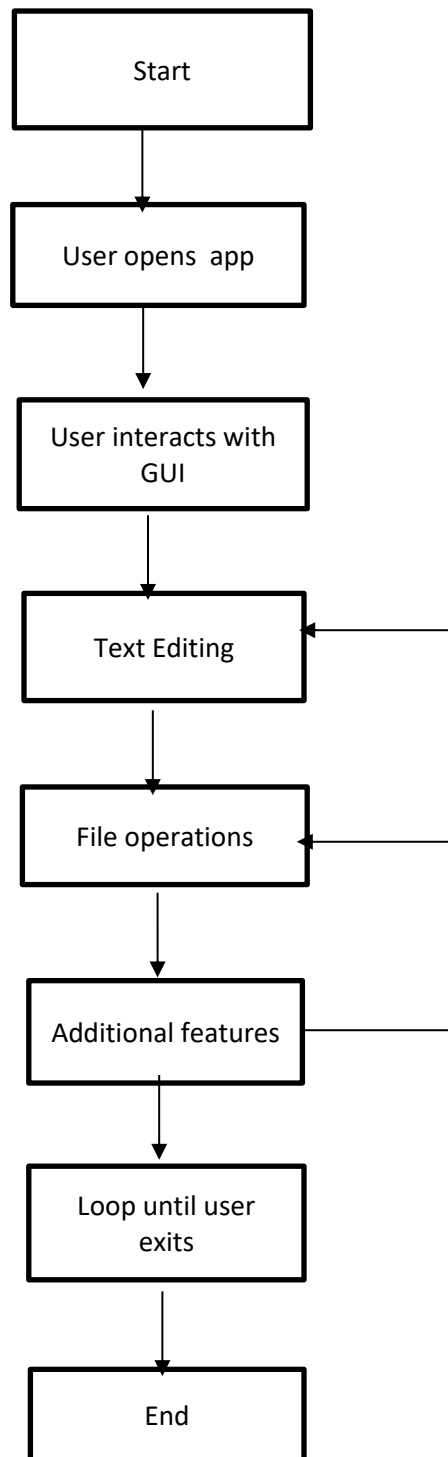
3.3 Project details:

The project "Simple Text Editor using Java Swing" aims to create a straightforward yet versatile text editing tool using Java Swing. The primary objective is to develop an application that enables users to perform common text editing tasks efficiently and intuitively. This entails implementing essential features such as cut, copy, paste, undo, and redo operations, as well as supporting file operations like opening, saving, and closing text files in various formats. The user interface will be designed to be clean and user-friendly, featuring menus, buttons, and other GUI components for easy navigation. Additionally, the editor will provide options for users to customize font styles, sizes, and colors, enhancing the editing experience. Error handling mechanisms will be implemented to manage exceptions gracefully, ensuring a robust and reliable application. Cross-platform compatibility will be prioritized to ensure accessibility across different operating systems. Throughout the project, thorough testing will be conducted to validate the functionality and usability of the text editor, with comprehensive documentation prepared to guide users on installation, usage, and troubleshooting.

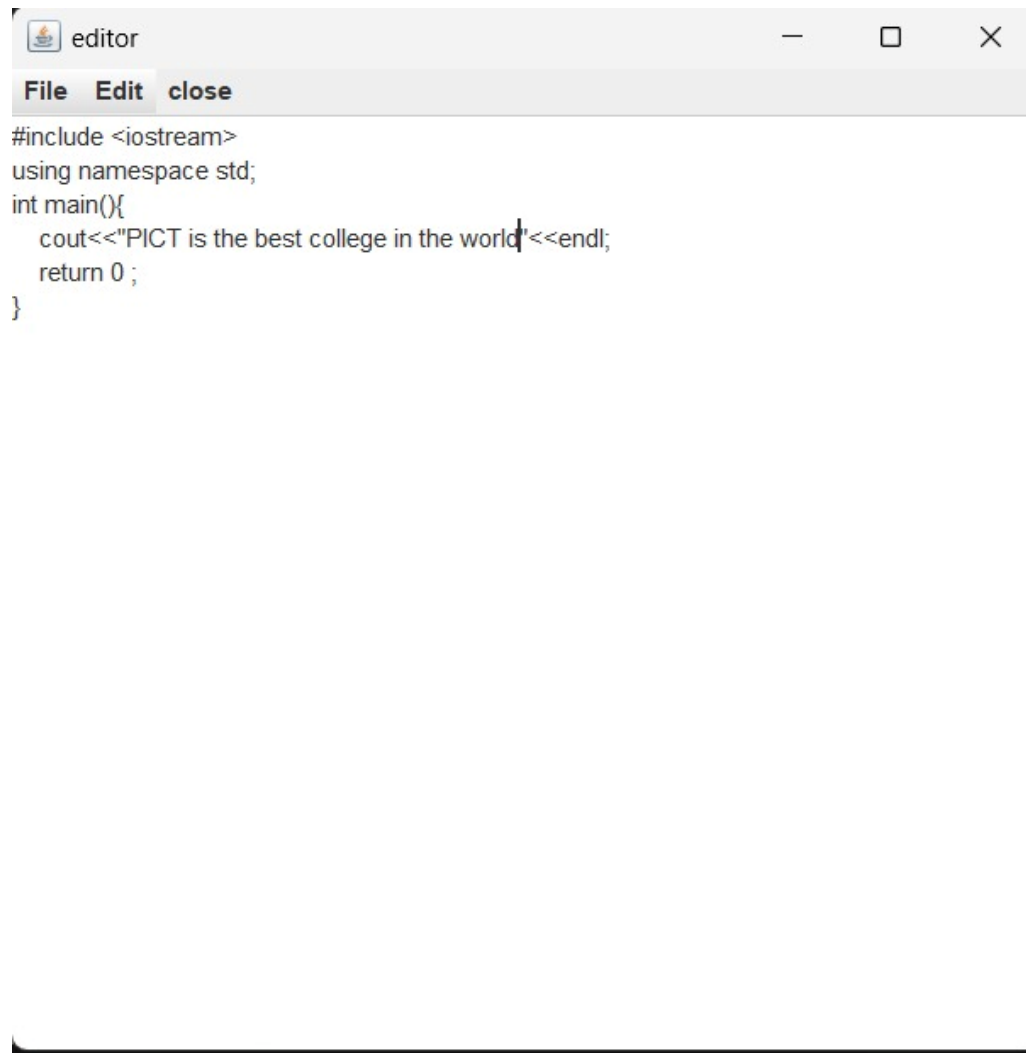
3.4 Scope:

The scope of the "Simple Text Editor using Java Swing" project revolves around the creation of a Java-based text editing application that caters to fundamental text manipulation needs. This encompasses developing a user interface with intuitive controls and functionalities, including basic editing options like cut, copy, paste, undo, and redo. Additionally, the application will facilitate file management operations such as opening, saving, and closing text files in different formats. While the primary focus lies in providing essential editing features and seamless file handling capabilities, the project will maintain simplicity and efficiency as core principles, avoiding unnecessary complexities that could hinder usability. The scope also involves ensuring cross-platform compatibility to enable users across various operating systems to utilize the application seamlessly. Throughout the development process, adherence to the defined scope will be crucial to delivering a lightweight, user-friendly text editor that meets the basic text editing requirements effectively.

4.Flow Chart and SOURCE CODE:



5.RESULT:



The image shows a screenshot of a C++ program being executed in a window titled "editor". The window has a menu bar with "File", "Edit", and "close" options. The code in the editor is as follows:

```
#include <iostream>
using namespace std;
int main(){
    cout<<"PICT is the best college in the world"<<endl;
    return 0 ;
}
```

The program has been executed, and the output "PICT is the best college in the world" has been printed to the console. The cursor is positioned at the end of the output line.

6. CONCLUSION:

"Simple Text Editor using Java Swing" project aims to develop a lightweight yet functional text editing application tailored for basic text manipulation tasks. By leveraging Java Swing, the project provides a platform-independent solution with a clean and intuitive user interface. Through essential features such as text editing operations, file management functionalities, and additional tools like search and replace, the application offers users a seamless and efficient text editing experience. The project emphasizes simplicity, usability, and reliability, ensuring that users can perform common text editing tasks with ease. Overall, the project serves as a valuable tool for users seeking a straightforward text editor solution while also providing a practical learning experience in Java GUI programming and software development.

7.APPLICATIONS:

- 1) Education: It can be used as a teaching tool in computer science courses to demonstrate GUI programming concepts, file I/O operations, and software development principles using Java Swing.
- 2) Software Development: Developers can use the project as a foundation to build more complex text editing applications or integrate it into larger software projects that require basic text editing capabilities.
- 3) Productivity Tool: Individuals can utilize the text editor for everyday tasks such as writing documents, editing code snippets, taking notes, or formatting text.
- 4) Learning and Practice: Aspiring programmers can use the project to practice Java programming skills, improve their understanding of GUI development, and gain hands-on experience in software development.
- 5) Customization: Users can customize the text editor to suit their specific needs by adding new features, modifying the user interface, or extending its functionality.

8. FUTURE SCOPE:

Future scope for the project includes enhancing text editor functionality with features such as syntax highlighting, code completion, and version control integration. Additionally, implementing advanced search capabilities, support for multiple tabs, and customizable themes can further improve user experience. Integration with cloud services for seamless file synchronization and collaboration would extend the application's utility. Enhancing performance optimization and expanding platform compatibility to mobile devices could broaden its reach. Lastly, incorporating accessibility features and localization support can make the text editor more inclusive and adaptable to diverse user needs.

9. COPY RIGHT AFFIRMATION:

We undersigned pledge and represent that the source code printed in this project report does not violate any proprietary or personal rights of others (including, without limitation, any copyrights or privacy rights); that the Work is factually accurate and contains no matter libelous or otherwise unlawful; that we have substantially participated in the creation of the Work and that it represents our original work sufficient for us to claim authorship.

Name of students

Sign

1. Omkar Vandare