

Project Design Phase-II

Technology Stack (Architecture & Stack)

| | |
|---------------|--|
| Date | 06/02/2026 |
| Team ID | LTVIP2026TMIDS80883 |
| Project Name | Prosperity Prognosticator: Machine Learning for Startup Success Prediction |
| Maximum Marks | 4 Marks |

Technical Architecture:

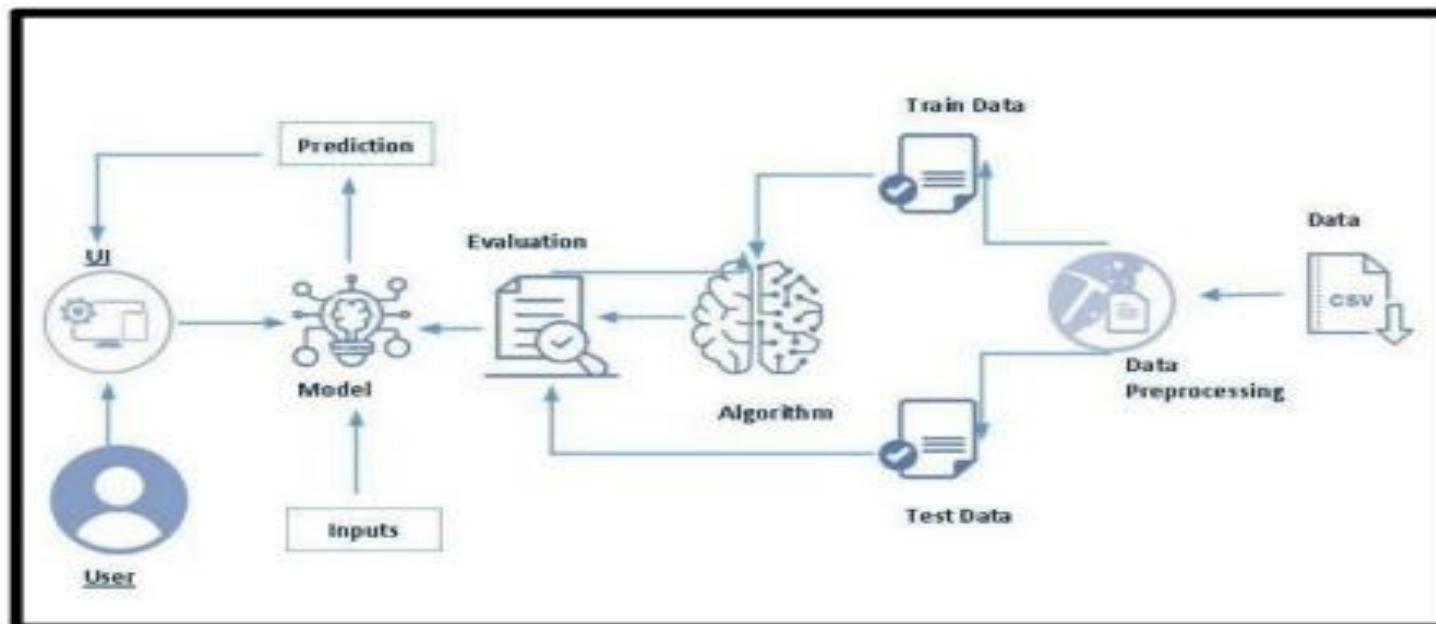


Table-1 : Components & Technologies:

| S. No | Component | Description | Technology |
|-------|---------------------------------|--|---|
| 1. | User Interface | Web interface where users enter startup details and view predictions | HTML, CSS, JavaScript, Bootstrap |
| 2. | Application Logic-1 | Backend logic for handling user requests and data validation | Python (Flask Framework) |
| 3. | Application Logic-2 | Machine Learning prediction processing | Scikit-learn (Random Forest) |
| 4. | Application Logic-3 | Prediction Result Processing & Confidence Score Generation | Python (Flask), Scikit-learn (predict_proba), JSON Handling |
| 5. | Database | Stores user details and startup data | MySQL |
| 6. | Cloud Database | Cloud-hosted database for production | AWS RDS / Cloud MySQL |
| 7. | File Storage | Stores ML model (.pkl) and reports | Local File System / AWS S3 |
| 8. | External API-1 | Gmail authentication for login | Google OAuth API |
| 9. | External API-2 | LinkedIn authentication integration | LinkedIn OAuth API |
| 10. | Machine Learning Model | Predicts startup success or failure | Random Forest Classifier |
| 11. | Infrastructure (Server / Cloud) | Application deployment environment | Local Server (Development) / AWS EC2 (Production) |

Table-2: Application Characteristics:

| S. No | Characteristics | Description | Technology |
|-------|--------------------------|---|---------------------------------------|
| 1. | Open-Source Frameworks | Frameworks used for development | Flask, Scikit-learn, Bootstrap |
| 2. | Security Implementations | User authentication, encrypted passwords, secure API access | SHA-256 Hashing, HTTPS, OAuth 2.0 |
| 3. | Scalable Architecture | 3-tier architecture allows scaling of frontend, backend, and database independently | AWS EC2, Load Balancer |
| 4. | Availability | System hosted on cloud ensuring 24/7 uptime | AWS Cloud Infrastructure |
| 5. | Performance | Fast prediction response (<3 seconds), optimized model, minimal latency | Flask API Optimization, Model Caching |