Project Design Phase-II

Technology Stack (Architecture & Stack)

|  |  |
| --- | --- |
| Date | 26 JUNE 2025 |
| Team ID | LTVIP2025TMID59454 |
| 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: Order processing during pandemics for offline mode Reference: https://[www.coursera.org](http://www.coursera.org/)

Table-1 : Components & Technologies:

|  |  |  |  |
| --- | --- | --- | --- |
| **S.No** | **Component** | **Description** | **Technology** |
| 1. | User Interface | Web-based interface for learners, educators, and  admins | HTML, CSS, JavaScript/ React Js etc. |
| 2. | Application Logic-1 | Logic for course creation, uploading videos, adding modules | Node js, Express js |
| 3. | Application Logic-2 | Logic for enrollment, tracking progress, submitting  assessments | Node js, Express js |
| 4. | Database | Stores user info, course details, quiz results, progress tracking | MongoDB |
| 5. | Authentication Module | Manages secure user login, registration, password  rese | JWT (JSON Web Tokens), bcrypt,  Passport.js |
| 6. | Application Logic – Admin | Logic for approving/rejecting courses, user management, monitoring usage | Node.js, Express.js |
| 7. | Certificate Generator | Auto-generates downloadable certificates on course completion | PDFKit / jsPDF / HTML2PDF |
| 8. | Discussion Forum / Chat | Enables student-instructor and student-student communication | Socket.io / Firebase Realtime DB / CometChat |

|  |  |  |  |
| --- | --- | --- | --- |
| 9. | Hosting & Deployment | Hosting frontend, backend, and database | Vercel / Netlify (Frontend), Render / Heroku / AWS EC2 (Backend), MongoDB Atlas (Database) |

Table-2: Application Characteristics:

|  |  |  |  |
| --- | --- | --- | --- |
| **S.No** | **Characteristics** | **Description** | **Technology** |
| 1. | Open-Source Frameworks | Frontend and backend built using open-source  technologies | React js, Node js, Express js |
| 2. | Scalable Architecture | 3-tier architecture with RESTful APIs-based  services. | Microservices, REST API, MVC pattern |
| 3. | Cross-Platform Compatibility | Works on all major browsers and mobile devices | HTML5, CSS3, Responsive Web Design |
| 4. | Cloud-Ready Infrastructure | Can be deployed on cloud platforms with  autoscaling | AWS, Heroku, Vercel, Docker |
| 5. | Real-Time Communication | Enables instant messaging and live interaction in forums | Socket.io, WebSockets |
| 6. | Secure User Management | Ensures secure registration, login, and role-based  access | JWT, OAuth 2.0, bcryp |
| 7. | Data-Driven Insights | racks learner behavior and progress for analytics | MongoDB Aggregation, Google  Analytics |

References:

React.js Documentation Node js Best Practice

https://medium.com/the-internal-startup/how-to-draw-useful-technical-architecture-diagrams-2d20c9fda90d