

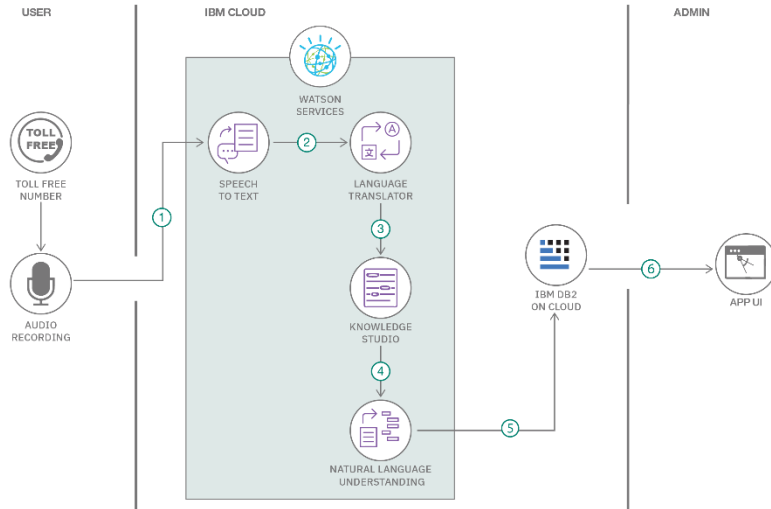
## Project Design Phase-II Technology Stack (Architecture & Stack)

Date	22 June 2025
Team ID	LTVIP2025TMID54898
Project Name	Resolvenow: your platform for online complaints
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:

- Include all the processes (As an application logic / Technology Block)
- Provide infrastructural demarcation (Local / Cloud)
- Indicate external interfaces (third party API's etc.)
- Indicate Data Storage components / services
- Indicate interface to machine learning models (if applicable)

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

S.No	Component	Description	Technology Used
1	User Interface	Web UI for customers, agents, and admin interactions	React.js, HTML, CSS, JavaScript, Bootstrap
2	Application Logic - 1	Handles routing, API processing, and business logic	Node.js, Express.js
3	Application Logic - 2	Role-based authentication and session handling	JWT (JSON Web Token), bcrypt.js
4	Application Logic - 3	Chat-based complaint discussion system	Socket.io (future enhancement)
5	Database	Stores user data, complaints, status, messages, roles	MongoDB
6	Cloud Database	Cloud-hosted NoSQL database	MongoDB Atlas
7	File Storage	Uploads like screenshots for complaints (optional)	Local filesystem or Cloudinary (optional)
8	External API - 1	Location or IP lookup (optional future enhancement)	IPInfo API (if used)
9	External API - 2	Not used in current version	—
10	Machine Learning Model	Not applicable in this version	—

11	Infrastructure (Server/Cloud)	Hosting backend and frontend	Render (Backend), Vercel/Netlify (Frontend)
----	----------------------------------	---------------------------------	--

**Table-2: Application Characteristics:**

S.No	Characteristics	Description	Technology Used
1	Open-Source Frameworks	Frameworks used in app development	React.js, Node.js, Express.js, MongoDB
2	Security Implementations	Role-based access, hashed passwords, token authentication	bcrypt.js, JWT, Helmet (Express middleware)
3	Scalable Architecture	Modular codebase with clear separation (Frontend, Backend, DB)	MERN Stack architecture (3-tier)
4	Availability	Hosted on cloud platforms accessible 24/7	Render, MongoDB Atlas
5	Performance	Uses asynchronous requests, light frontend framework, efficient DB queries	React, Axios, MongoDB

