

## **B.Sc. (Computer Science) – Software Development Project Proposal**

<b>Course Title</b>	<b>Software Development Project</b>
<b>Project Title</b>	<b>BharatVote – A Blockchain-Based Secure Voting System</b>
<b>Student(s)</b>	<ul style="list-style-type: none"><li>• Archee Arjun (2022EBCS172)</li><li>• Shivangi Priya (2022EBCS261)</li><li>• Mohd Sultan (2022EBCS302)</li><li>• Keshav Gupta (2022EBCS129)</li></ul>
<b>Project Advisor</b>	Prof. Pravin Pawar
<b>Date of Submission</b>	04 July 2025

### **1. Project Idea Summary**

- Title of Project:**

**BharatVote** – A Blockchain-Based Secure Voting Application

- Abstract:**

**BharatVote** is a proposed voting application that leverages blockchain technology and commit-reveal cryptography to enable secure and privacy-preserving remote elections. The project envisions a system where voters can submit encrypted ballots that are later revealed and verified, ensuring both confidentiality and auditability. In Phase 1, the focus is on designing the system architecture, specifying requirements, and exploring a mock VoterID-based KYC mechanism for voter verification. The expected outcome is a clear and technically sound blueprint that can be implemented in the next phase.

### **2. Project Background and Motivation**

- Problem Statement:**

India's current Electronic Voting Machines (EVMs) operate as closed systems, limiting public transparency and independent verifiability. Only a small percentage of VVPAT slips are manually verified, and the system heavily depends on physical infrastructure, which excludes many voters such as internal migrants, NRIs, and personnel with mobility constraints. These challenges highlight the need for a secure, verifiable, and remotely accessible digital voting mechanism.

- Motivation:**

The project was chosen to explore how blockchain-based smart contracts can improve election transparency while preserving voter privacy. **BharatVote** proposes a modern,

mobile-friendly voting solution that supports cryptographic verification and digital identity checks. The initiative aligns with real-world needs, addresses a socially relevant gap, and provides a strong opportunity to apply concepts from cryptography, software engineering, and full-stack development in a meaningful way.

### **3. Objectives**

- **Primary Objectives:**

- Define and document a secure voting system design based on commit-reveal cryptography.
- Propose a voter verification flow using a mock VoterID-based KYC service.
- Design a modular application architecture for both web and mobile platforms with MetaMask integration.
- Specify functional and non-functional requirements in the form of an SRS document.
- Plan for on-device facial recognition support as part of the system architecture, ensuring privacy and eliminating server-side load.

- **Secondary Objectives:**

- Explore the feasibility of implementing facial recognition using browser- and app-based local processing libraries (e.g., face-api.js, ML Kit).
- Maintain a scalable and open-source design to support future upgrades like public blockchain deployment or real KUA integration.
- Present a clear roadmap for Phase-2 implementation based on design and documentation completed in this phase.

### **4. Project Scope and Expected Deliverables (w.r.t. the study project)**

- **Scope Definition:**

This phase focuses solely on the planning and design of a secure, blockchain-based voting system. It includes problem analysis, technology selection, system architecture, and documentation of functional and non-functional requirements. The project will explore wallet-based authentication, mock VoterID KYC integration, and incorporate architectural planning for facial recognition using on-device processing. No live implementation or deployment will be carried out in this phase. Actual software development and testing will be part of the next semester's full project course.

- **Deliverables:**

- **Software Solution:**

A proposed design blueprint for the **BharatVote** application, including smart contract logic, mobile/web architecture, identity verification flow (mock VoterID-based KYC), and a facial recognition plan designed to run locally on user devices.

- **Documentation:**

- Project Proposal (this document)
    - Software Requirements Specification (SRS)

- High-level architecture diagrams
- Role and milestone breakdown
- Technology stack rationale

- **Additional Assets:**

- A 10-slide summary presentation
- A short (3-minute) demo video describing the proposed design, interface concepts, and project vision

## 5. Preliminary Project Timeline and Milestones (w.r.t. the study project)

- **Proposed Schedule:**

Phase	Duration (Approx.)	Key Activities
Phase 1	Week 1–2	Finalize project idea, define objectives, and prepare initial proposal draft
Phase 2	Week 3–6	Conduct requirement analysis, finalize tech stack, and prepare the SRS document
Phase 3	Week 7–10	Design system architecture, user flow diagrams, and technical documentation
Phase 4	Week 11–12	Prepare final deliverables including presentation, demo walkthrough, and team reflection

- **Key Milestones:**

- Project idea and scope finalized
- Proposal approved by mentor
- SRS and architecture documentation completed
- Presentation and evaluation-ready assets prepared

## 6. Team Structure and Collaboration

- **Roles and Responsibilities:**

Team Member	Role	Responsibilities
Archee Arjun	Blockchain Design Lead	Define smart contract logic, commit-reveal mechanism, and technical feasibility
Shivangi Priya	Documentation Coordinator	Prepare proposal, SRS, architecture docs, and assist in presentation creation
Mohd Sultan	UI/UX & Integration Planner	Draft frontend architecture (web and mobile), design mock interfaces, and outline MetaMask integration
Keshav Gupta	Research & Security Analyst	Analyze cryptographic design, explore KYC workflow, and assess risks and privacy compliance

### **Collaboration Plan:**

The team will meet twice weekly for internal progress reviews and once weekly with the assigned mentor for feedback and guidance. All documents and plans will be maintained collaboratively using Google Docs and GitHub. Version control practices will be followed for documentation drafts using a shared Git repository. Real-time coordination will be handled via Google Chat and Google Meet.

### **7. Supervisor Review and Approval**

- Advisor Feedback:

*(To be filled in by the project advisor after reviewing the proposal. May include comments, suggestions, or approval status.)*

---

Supervisor Name: \_\_\_\_\_

Designation: \_\_\_\_\_

Date: \_\_\_\_\_

Signature: \_\_\_\_\_