**Project Design Phase-II**

**Technology Stack (Architecture & Stack)**

|  |  |
| --- | --- |
| Date | 26 june.2025 |
| Team ID | LTVIP2025TMID53135 |
| Project Name | LearnHUb:Your Center For Skill Enhancement |
| 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**

**Reference:** [**https://developer.ibm.com/patterns/ai-powered-backend-system-for-order-processing-during-pandemics/**](https://developer.ibm.com/patterns/ai-powered-backend-system-for-order-processing-during-pandemics/)

**[Frontend (React.js / HTML5)]**

**↓**

**[Axios API Calls]**

**↓**

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)

**[Backend (Express.js / Node.js)]**

**↓**

**[MongoDB Database]**

**↓**

**[Certificate Generator, Payment Gateway]**

**Additional:**

**- Admin Panel: Manages instructors, students, and content**

**- Cloud Deployment (AWS / Render / Firebase)**

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

|  |  |  |
| --- | --- | --- |
| **Component** | **Description** | **Technology** |
| **UI** | **Web Interface** | **HTML, CSS, ReactJS, Bootstrap** |
| **Application Logic** | **Backend services** | **Node.js, Express.js** |
| **Database** | **Course/User storage** | **MongoDB** |
| **Cloud** | **Hosting & scaling** | **AWS or Render** |
| **File Storage** | **Certificates, course media** | **AWS S3 or local** |
| **API** | **Payment & Email** | **Razorpay, Nodemailer** |

**Table 2: Application Characteristics**

|  |  |  |
| --- | --- | --- |
| **Characteristic** | **Description** | **Technology** |
| **Open Source** | **Frameworks used** | **ReactJS, ExpressJS** |
| **Security** | **Authentication & Encryption** | **JWT, bcrypt, HTTPS** |
| **Scalability** | **Modular & scalable design** | **Microservices, Docker (optional)** |
| **Availability** | **Uptime & deployment** | **Cloud hosting, load balancer** |
| **Performance** | **Optimized delivery** | **CDN, lazy loading, MongoDB indexing** |

**References:**

[**https://c4model.com/**](https://c4model.com/)

[**https://developer.ibm.com/patterns/online-order-processing-system-during-pandemic/**](https://developer.ibm.com/patterns/online-order-processing-system-during-pandemic/)

[**https://www.ibm.com/cloud/architecture**](https://www.ibm.com/cloud/architecture)

[**https://aws.amazon.com/architecture**](https://aws.amazon.com/architecture)

[**https://medium.com/the-internal-startup/how-to-draw-useful-technical-architecture-diagrams-2d20c9fda90d**](https://medium.com/the-internal-startup/how-to-draw-useful-technical-architecture-diagrams-2d20c9fda90d)