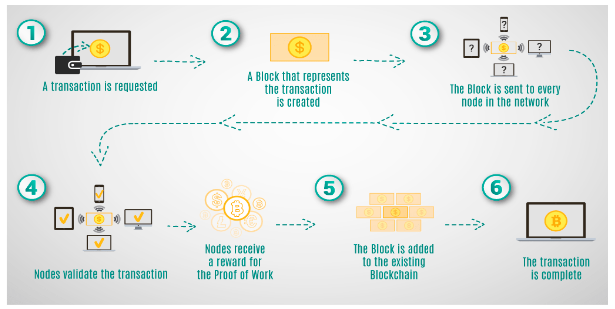
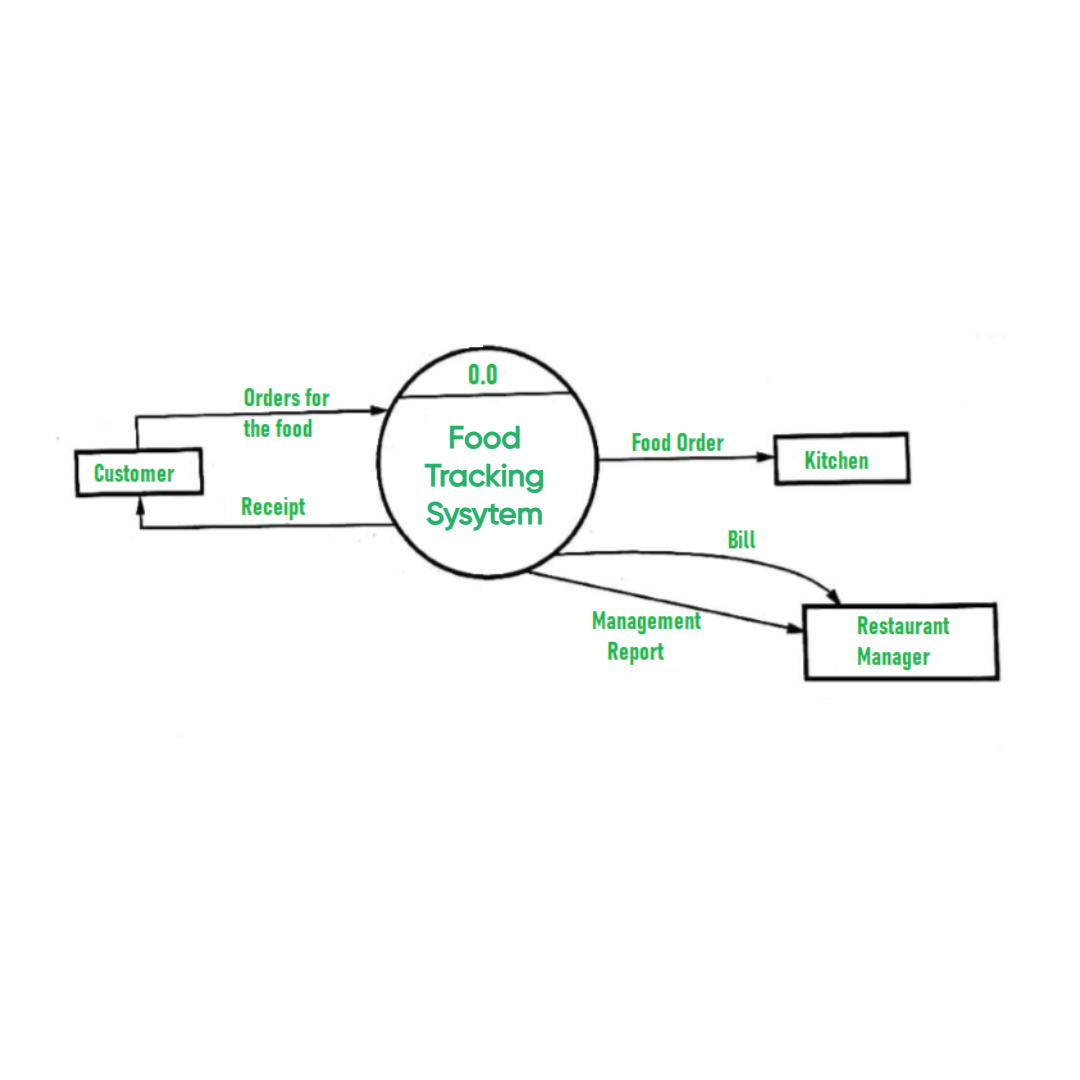
**Project Design Phase-II**

**Technology Stack (Architecture & Stack)**

|  |  |
| --- | --- |
| Date | 19 Oct 2023 |
| Project Name | Food Tracking System |

**Technical Architecture:**

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

****

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

|  |  |  |  |
| --- | --- | --- | --- |
| **S No** | **Component** | **Description** | **Technology** |
|  | User Registration | Allows users to register within the system and specify their roles and permissions.  Utilizes a user-friendly React.js interface for registration. | React.js: A user-friendly web framework for building the registration interface.  Solidity: Smart contract language for managing user roles and permissions on the blockchain. |
|  | Resource Catalog | Enables supply chain stakeholders to catalog and manage food products, adding comprehensive metadata such as origin, processing, and transportation details. | Utilizes a user-friendly React.js interface for cataloging and indexing.React.js for the user interface, Solidity for smart contracts, and IPFS for decentralized storage. |
|  | Resource Tracking | Allows consumers to search for food products, verify their authenticity, and trace their journey in real-time. | Offers a user-friendly React.js interface for tracking and traceability.  Integrates with the blockchain, particularly Ethereum, for transparent cataloging and real-time updates. |
|  | User Dashboard | The user dashboard provides a user-friendly interface for consumers to manage their interactions with the system. | User Interface: Develops a user-friendly dashboard using React.js for a seamless user experience. |
|  | Security and Permissions | This component manages access control to ensure that the right users have the appropriate permissions within the system. | Utilizes Solidity smart contracts for enforcing access control.  Provides a React.js user interface for user authentication, and may integrate with appropriate security mechanisms. |
|  | Blockchain Integration | Forms the core of the system, integrating with blockchain technology (e.g., Ethereum).  Involves the development and deployment of smart contracts using Solidity. | Interfaces with relevant blockchain wallet services (e.g., Metamask) to ensure secure and seamless interactions. |

**Table-2: Application Characteristics:**

| **S No** | **Characteristics** | **Description** | **Technology** |
| --- | --- | --- | --- |
|  | Decentralization | The system operates on a decentralized blockchain network, eliminating the need for a central authority. This ensures data integrity and builds trust among supply chain stakeholders. | Ethereum blockchain for decentralization, Solidity smart contracts. |
|  | Immutability | Once data is recorded on the blockchain, it becomes immutable, meaning it cannot be altered or deleted. This ensures the integrity and reliability of the recorded information. | Blockchain technology (e.g., Ethereum) for data immutability. |
|  | Transparency | The system offers transparent access to food product information, supply chain transactions, and origin details. This enhances accountability and fosters user confidence. | Blockchain for transparent ledger, React.js for user interface. |
|  | Security | The system prioritizes data security by implementing encryption, access control, and decentralized storage. | Data encryption, blockchain for access control, Solidity smart contracts. |
|  | Data Privacy | Data privacy is a paramount concern, and the system implements stringent measures to safeguard sensitive user and food product information from unauthorized access. | Data encryption, access control, and blockchain for privacy. |