Case Study- iBank

iBank is a bank which is planning to work as a completely online banking solution for the customers. All the operations from opening an account to applying for a loan will be done online.

Functional Requirements:

For simplicity the operations being planned in the initial phase of the banking and mentioned below:

- Registering a user on the website
- Opening a savings account (should be scalable to cater various types)
- Opening a current account
- E-KYC
- Checking profile and updating related details
- Account related operations
    - Check account balance
    - Generate pdf statement
        - Absolute time frame
        - Relative time frame
    - Issue a cheque book
    - Generate e-statement
    - Email e-statement
    - Stop cheque
- Bank Transfer through RTGS, IMPS, NEFT
- Debit/Credit Card operations
    - Issue a debit card (should be scalable to cater various types)
    - Issue a credit card based on eligibility
    - Block a credit card
    - Increase/decrease credit limit
- Deposit Schemes and related operations
    - Open one/multiple fixed deposit accounts
    - Open one/multiple recurring deposit accounts
    - Select Tenure of deposits
    - Select monthly debit date for recurring deposits
    - Select mode of payment for fixed deposit/recurring deposit
    - Map each deposit account to savings account
- Loan Services
    - Types of Loan (Home Loan, Personal Loan etc.)
    - Apply for loan
    - Handle supporting documents
    - Apply for loan on credit card
    - Track loan disbursement details
    - Check loan account statement

Non-Functional Requirements

- Proposed solution must provide guard against various security threats (OWASP Top 10 etc.). System components should utilize industry-proven security standards and protocols.
- The website should be mobile friendly
- The website should cater 10 million users and 10,000 parallel users with average response time less than 5 seconds
- The offered solution must complete 99% of provided services in less than 5000 milliseconds, over both the peak and non-peak hours
- It should support both on premise and cloud-based deployment with auto-scaling to match demand.
- Proposed solution will have the ability to monitor and maintain the system, and include testability, configurability and upgradeability.
- System should have flexible design and it should be able to handle increased load in future.
- Systems is only in English with no multi-lingual requirements.

Deliverables:

Create an HLD which will contain the following items:

- A suitable architecture for the application (depict with the help of an architecture diagram and explanation for each component Propose a suitable tech stack for the application.
- Add description and scope for each service identified.
- Tech Stack for the application.
- A deployment diagram with detailed explanation of all the components involved.