Project Design
Phase-I Solution
Architecture

Date	19 September 2022
Team ID	Team-592903
Project Name	Project - **Detecting COVID-19 From Chest X-Rays Using Deep Learning Techniques**
Maximum Marks	4 Marks

Solution Architecture:

Solution architecture is a complex process – with many sub-processes – that bridges the gap between business problems and technology solutions. Its goals are to:

- Find the best tech solution to solve existing business problems.
- Describe the structure, characteristics, behavior, and other aspects of the software to project stakeholders.
- Define features, development phases, and solution requirements.
- Provide specifications according to which the solution is defined, managed, and delivered.

Example - Solution Architecture Diagram:

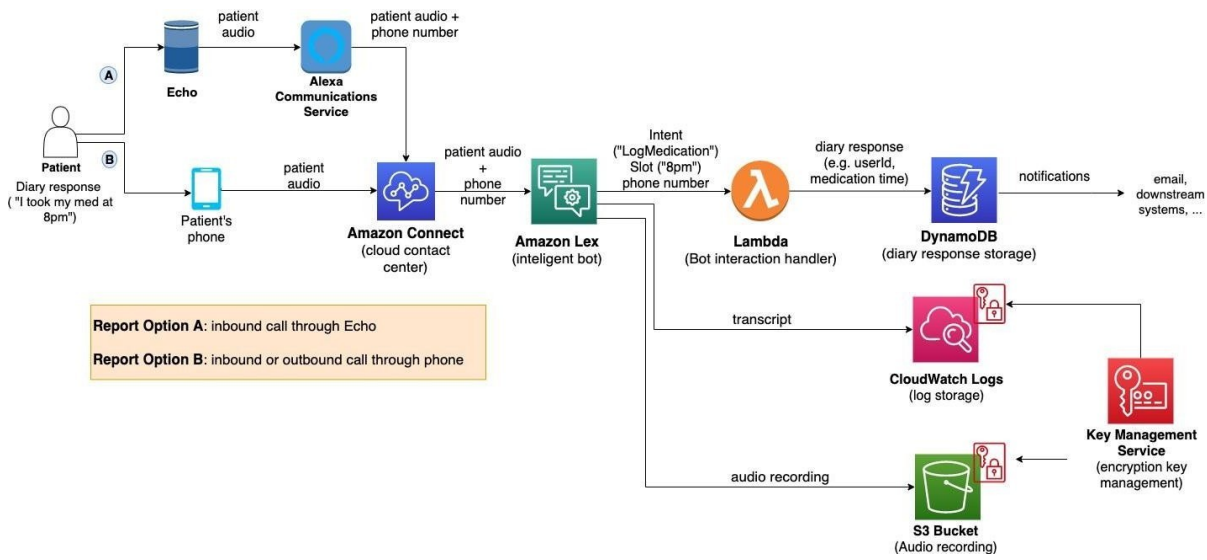

Figure 1: Architecture and data flow of the voice patient diary sample application

Reference: https://aws.amazon.com/blogs/industries/voice-applications-in-clinical-research-powered-by-ai-on-aws-part-1-architecture-and-design-considerations/

Solution on next Page:

1. Define Project Scope:

- Identify the specific goals of the COVID-19 Chest X-Ray Detection project.
- Determine the user base, such as healthcare professionals, hospitals, and clinics.

2. Gather Requirements:

- Conduct discussions with healthcare professionals to understand their needs.
- Define functional requirements (e.g., real-time analysis, accuracy) and non-functional requirements (e.g., security, scalability).

3. Identify Key Components:

- **Deep Learning Model:** Custom CNN architecture for COVID-19 detection.
- **Web-Based Application:** User interface for uploading X-ray images and displaying results.
- **Database:** Storage for curated X-ray datasets and model training data.
- **API:** Integration point for external systems.
- **Training Pipeline:** Mechanism for model training and updates.
- **Result Visualization Module:** Presenting results to healthcare professionals.

4. Define Interactions and Workflows:

- Map the flow of data and interactions between components.
- Define user workflows, from uploading X-ray images to receiving results.

5. Choose Technology Stack:

- **Deep Learning Framework:** TensorFlow for model development.
- **Web Application Framework:** Flask for the user interface.
- **Database Management System:** MongoDB for data storage.
- **API Framework:** Flask-RESTful for API development.
- **Cloud Services:** AWS for scalability.

6. Security Considerations:

- Implement encryption for data in transit and at rest.
- Define user authentication and authorization mechanisms.
- Ensure compliance with healthcare data security standards.

7. Scalability and Performance:

- Design components to scale horizontally.
- Optimize algorithms and code for real-time performance.

8. Data Flow and Storage:

- Define how data flows through the system from user input to model output.
- Choose efficient data storage solutions for X-ray datasets.

9. Integration Points:

- Identify external systems for integration (e.g., EHRs).
- Design API endpoints for seamless data exchange.

10. Error Handling and Logging:

- Implement error-handling mechanisms for unexpected situations.
- Set up logging to monitor system behavior and diagnose issues.

11. Testing and Validation:

- Develop unit tests for individual components.
- Plan for end-to-end testing and validation using diverse X-ray datasets.

12. Documentation:

- Create detailed documentation for each component and the overall system.
- Include architecture diagrams, data flow charts, and component specifications.

_----------x----------