NEW HORIZON COLLEGE OF ENGINEERING

IEEE Advancing Technology for Humanity

International Conference on Multidisciplinary Research in Technology and Management - MRTM 23

Paper Id :1244

PyEdit Pro: The Ultimate Advanced Text Editor in Python - Empowering Text Editing Experience

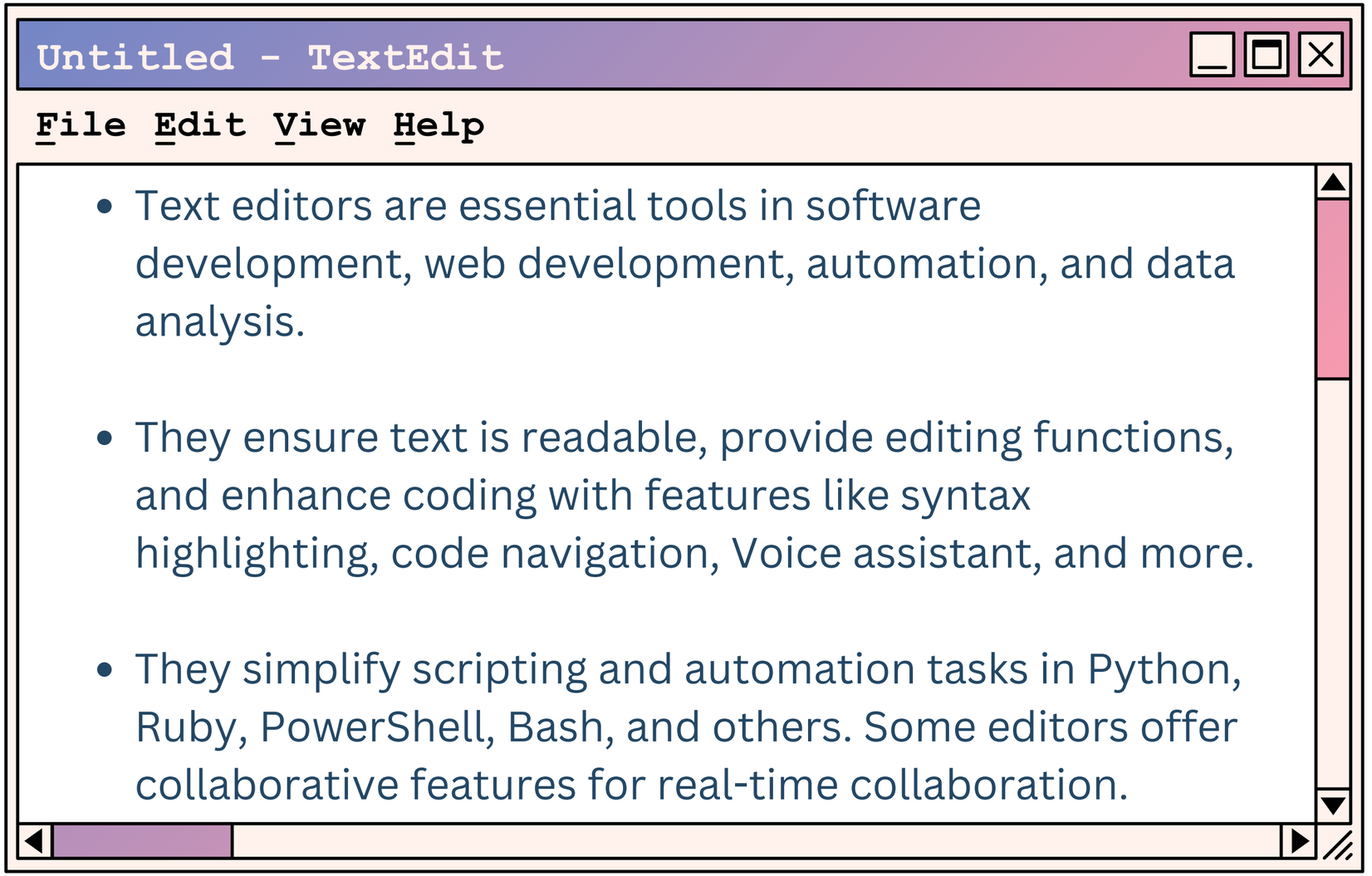
Presented
by
Rohan C
Co-Authors: Rahul Roshan G, Rohit Roshan, S M Sutharsan Raj and Dr. Sapna Vikram Mewundi, PES University.

22nd & 23rd September 2023

PES UNIVERSITY

Contents

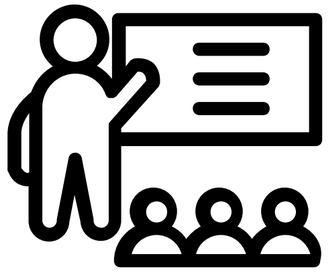
INTRODUCTION



CHALLENGE(S)

The challenges addressed by our advanced text editor were:

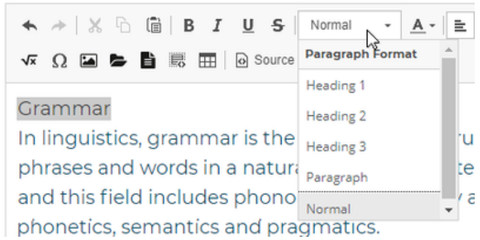
- **Visually Impaired Users Using Normal Text Editors**
- **Student Comprehension with Text-Only Slides**
- **System hardware compatibility - Efficient usage of hardware resources**



OBJECTIVES

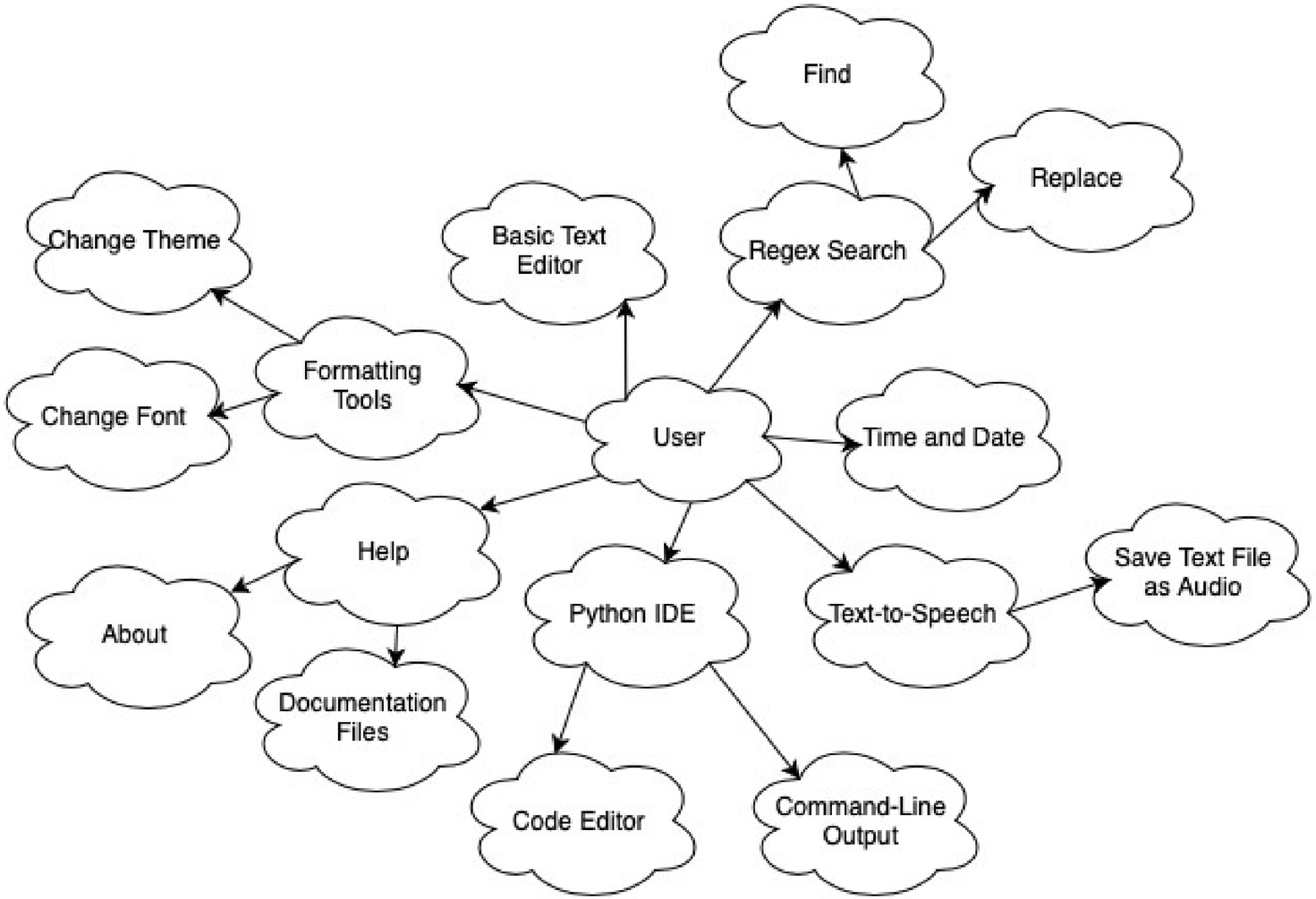
The main objectives of our paper are as follows:

- **Creation of an innovative text editor**
- **Exploration of Design Principles and Features:**
- **Insights for Python Developers**
- **Addressing Accessibility Challenges**



Text editor

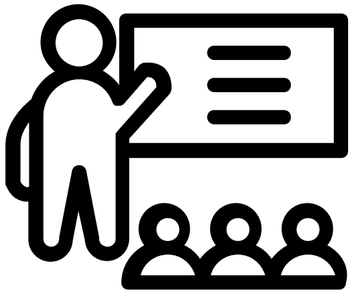
PROPOSED SYSTEM



IMPLEMENTATION DETAILS

We have used tkinter, win32api, pyttsx3, playsound, gtts, and subprocess libraries for creating this text editor.

- **Tkinter** to design the widgets.
- **win32api** to execute a "print" action on the file specified by **filename** using the default printer configured on the Windows system.



- **Pyttsx3** allows users to listen to the saved text through **audio playback.**
- **Playsound** library to provide visually impaired users with **audio-guided** instructions for using the application.

IMPLEMENTATION DETAILS

- **Gtts** library to **convert text into audio** and enable users to save the resulting audio file on their local device.

- **Subprocess** library used to **constructs** a command to run a Python script, executes the command, captures the script's output and error.

So, this text editor is effective and feature-rich text editor that provides comprehensive audio guidance for visually impaired users and students, covering both basic and advanced functionalities.

RESULT & ANALYSIS

- High features with good **GUI tkinter** python.
- **Code editor** with
  - **Saving** option
  - **Opening** a new file
- Text editor enables the creation of **fully-fledged programs**.
- It is capable of carrying out all the standard functions of a text editor as well as **sophisticated compiler operations**.
- Text editor with **sum, average and count** of number in text editor and time-date respectively.
- The following Table shows the sample execution time in both serial and parallel way, and other features.:

TABLE II: SERIAL EXECUTION TIME TO OPEN THE APPLICATION

	2 cores	4 cores	4 cores	8cores
Minimum	2.0996 s	0.4028 s	0.1121s	0.1853s
Average	2.5627 s	0.4150 s	0.1188s	0.4351s

TABLE III: PARALLEL EXECUTION TIME TO OPEN THE APPLICATION

	2 cores	4 cores	4cores	8cores
Minimum	7.42 s	7.90s	6.11s	6.26s
Average	7.48 s	7.93s	6.16s	6.34s

TABLE IV: SAVING/DOWNLOADING AUDIO FILE.

97 words	9.0719 seconds
194 words	14.8728 seconds
388 words	29.229 seconds

TABLE V: TESTING BUILT-IN CODE EDITOR COMPILING TIME

one line "print" code	0.195 seconds
one for loop "print" code	0.182 seconds
two for loop "print" code	0.179 seconds
three for loop "print" code	0.165 seconds

Summary/References

- **Versatile Tool:** Text editors are versatile tools with a significant impact on various fields across. They are also beneficial for visually impaired individuals.
- **PyEdit Pro Standout:** PyEdit Pro stands out as a text editor. It offers an array of advanced features and a user-friendly interface.
- **Rich Toolset and Extendibility:** Rich toolset and the ability to extend its functionality, enhancing its appeal to developers seeking customization and efficiency.
- **Efficient Resource Management:** Efficient memory management and resource optimization, further contributing to its popularity among developers.

In essence, PyEdit Pro, play a crucial role in various domains, offer advanced features, customization options, and efficient resource management, making them essential tools for professionals and visually impaired users alike.

Summary/References

- “R, Balaji & Babu, M.. (2011). **Multimodal Input Interface Using Multicore**. International Journal of Computer Information System. 2. 67-71.”

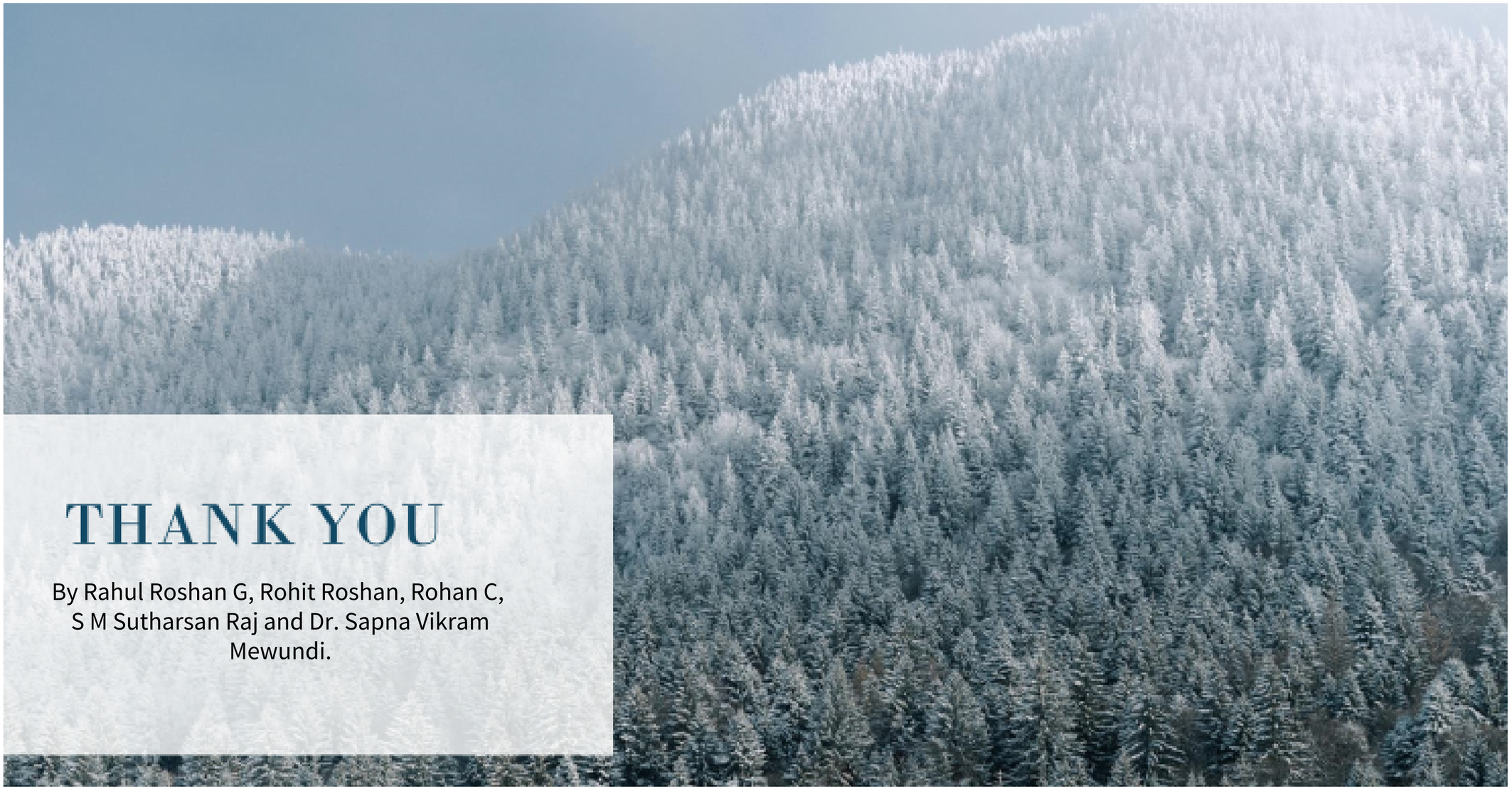
- “E. Altuncu, B. K. Bilgehan, Y. S. Kartal, S. Kızılgüneş, M. S. Nikoo and H. Oğuztüzün, "**Blocks and text integration in a language-based editor for a domain-specific language**," 2017 International Conference on Computer Science and Engineering (UBMK), Antalya, Turkey, 2017, pp. 1084-1089, doi: 10.1109/UBMK.2017.8093484.”

- “Christopher W. Fraser. 1980. **A generalized text editor**. Commun. ACM 23, 3 (March 1980), 154– 158.” https://doi.org/10.1145/358826.358834

- “S. Shrivastav and H. Attri, "**Terminal Based Text Editor**," 2022 3rd International Conference for Emerging Technology (INCET), Belgaum, India, 2022, pp. 1-4, doi: 10.1109/INCET54531.2022.9824597.”

- Joao Ribeiro Bezerra, Lu´ıs Fabr´ıcio Wanderley Goes & Wladmir Cardoso Brandao (2022). NEWRITER: **A Text Editor for Boosting Scientific Paper Writing.**

- Jiseong Gu and Geehyuk Lee. 2019. “**In-Place Ink Text Editor: A Notepad-like Simple Text Editor with Direct Writing**.” In Proceedings of the 2019 ACM International Conference on Interactive Surfaces and Spaces (ISS '19). Association for Computing Machinery, New York, NY, USA, 327– 329. https://doi.org/10.1145/3343055.3360744



THANK YOU

By Rahul Roshan G, Rohit Roshan, Rohan C, S M Sutharsan Raj and Dr. Sapna Vikram Mewundi.