# CANDIDATE'S DECLARATION

We **Mili Srivastava (205433)** and **Rachit Bharadwaj (205445)** students of B.Tech. (Information Technology) at Uma Nath Singh Institute of Engineering and Technology, VBS Purvanchal University, Jaunpur, declare that the work presented in this Project Report titled **“Web 3.0 and Blockchain Based Decentralized Social Connectivity Web Application”** submitted to the **Department of Information Technology** for the award of Bachelor of Technology degree in Information Technology. All the work done in this project report is our own except for the quoted reference. To the best of our knowledge, this work has not been submitted to any other university or Institution for the award of any degree.

Date: **Student’s Name**

Place: UNSIET, Jaunpur Mili Srivastava (205433)

Rachit Bharadwaj (205445)

# CERTIFICATE

It is certified that, this project entitled “**Web 3.0 and Blockchain Based Decentralized Social Connectivity Web Application**”, submitted by Mili Srivastava and Rachit Bharadwaj in partial fulfillment of the requirement for the award of Bachelor of Technology in Information Technology degree from VBS Purvanchal University, Jaunpur, is a record of students own study carried under my supervision. This Project report has not been submitted to any other university or institution for the award of any degree.

**Project Guide**

Mr. Pravin Kumar Pandey (Assistant Professor)

Date:

Place: UNSIET, Jaunpur

# ACKNOWLEDGEMENT

We would like to express our special thanks of gratitude to our project guide Mr. Pravin Kumar Pandey, our HOD, our Dean, and all the other faculty members for their contributions to the completion of this project. We would also like to thank all our respected professors who helped us a lot in doing this work. This helped us in doing a lot of research and we learned about many things related to this topic and it also upgraded our skills.

# ABSTRACT

The "**Web 3.0 and Blockchain Based Decentralized Social Connectivity Web Application**" project presents a groundbreaking approach to social networking by leveraging decentralized technologies. Developed with Next.js for the frontend, Solidity for Ethereum smart contracts, MongoDB for user data management, and IPFS for decentralized storage, the platform prioritizes user privacy, security, and data ownership. Users can register, create and edit posts with IPFS integration, engage in social interactions, send messages, and manage profiles. The application's decentralized nature, driven by IPFS and Ethereum smart contracts, ensures tamper-resistant content storage and trustless transactions. The project aims to address the shortcomings of centralized social media, offering a decentralized alternative that empowers users and embraces the principles of transparency, security, and user control.

This decentralized social media web application not only addresses current challenges but also positions itself for scalability and future innovation. The modular architecture built with Next.js allows for seamless scalability of the front-end, ensuring a responsive and dynamic user interface as the platform grows. The integration of Ethereum smart contracts provides a foundation for potential future functionalities, including the implementation of decentralized autonomous organizations (DAOs) and novel token-based incentives, fostering a community-driven ecosystem.

# TABLE OF CONTENTS

|  |  |  |
| --- | --- | --- |
|  | Candidate’s Declaration | ii |
|  | Certificate | iii |
|  | Acknowledgement | iv |
|  | Abstract | v |
|  | Table of Contents | vi |
|  | List of figures | x |
|  |  |  |
| Chapter 1 | Introduction | 1-3 |
|  | 1.1 Motivation | 1 |
|  | 1.2 Problem Statement | 2 |
|  | 1.3 Objective | 2 |
|  | 1.4 Summary | 3 |
|  |  |  |
| Chapter 2 | Problem Formulation and Proposed Work | 4-7 |
|  | 2.1 Problem Definition | 4 |
|  | 2.1.1 Centralized Control | 4 |
|  | 2.1.2 Data Privacy Risks | 4 |
|  | 2.1.3 Content Censorship | 4 |
|  | 2.1.4 Single Points of Failure | 5 |
|  | 2.1.5 Lack of User Control | 5 |
|  | 2.1.6 Security Concerns | 5 |
|  | 2.1.7 Need for Resilience | 5 |
|  | 2.1.8 Privacy Invasion | 5 |
|  | 2.1.9 Monopolization of User Data | 5 |
|  | 2.1.10 Lack of Transparency | 6 |
|  |  |  |
|  | 2.2 Proposed Work | 6 |
|  | 2.2.1 Decentralized Architecture | 6 |
|  | 2.2.2 User-Centric Features | 6 |
|  | 2.2.3 Blockchain Integration | 6 |
|  | 2.2.4 IPFS for Decentralized Storage | 6 |
|  | 2.2.5 User-Friendly Frontend | 7 |
|  | 2.2.6 Security Measures | 7 |
|  | 2.2.7 Testing and Optimization | 7 |
|  |  |  |
| Chapter 3 | Literature Survey | 8-10 |
|  | 3.1 Methodologies | 8 |
|  | 3.1.1 Requirements Analysis | 8 |
|  | 3.1.2 Literature Review | 9 |
|  | 3.1.3 Technology Selection | 9 |
|  | 3.1.4 Architecture Design | 9 |
|  | 3.1.5 Prototyping | 9 |
|  | 3.1.6 Agile Development | 9 |
|  | 3.1.7 Smart Contract Development | 9 |
|  | 3.1.8 Frontend Development | 10 |
|  | 3.1.9 Integration Testing | 10 |
|  | 3.1.10 Security Measure Integration | 10 |
|  | 3.1.11 User Testing | 10 |
|  | 3.1.12 Optimization | 10 |
|  | 3.1.13 Continuous Improvement | 10 |
|  |  |  |
| Chapter 4 | Modeling (ER & DFD) | 11-15 |
|  | 4.1 Process Model (Iterative Waterfall Model) | 11 |
|  | 4.1.1 Requirements Gathering and Analysis | 11 |
|  | 4.1.2 System Design | 11 |
|  | 4.1.3 Implementation | 11 |
|  | 4.1.4 Integration and Testing | 12 |
|  | 4.1.5 Deployment | 12 |
|  | 4.1.6 Maintenance | 12 |
|  |  |  |
|  | 4.2 Data flow Diagrams | 12 |
|  | 4.2.1 Level 0 DFD | 13 |
|  | 4.2.2 Level 1 DFD | 14 |
|  | 4.2.3 Level 2 DFD | 14 |
|  |  |  |
|  | 4.3 Entity Relationship Diagram | 15 |
|  |  |  |
| Chapter 5 | System Requirements | 16-22 |
|  | 5.1 Hardware Requirements | 16 |
|  | 5.1.1 Server Infrastructure | 16 |
|  | 5.1.2 Blockchain Node | 17 |
|  | 5.1.3 Database Server | 17 |
|  | 5.1.4 IPFS Node | 17 |
|  | 5.1.5 Networking | 17 |
|  |  |  |
|  | 5.2 Software Requirements | 17 |
|  | 5.2.1 Operating System | 17 |
|  | 5.2.2 Web Server | 18 |
|  | 5.2.3 Database Management System | 18 |
|  | 5.2.4 Blockchain Node | 18 |
|  | 5.2.5 Smart Contract Development | 18 |
|  | 5.2.6 Frontend Framework | 18 |
|  | 5.2.7 IPFS Integration | 18 |
|  | 5.2.8 Version Control | 18 |
|  | 5.2.9 Package Managers | 19 |
|  | 5.2.10 Development Environment | 19 |
|  |  |  |
|  | 5.3 Functional Requirements | 19 |
|  | 5.3.1 User Authentication | 19 |
|  | 5.3.2 Profile Management | 19 |
|  | 5.3.3 Post Management | 19 |
|  | 5.3.4 Social Interactions | 20 |
|  | 5.3.5 User Relationships | 20 |
|  | 5.3.6 Search and Discovery | 20 |
|  | 5.3.7 Decentralized Storage Integration | 20 |
|  | 5.3.8 Security Measures | 20 |
|  |  |  |
|  | 5.4 Non-Functional Requirements | 20 |
|  | 5.4.1 Performance | 21 |
|  | 5.4.2 Security | 21 |
|  | 5.4.3 Reliability | 21 |
|  | 5.4.4 Scalability | 21 |
|  | 5.4.5 Usability | 21 |
|  | 5.4.6 Compatibility | 21 |
|  | 5.4.7 Interoperability | 22 |
|  | 5.4.8 Maintainability | 22 |
|  | 5.4.9 Privacy | 22 |
|  | 5.4.10 Regulatory Compliance | 22 |
|  | 5.4.11 Auditability | 22 |
|  | 5.4.12 Cost Efficiency | 22 |
|  |  |  |
| Chapter 6 | Testing | 23-26 |
|  | 6.1 Types of Testing | 23 |
|  | 6.1.1 Blackbox Testing | 23 |
|  | 6.1.2 Whitebox Testing | 24 |
|  |  |  |
|  | 6.2 Levels of Testing | 25 |
|  | 6.2.1 Unit Testing | 25 |
|  | 6.2.2 Integration Testing | 25 |
|  | 6.2.3 System Testing | 26 |
|  | 6.2.4 User Acceptance Testing | 26 |
|  |  |  |
|  | Conclusion | 27 |
|  | References | 28 |
|  |  |  |

|  |  |  |
| --- | --- | --- |
|  | **TABLE OF FIGURES** |  |
| **Fig No.** | **Fig Name** | **Page No.** |
| 4.1 | Iterative Waterfall Model | 12 |
| 4.2 | Level 0 DFD | 13 |
| 4.3 | Level 1 DFD | 14 |
| 4.4 | Level 2 DFD | 14 |
| 4.5 | ER Diagram | 15 |
| 6.1 | Black Box Testing | 24 |
| 6.2 | White Box Testing | 24 |
| 6.3 | Unit Testing & Integration Testing | 25 |