**Project Design Phase-II**

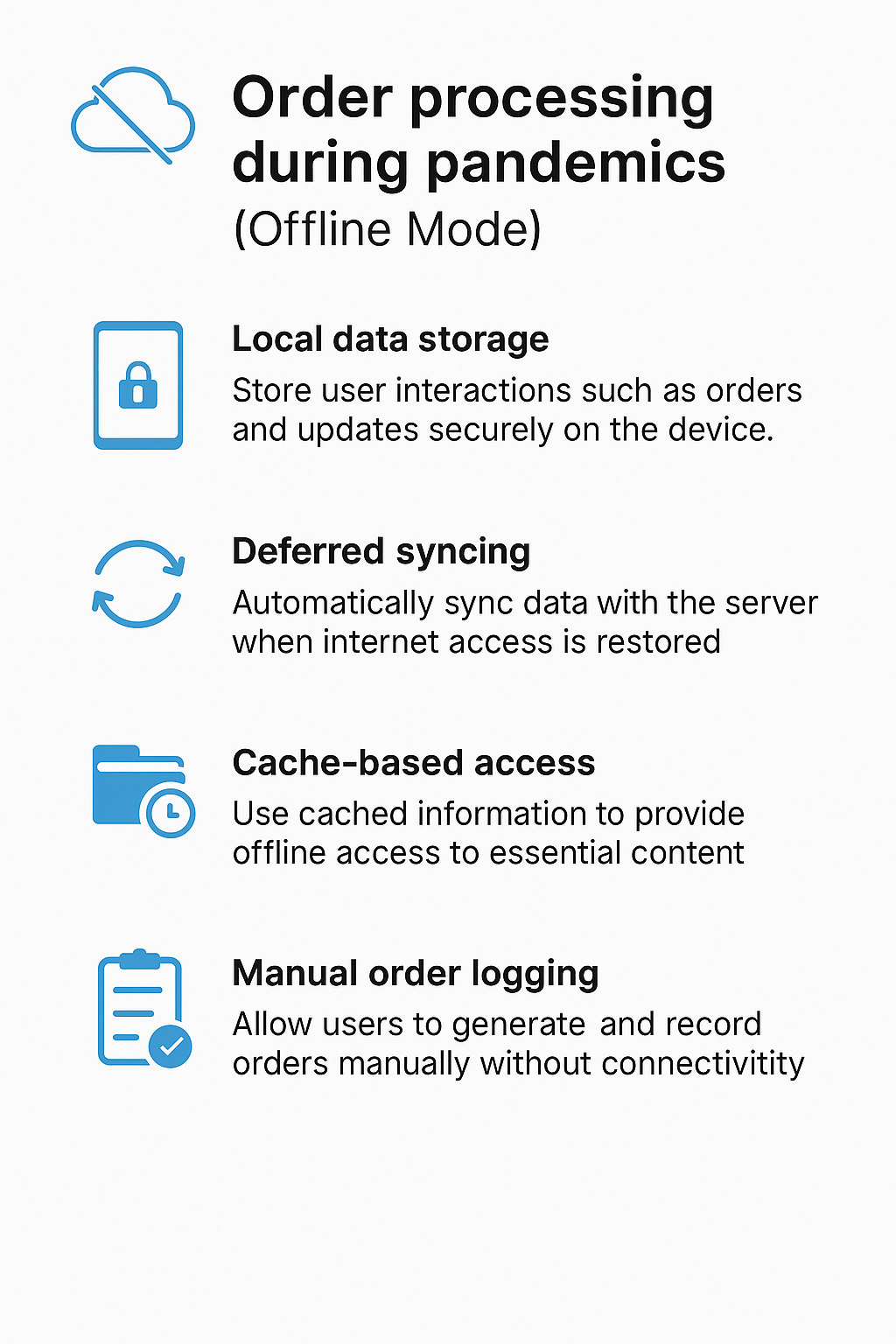
**Technology Stack (Architecture & Stack)**

|  |  |
| --- | --- |
| Date | 20 July 2025 |
| Team ID | LTVIP2025TMID41443 |
| Project Name | Transfer Learning-Based Classification of Poultry Diseases for Enhanced Health Management |
| Maximum Marks | 4 Marks |

**Technical Architecture:**

The Deliverable shall include the architectural diagram as below and the information as per the table1 & table 2

**Example: Order processing during pandemics for offline mode**

****

Guidelines:

1. Store poultry image uploads and disease predictions locally on the device.

2. Enable the app to work without internet using cached disease models and treatment info.

3. Sync all saved reports to the server automatically once internet is restored.

4. Allow farmers to log symptoms and get offline suggestions from cached data.

5. Secure all offline data and avoid duplicate submissions during sync.

**Table-1 : Components & Technologies:**

|  |  |  |  |
| --- | --- | --- | --- |
| **S.No** | **Component** | **Description** | **Technology** |
|  | User Interface | Web and mobile-based UI for farmers to upload poultry images | HTML, CSS, JavaScript / Angular Js / React Js etc. |
|  | Application Logic-1 | Backend logic for image processing and disease classification | Python |
|  | Application Logic-2 | Disease prediction using trained ML model | TensorFlow / Keras with Transfer Learning |
|  | Application Logic-3 | Treatment suggestion based on predicted disease | Python logic + JSON-based mapping |
|  | Database | Stores user accounts and image metadata | SQLite / MySQL |
|  | Cloud Database | Stores remote data for analytics (optional) | IBM Cloudant / Firebase Realtime DB |
|  | File Storage | Stores uploaded poultry images locally or in the cloud | Local Filesystem / IBM Cloud Object Store |
|  | External API-1 | Weather conditions for disease correlation (optional) | IBM Weather API, etc. |
|  | External API-2 | |  | | --- | |  |  |  | | --- | | Location-based vet services (optional) | | Aadhar API, etc. |
|  | Machine Learning Model | Classifies poultry images into disease categories | MobileNet / ResNet Transfer Learning Model |
|  | Infrastructure (Server / Cloud) | Hosts the app and ML model backend | Local Server / Render / Heroku / IBM Cloud |

**Table-2: Application Characteristics:**

| **S.No** | **Characteristics** | **Description** | **Technology** |
| --- | --- | --- | --- |
|  | Open-Source Frameworks | Frontend, backend, and ML libraries used are open-source | Flask, TensorFlow, Bootstrap, SQLite |
|  | Security Implementations | Password hashing, file validation, and API protection | SHA-256, HTTPS, Flask-CORS, JWT (optional) |
|  | Scalable Architecture | Justify the scalability of architecture (3 – tier, Micro-services) | 3-tier architecture + Docker-ready |
|  | Availability | Justify the availability of application (e.g. use of load balancers, distributed servers etc.) | Render, Heroku, IBM Cloud Foundry |
|  | Performance | Caching predictions, optimized image size, fast API responses | Flask Cache, CDN (optional), SQLite Indexing |