# **Modules Distribution**

## **1. Backend & Schema Development**

**Team**: Vaibhav, Chaitanya  
**Tasks**:

* Define database schema (tables, relations, and constraints).
* Set up the backend environment (Node.js, Express, etc.).
* Integrate database (MySQL/PostgreSQL/MongoDB).
* Implement APIs for CRUD operations (e.g., user management, voting).
* Unit test API endpoints.
* Document APIs for front-end use.

**Dependencies**: Inputs from Smart Contracts module.

## **2. Smart Contract**

**Team**: Aryan, Vaibhav, Chaitanya  
**Tasks**:

* Design the architecture for blockchain integration.
* Write smart contracts for the voting system (using Solidity, Rust, etc.).
* Include logic for voter authentication, vote casting, and results computation.
* Test smart contracts on a testnet (e.g., Ethereum, Polygon).
* Deploy on a local blockchain environment.
* Share endpoints or contract interaction methods with the backend team.

**Dependencies**: Collaboration with the Security module to ensure contract integrity.

## **3. Security**

**Team**: Aryan, Tanay, Ayush  
**Tasks**:

* Conduct a security analysis of the smart contracts.
* Implement authentication (OAuth, JWT) and data encryption in the backend.
* Perform penetration testing for the backend, smart contracts, and APIs.
* Review front-end security for data transfer and storage.

**Dependencies**: Inputs from Backend and Smart Contracts modules.

## **4. Testing**

**Team**: Ayush,Chaitanya  
**Tasks**:

* Create unit and integration test cases for all modules.
* Perform load and performance testing for real-time operations.
* Conduct end-to-end testing with front-end integration.

**Dependencies**: All modules must be ready or near completion.

## **5. Real-time Operations & Concurrency**

**Team**: Ayush, Tanay  
**Tasks**:

* Design real-time communication (WebSocket, SignalR, or polling) for live voting updates.
* Implement concurrency handling for high traffic during voting.
* Test real-time functionality with simulated traffic.

**Dependencies**: Integrated with Backend and Front-end.

## **6. Front-end Development**

**Team**: All Members  
**Tasks**:

* Design UI/UX mockups for key features (dashboard, voting interface).
* Implement components using React, Angular, or Vue.
* Integrate front-end with backend APIs and smart contracts.
* Conduct usability testing and collect feedback.

**Dependencies**: Backend and Smart Contracts modules.