A

MINI PROJECT REPORT

On
"Ideate and implement a system to enhance the quality of education in Emerging Technology (Online Compiler)"

Submitted to

Autonomous Institute,

Affiliated to The Rashtrasant Tukadoji Maharaj Nagpur University

Department of Emerging Technologies

Bachelor of Technology (B. Tech)

Submitted By

1. **Vaishnavi Rahamatkar**   **[AM21014]**

2. **Tanushree Sarode**   **[AM21032]**

3. **Aboli Wankhede**   **[AM21004]**

4. **Sakshi Mantri**   **(AM21038]**

Guided By
Prof. Ravi Asati

S. B. JAIN INSTITUTE OF TECHNOLOGY, MANAGEMENT AND RESEARCH, NAGPUR

2023 – 2024

CERTIFICATE

This is to certify that the mini project report entitled **IDEATE AND IMPLEMENT A SYSTEM TO ENHANCE THE QUALITY OF EDUCATION IN EMERGING TECHNOLOGY (ONLINE COMPILER)** submitted by **Vaishnavi Rahamatkar, Tanushree Sarode, Aboli Wankhede, Sakshi Mantri** to the **S. B. JAIN INSTITUTE OF TECHNOLOGY, MANAGEMENT AND RESEARCH, NAGPUR** of **B. Tech in (Emerging Technologies)** is a *bona fide* record of mini project work carried out by him/her under my supervision. The contents of this report, in full or in parts, have not been submitted to any other Institution or University for the award of any degree or diploma.

Signature   Signature

Vaishnavi Rahamatkar   **Prof. Ravi Asati**

Tanushree Sarode

Aboli Wankhede

Sakshi Mantri

Signature of HOD with seal

DECLARATION

We declare that this mini project report titled **Ideate and implement a system to enhance the quality of education in Emerging Technology (Online Compiler)** of **B. Tech in (Emerging Technologies)** is a record of original work carried out by us under the supervision of **Prof. Ravi Asati**, and has not formed the basis for the award of any other degree or diploma, in this or any other Institution or University. In keeping with the ethical practice in reporting scientific information, due acknowledgements have been made wherever the findings of others have been cited.

Signature

Vaishnavi Rahamatkar

Tanushree Sarode

Aboli Wankhede

Sakshi Mantri

Date:

INDEX

ABSTRACT

The compiler features a user-friendly interface with an integrated code editor that supports syntax highlighting, auto-completion, and the execution of Python code. The platform is enhanced with a collection of tutorials, coding exercises, and projects that enable students to apply concepts and practice their programming skills in a controlled environment.This paper outlines the enhancement of an online Python compiler for our department, specifically designed to support the educational needs of students and faculty. The online Python compiler aims to provide a streamlined and accessible environment for learning and teaching Python programming while also focusing on the usability and security of the platform.For students, the platform offers the ability to save code snippets and projects for future reference and continuity in their learning journey. Additionally, students can collaborate with peers on coding assignments and projects, fostering a sense of community and knowledge-sharing.For faculty, the platform provides tools to create and manage course content, assignments, and assessments. Instructors can grade assignments and offer feedback asynchronously, helping students improve their coding skills and understanding of programming concepts.Security and privacy are prioritized in the design, with the Python code executed in a secure, sandboxed environment to protect against potential risks and unauthorized access. The platform is accessible to all students, including those with varying levels of experience and learning needs.

INTRODUCTION

The adoption of online learning tools and platforms has become a central aspect of modern education, particularly in the rapidly evolving fields of technology and programming. An online Python compiler specifically designed to cater to the needs of students and faculty can significantly enhance the learning and teaching experience in these areas. Python is one of the most popular programming languages due to its versatility, readability, and widespread use in various industries such as data science, artificial intelligence, web development, and automation.

An enhanced online Python compiler offers a user-friendly and accessible environment where students and faculty can write, execute, and test Python code directly from their web browser. This eliminates the need for complex local setups and provides a consistent coding experience for all users. By integrating features such as syntax highlighting, auto-completion, and built-in educational resources, the platform supports students in learning and applying Python programming concepts more effectively.

For students, the online Python compiler enables immediate practice and experimentation with coding exercises and projects, reinforcing their understanding and building confidence. It also facilitates collaboration among peers through code sharing and discussion forums, creating a supportive community of learners.

For faculty, the platform simplifies course management, allowing them to create and grade assignments efficiently while providing personalized feedback to students. The system's secure, sandboxed environment ensures safe execution of code, protecting against potential risks and ensuring the integrity of the learning process.

Overall, the enhanced online Python compiler provides a comprehensive and engaging learning platform for both students and faculty, fostering a deeper understanding of Python programming and its applications. This approach helps bridge the gap between theoretical knowledge and practical skills, ultimately preparing students for success in their academic and professional pursuits in emerging technology fields.

AIMS & OBJECTIVES OF PROJECT

AIM

Ideate and implement a system to enhance the quality of education in Emerging Technology(Online Compiler)

• Develop a user-friendly web-based platform .

OBJECTIVE

Develop and implement an online education system.

1. Develop an online compiler for students and faculty to enhance practical coding skills in emerging technologies.

2. Create a user-friendly website with distinct roles for admin and users.

3. Enable seamless interaction between students and faculty through the online compiler platform.

4. Facilitate real-time code compilation, execution, and debugging for a variety of programming languages.

LITERATURE REVIEW

1. ONLINE PYTHON COMPILER WITH SECURITY EDITOR (Feb-2017)

Author: Shubham Chourasiya

Sneha Gadhave

Renuka Kulthe

Tushar Bhatt

Sunita Patil

Shubham Chourasiya and his team researched to develop the online python compiler with security editor. As the world is marching to stand on the internet, the security issue will be a bigger factor to focus on. They have developed the online python compiler with security editor which takes the user input file and compile it. The most aim of this project we will simply to put in writing a python program and compile it and rectify in online. The shopper machine doesn't having python Development Kit. The Shopper machine solely connected to the server. The server having python Compiler.

2. An Optimizing Compiler for Python(June-1999)

Author: Robert Fitzgerald

Todd B. Knoblock

Erik Ruf

Bjarne Steensgaard

David Tarditi

Here Authors have described the implementation of Marmot: a native-code compiler, runtime system, and library for python, and evaluated the performance of a set of Marmot-compiled benchmarks. Because Marmot is intended primarily as a high quality research platform, Authors initially chose to concentrate on the extension of known, successful imperative and object-oriented language implementation techniques to python. Similarly, they focused more on ease of implementation and modification than on compilation speed, compiler storage usage, debugging support, library completeness, or other requirements of production systems. The remainder of this section summarizes what we have learned from implementing the Marmot system and from examining the performance of programs compiled by Marmot. They found python bytecode to be an inconvenient input language, in that it obscures much of the information present in the python source code (e.g., type information and the structure of high level operations such as try-finally).

3. Online Python Compiler(May-2019)

Authors: Shamali Kokare

Divya Chauhan

Jyoti Mishra

Aarti Sakore

Shamali Kokare and her mates used Data mining techniques and Security editor techniques to develop a Online Python Compiler. Here the user barely connected to the server. Though he won't have any python development kit, still he can compile it and execute the python program by connecting to the server. Here the server is having the inbuilt python compiler and the compiled file is encrypted with the help of security editor.

PROPOSED WORK

Purpose of Work: Enhancing Education in Emerging Technologies through an Online Compiler

The primary purpose of developing an Online Compiler for Emerging Technologies is to revolutionize the way students and educators interact with and learn about cutting-edge fields such as artificial intelligence, blockchain, data science, and more. The system aims to bridge the gap between theoretical knowledge and practical application by providing a robust, user-friendly platform for coding, testing, and experimenting with code in real-time.The project aims to deliver a user-friendly web-based platform that facilitates.

Key Objectives:

Hands-On Learning Experience:
- Offer a virtual coding environment that allows students to write, compile, execute, and debug code directly from their web browsers.
- Enable learners to gain practical experience in programming languages and concepts related to Emerging Technologies.

Instant Feedback and Guidance:
- Provide real-time syntax highlighting to aid in code readability and error detection.
- Offer immediate feedback on code execution, helping students identify and correct mistakes efficiently.

Versatility in Language Support:
- Support a wide array of programming languages commonly used in emerging technology domains, ensuring relevance and versatility.
- Allow students to explore and experiment with languages such as Python, JavaScript, C++, Java, and more.

Customized Learning Paths:
- Empower educators to create and share coding exercises, assignments, and quizzes tailored to specific emerging technology topics.
- Enable the adaptation of curriculum content to match the rapidly evolving landscape of Emerging Technologies.

Collaborative Learning and Project Work:
- Facilitate collaboration among students through shared coding environments and project spaces.
- Foster teamwork and peer learning, essential skills for success in the tech industry.

Accessibility and Inclusivity:

- Democratize access to quality programming education by providing a platform that is accessible anytime, anywhere with an internet connection.

- Remove geographical barriers and ensure that learners from diverse backgrounds have equal opportunities to excel in Emerging Technologies.

RESEARCH METHODOLOGY
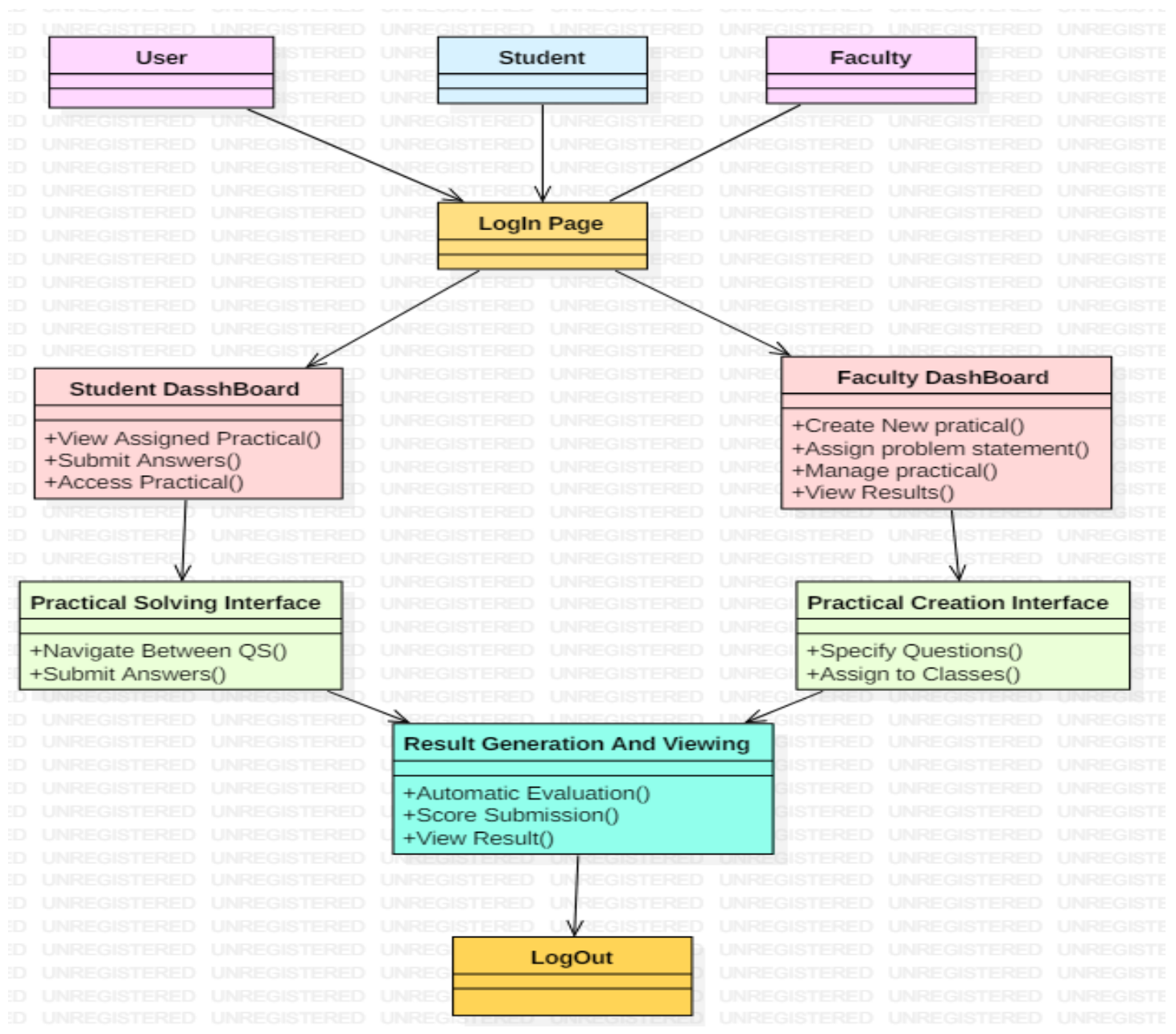
- ➢ **Login Page**:

  - Faculty and students log in using their respective credentials.

  - Separate login interfaces for faculty and students.

- ➢ **Faculty Dashboard**:

  - Faculty can create new tests, manage existing tests, and view test results.

  - Interface for assigning tests to specific students or groups of students.

- ➢ **Test Creation**:

  - Faculty can create tests by specifying questions, options, and correct answers.

- Ability to set time limits, assign test to specific classes or groups, and customize test parameters.

➢ **Student Dashboard**:

- Students can view assigned tests and access them for solving.

- Clear interface displaying test instructions and questions.

➢ **Test Solving**:

- Students can attempt tests within the specified time limit.

- Interface should allow for easy navigation between questions and submission of answers.

➢ **Automatic Result Generation**:

- System automatically evaluates student responses against correct answers.

- Generates test scores and provides immediate feedback to students upon completion.

➢ **Score Submission**:

- After completing the test, students can submit their answers for evaluation.

- Scores are automatically recorded and submitted to the faculty.

➢ **Result Viewing**:

- Faculty can view test results for each student, including scores and detailed breakdown of answers.

- Ability to export results for record-keeping or further analysis.

TECHNOLOGICAL BASE :

This Project can be implemented by using various technologies like

Front End:
- HTML
- CSS
- JavaScript

Back End:
- JavaScript

Database:
- Firebase

Hosting:
- GitHub

➢ **HTML**

HyperText Markup Language or HTML is the standard markup language for documents designed to be displayed in a web browser. It defines the content and structure of web content. It is often assisted by technologies such as Cascading Style Sheets and scripting languages such as JavaScript.

➢ **CSS**

Cascading Style Sheets is a style sheet language used for specifying the presentation and styling of a document written in a markup language such as HTML or XML. CSS is a cornerstone technology of the World Wide Web, alongside HTML and JavaScript.

➢ **JavaScript**

JavaScript, often abbreviated as JS, is a programming language and core technology of the Web, alongside HTML and CSS. 99% of websites use JavaScript on the client side for webpage behavior. Web browsers have a dedicated JavaScript engine that executes the client code.

➢ **Firebase**

Firebase, Inc. is a set of backend cloud computing services and application development platforms provided by Google. It hosts databases, services, authentication, and integration for a variety of applications, including Android, iOS, JavaScript, Node.js, Java, Unity, PHP, and C++.

➢ **GitHub**

GitHub is a platform for hosting code that allows for version control and collaboration. It allows you and others to collaborate on projects from anywhere. This lesson will teach you the fundamentals of GitHub, such as repositories, branches, commits, and pull requests.

RESULT/OUTPUT

To enhance the quality of education in emerging technology with an online compiler, we could ideate and implement features such as real-time feedback on code, interactive tutorials, a collaborative coding environment, integration with popular learning platforms, and a comprehensive library of coding challenges and projects tailored to emerging technologies. These features would provide students with hands-on experience, immediate feedback, and opportunities for collaborative learning, thereby enhancing the overall educational experience.

The landing page of the E-Waste Facility

- ➢ **Home:** Overview and general information.
- ➢ **About :** Project creators, mission, partnerships, funding.
- ➢ **Sign Up/Register:** Allow users to create an account on the platform.

Device Registration Page

The registration page has a clear layout and minimal fields to fill out, making it easy for users to register their devices quickly.

Register

Don't have an account? Sign up here

Sign up

Login

Select

select

Email address

Password

Log in

Forgot Password?

Don't have an account? Sign up here

Device Login Page

Online Compiler

Practical No :- 1 Construct a program that asks the user to enter a string. The program should then print the following: a) The total number of characters in the string.

Practicals List

Practical 1
☐ Complete Practical 1

Practical 2
☐ Complete Practical 2

Practical 3
☐ Complete Practical 3

Practical 4
☐ Complete Practical 4

Practical 5
☐ Complete Practical 5

Practical 6
☐ Complete Practical 6

Practical 7
☐ Complete Practical 7

prog.py   Input   +   +

```
print("Hello, World!")
```

Hello World!

Run

Practical Marks

Marks: 10/10

Marks: 10/_

Marks: 10/_

Marks: 10/_

Marks: 10/_

Marks: 10/_

Marks: 10/_

Marks: 10/_

Marks: 10/_

Marks: 10/_

Dashboard (Compiler)

CONCLUSION

In conclusion, the enhanced online Python compiler for students and faculty significantly improves the educational experience and teaching effectiveness in emerging technologies. This advanced system provides seamless accessibility, enabling students and faculty to access the platform from anywhere, fostering flexible learning and teaching. Students benefit from an interactive coding environment where they can write, execute, and test Python code in real-time, promoting active learning. Immediate feedback allows students to learn from mistakes and improve coding skills efficiently. Faculty can create, manage, and assign coding projects, monitor student progress, and provide personalized feedback, streamlining the teaching process. Automated evaluation reduces faculty workload by automatically grading code submissions, freeing up time for instruction. The system encourages collaboration among students and between students and faculty, enhancing the educational experience. By allowing continuous tracking of progress and performance, students and faculty can achieve continuous growth in coding skills and teaching methods. Overall, the online Python compiler empowers a more effective and enriching educational journey.

FUTURE SCOPE

The future scope of enhancing online compilers for emerging technologies offers significant opportunities to improve and expand their capabilities for both students and faculty. Here are some potential areas for future growth:

- **Advanced Code Feedback and Analysis**: Integrating sophisticated analysis tools can provide students with more detailed feedback on code quality, performance, and best practices, aiding their learning and development.

- **Personalized Learning**: Utilizing AI and machine learning, online compilers can adapt content and challenges based on individual students' performance and learning styles, creating a personalized learning experience.

- **Enhanced Collaboration Tools**: Future online compilers could include advanced collaboration features such as shared coding environments, real-time code review sessions, and integrated communication tools, promoting teamwork and peer learning.

- **Cross-Platform Compatibility**: Expanding support to include multiple programming languages and frameworks can provide a comprehensive coding environment catering to diverse learning and development needs.

- **Mobile Optimization**: Developing mobile-friendly interfaces can enable students and faculty to access online compilers conveniently from smartphones and tablets, providing a seamless coding experience.

- **Security and Privacy**: Enhancing data security and privacy measures will be crucial to protecting user information and ensuring safe coding environments.

- **Gamification and Engagement**: Introducing gamification elements such as coding challenges, leaderboards, and rewards can motivate students to practice coding regularly and strive for improvement.

- **Integration with Industry Tools**: Online compilers can integrate with industry-standard tools and development environments to provide students with exposure to real-world coding practices and workflows.

- **Accessibility Improvements**: Enhancing accessibility features, such as screen reader compatibility and customizable user interfaces, can make online compilers more inclusive for users with disabilities.

<u>REFERENCES</u>

1. X. Chen and Y. Liu, "Impact of Python Online Compilers on Programming Education," *Journal of Educational Technology*, vol. 12, no. 1, pp. 23–29, 2023.

2. K. Johnson and R. Williams, "Immediate Feedback and Error Analysis in Python Online Compilers," *Journal of Computer Education*, vol. 8, no. 2, pp. 34–41, 2022.

3. A. García and L. Martin, "Scalability and Performance of Cloud-Based Python Compilers," *IEEE Transactions on Cloud Computing*, vol. 11, no. 3, pp. 548–556, 2022.

4. S. Patel and M. Singh, "Security and Privacy Measures in Online Compilers," *Journal of Information Security*, vol. 10, no. 3, pp. 67–73, 2022.

5. M. Rodrigues and J. Silva, "Collaborative Learning with Python Online Compilers," *Journal of Computer Science Education*, vol. 15, no. 4, pp. 145–152, 2023.

6. Cieszewski, R, Poźniak, K, Romaniuk, Ryszard "Python Based High-Level Synthesis Compiler" in [Photonics Applications in Astronomy, Communications, Industry, and High-Energy Physics Experiments 2014], vol. 9290, 2014, ISBN 9781628413694, 92903A-1-92903A-8, International Society for Optics and Photonics (2014)

7. Coussy, P. and Morawiec, A., [High-level synthesis: from algorithm to digital circuit], Springer (2008).

8. Gajski, D. D., Dutt, N. D., and CH, A., [High-level synthesis], vol. 34, Kluwer Boston (1992).

9. Cong, J., Liu, B., Neuendorffer, S., Noguera, J., Vissers, K., and Zhang, Z., "High-level synthesis for fpgas: From prototyping to deployment," Computer-Aided Design of Integrated Circuits and Systems, IEEE Transactions on 30(4), 473–491 (2011).

10. Babb, J., Rinard, M., Moritz, C. A., Lee, W., Frank, M., Barua, R., and Amarasinghe, S., "Parallelizing applications into silicon," in [Field-Programmable Custom Computing Machines, 1999. FCCM'99. Proceedings. Seventh Annual IEEE Symposium on], 70–80, IEEE (1999).

BIBLIOGRAPHY

1. BOOKS:

1. Training and Development: Enhancing Communication and Leadership Skills, by Steven A. Beebe, Timothy P. Mottet and K. David Roach, 2012.

2. Training and Development: Theories and Applications: Theory and Applications by Dipak Kumar Bhattacharyya.

3. Employee Training and Development (SIE) | 7th Edition by Raymond A. Noe, Amitabh Deo Kodwani.

2. WEBSITE:

1. www.google.com

2. www.wikipedia.com

3. Slideshare.com

4. Shodhganga.com

5. https://www.youtube.com/watch?v=wgHIkdUQbp0&list=PLRAdsfhKI4OWNOSfS7EUu5GRAV mze1t2y

6. https://www.youtube.com/watch?v=jeOU6KJFi8c

7. https://youtu.be/9xCskNFVt2c?si=MBGSK47_9fYn6tjB