**Requirement Gathering and Analysis Phase**

**Solution Requirements (Functional & Non-functional)**

|  |  |
| --- | --- |
| Date | 06-07-2024 |
| Team ID | PNT2022TMID SWTID1720175019 |
| Project Name | Nexus Learn – Online learning platform |
| Maximum Marks |  |

**Functional Requirements:**

Following are the functional requirements of the proposed solution.

|  |  |  |
| --- | --- | --- |
| **FR No.** | **Functional Requirement (Epic)** | **Sub Requirement (Story / Sub-Task)** |
| FR-1 | User Registration | Registration through Form  Registration through Gmail  Registration through LinkedIN |
| FR-2 | User Confirmation | Confirmation via Email  Confirmation via OTP |
| FR-3 | Learner | Browse courses  Enroll courses  Access learning materials  Assessments  Track courses  Interactive feedback |
| FR-4 | Instructor | Create and manage courses  Upload content and resources  Deliver lectures  Review assignments  Communicate with learners |
| FR-5 | Administrator | Manage users  Courses centre  Content library  Payment system  Monitor platform performance |
| FR-6 | Course Management | Create  Edit  Publish courses with modules  Lessons  Quizzes |
| FR-7 | Community Features | Discussion forums  Live Q&A sessions  Messaging system. |
| FR-8 | Search and Filtering | Search by Course Name  Filter by category, rating, price |

**Non-functional Requirements:**

Following are the non-functional requirements of the proposed solution.

|  |  |  |
| --- | --- | --- |
| **FR No.** | **Non-Functional Requirement** | **Description** |
| NFR-1 | **Usability** | React.js for a responsive and intuitive user interface. User testing to refine UI/UX. |
| NFR-2 | **Security** | JWT for secure authentication, HTTPS for secure communication, bcrypt for password hashing, CORS policies for access control. |
| NFR-3 | **Reliability** | MongoDB Atlas for cloud database services with redundancy and backup features. Robust error handling in Node.js. |
| NFR-4 | **Performance** | Redis for caching to improve response times, CDN integration for efficient delivery of static assets. |
| NFR-5 | **Availability** | Deployment on AWS EC2 with Elastic Load Balancing to ensure high availability and minimal downtime. |
| NFR-6 | **Scalability** | Microservices architecture with Node.js for scalable backend services. Horizontal scaling using containerization. |