**Task : 08 :**

**Software Requirements Specification (SRS) for Study App Application :**

**1. Project Initiation :**

**- Purpose & Scope :** Develop a user-friendly Study App to enhance the learning experience.

**- Team Members & Roles :** Identify stakeholders and assign roles for efficient project management.

**- Goals & Objectives :** Create a versatile app with rich study resources, quizzes, and effective user interaction.

**2. Requirements Gathering :**

**- App Features & Functionality :** Study materials, quizzes, progress tracking, user profiles, and interactive quizzes.

**- User Experience Design :** Intuitive UI/UX design to ensure a seamless user experience.

**3. Design & Planning :**

**- App Architecture & Technology Stack :** Utilize modern technologies and scalable architecture.

**- Wireframes & Mockups :** Develop detailed visual representations of the app's interface.

**4. Development :**

**- Front-end Development :** Create engaging UI components and user interactions.

**- Back-end Development :** Build a robust back-end, database, and APIs.

**- Integration & Testing :** Integrate and rigorously test both front-end and back-end components.

**5. Quality Assurance :**

**- Testing & Bug Fixing :** Conduct comprehensive testing and promptly address issues and bugs.

**- Performance Optimization :** Optimize app performance, responsiveness, and loading times.

**6. Deployment :**

**- App Launch Plan :** Prepare for a successful launch, including app store submissions.

**- User Communication :** Develop strategies for effective communication with users.

**7. Post Launch Activities :**

**- User Support & Feedback :** Offer continuous support and gather feedback for improvements.

**- Ongoing Updates & Enhancements :** Plan future updates based on user feedback.

**8. Project Closure :**

**- Finalize Documentation :** Ensure complete documentation, including user guides and technical specifications.

**- Evaluate Success :** Review the project's success and areas for improvement.

**Materials Used :**

**- Development Tools :** Utilize industry-standard development tools, such as integrated development environments (IDEs), version control systems, and project management software.

**- Programming Languages :** Use languages like JavaScript, Python, and SQL for front-end, back-end, and database development.

**- Database Management System (DBMS) :** Implement a reliable DBMS for data storage and retrieval.

**- UI/UX Design Tools :** Employ design software like Adobe XD or Sketch for wireframes and mockups.

**- Testing Frameworks :** Use testing frameworks like Selenium or Jest for automated testing.

**- Documentation Tools :** Utilize word processing software (e.g., Microsoft Word) for creating user guides, technical specifications, and project reports.

**- Communication Tools :** Employ email, video conferencing, and collaboration platforms for team communication and user support