Aim: Mini Project on object oriented software engineering for a business application.

Resources needed:

Activities:

Students are required to choose one business application and prepare following for the same.

1. Problem Definition

2. Project Scope

3. Choice of Process Model

4. Roles and Responsibilities

5. GUI based Implementation of one Module(one use case)

Results:

1. **Problem Definition:**

Problem Statement: QuirkIQ - Revolutionizing Science Education for 11th and 12th Grade Students

In the rapidly evolving landscape of education, there exists a pressing need to redefine and enhance the learning experience for 11th and 12th-grade science students. Traditional educational methods often fall short in engaging students and providing them with practical exposure to real-world applications of their theoretical knowledge. In this context, QuirkIQ emerges as a pioneering e-learning platform designed to address these challenges and elevate the educational journey of science enthusiasts.

Unique Enhancements:

1. Virtual Internship Platform:

- QuirkIQ introduces a groundbreaking Virtual Internship platform that allows science students to gain practical, industry-relevant experience within a controlled digital environment. This feature bridges the gap between theoretical understanding and practical application.

2. Multi-language Support:

- QuirkIQ is committed to inclusivity by offering multi-language support. Students can access learning materials, lectures, and assessments in their preferred

language, breaking down language barriers and ensuring a personalized learning experience.

2. **Project Scope:**

Project Scope for QuirkIQ: E-Learning Platform for 11th and 12th Grade Science Students

1. Platform Overview:

- Develop an intuitive and user-friendly e-learning platform named QuirkIQ, specifically designed for 11th and 12th-grade science students.

- The platform will provide comprehensive coverage of physics, chemistry, and biology, aligning with the curriculum standards for these grades.

2. User Registration and Profiles:

- Implement a secure user registration system allowing students to create accounts with unique profiles.

- User profiles will track individual progress, achievements, and preferences, providing a personalized learning experience.

3. Course Content Delivery:

- Organize and present course content through structured modules, covering topics from the 11th and 12th-grade science curriculum.

- Content delivery includes text-based materials, multimedia presentations, and interactive simulations to enhance understanding.

4. Virtual Internship Platform:

- Develop a Virtual Internship platform where students can engage in simulated real-world scenarios related to their respective science fields.

- Incorporate practical challenges, experiments, and projects to allow students to apply theoretical knowledge in a practical setting.

5. Multi-language Support:

- Integrate multi-language support to ensure accessibility for students from diverse linguistic backgrounds.

- Allow users to select their preferred language for learning materials, assessments, and platform navigation.

6. Interactive Assessments:

- Create a variety of interactive assessments, including quizzes, assignments, and practical evaluations, to gauge student understanding and progress.

- Provide instant feedback and performance analytics to help students track their strengths and areas for improvement.

7. Gamification Elements:

- Introduce gamification elements, such as badges, leaderboards, and rewards, to enhance user engagement and motivation.

- Recognize and celebrate achievements to create a positive learning environment.

8. Responsive Design:

- Ensure that QuirkIQ is accessible on various devices, including desktops, laptops, tablets, and smartphones.

- Implement responsive design principles to optimize the user experience across different screen sizes.

9. Scalability:

- Design the platform with scalability in mind, allowing for the addition of new courses, features, and user capacity as the platform grows.

10. Documentation and Training:

- Provide comprehensive documentation for users, including tutorials and guides on using the platform effectively.

11. Continuous Improvement:

- Establish mechanisms for collecting user feedback and use it to continuously improve the platform's features, content, and overall user experience.

3. Choice of Process Model:

Suitable process model for the development of QuirkIQ could be the Spiral Model. The Spiral Model combines the idea of iterative development (like Agile) with the systematic aspects of the waterfall model. Given the complexity of the project, potential changes in requirements, and the need for risk management, the Spiral Model could offer a structured yet flexible approach.

Spiral Model:

- Rationale:

- QuirkIQ involves multiple unique features and requirements that may evolve during the development process.

- The Spiral Model allows for risk assessment and mitigation, making it suitable for a project where uncertainties exist.

- Characteristics:

- Iterative Development: The project is divided into small cycles, each involving planning, risk analysis, engineering, and testing.

- Risk Management: The model incorporates a focus on risk assessment and mitigation in each iteration.

- Flexibility: Adaptability to changes in requirements, and the ability to introduce new features in subsequent spirals.

- Phases:

- Planning: Define objectives, identify risks, and plan the development process.

- Risk Analysis: Evaluate identified risks and develop strategies for mitigation.

- Engineering: Develop, test, and integrate the features planned for the current spiral.

- Evaluation: Review the progress, gather feedback, and plan for the next iteration.

4. Roles and Responsibilities:

Roles:	Responsibility By:
1. Project Manager: - Responsible for overall project planning, coordination, and successful delivery. - Oversees the team and ensures that the project stays on schedule and within budget. - Communicates with stakeholders and gathers feedback to make informed decisions.	Pragathi Rao and Chinmay Mhatre
2. Development Team: **- Lead Developer:** - Takes charge of the technical aspects of the project. - Guides the development team in coding standards, best practices, and architectural decisions. - Ensures the quality of the codebase and adherence to project timelines. **- Software Developers:** - Responsible for writing code, implementing features, and maintaining coding standards. - Collaborates with the testing team to identify and address issues.	Chinmay Mhatre Pragathi Rao

Roles:	Responsibility By:
3. Testing Team: - **Quality Assurance (QA) Lead:** - Manages the testing process and ensures that quality standards are met. - Develops and implements test plans, test cases, and test scripts. - **Testers:** - Execute test cases to identify and report bugs. - Collaborate with the development team to ensure timely bug fixes.	Chinmay Mhatre Pragathi Rao
4. User Interface (UI/UX) Designers: - **Lead Designer:** - Creates the overall design vision for the e-learning platform. - Ensures a user-friendly and visually appealing interface. - **UI/UX Designers:** - Responsible for designing individual screens, navigation, and user interactions. - Collaborates with developers to ensure seamless integration of design elements.	Chinmay Mhatre Pragathi Rao
5. Content Creators: - **Subject Matter Experts (SMEs):** - Contribute expertise in science subjects for content creation. - Collaborate with the development team to ensure accurate representation of educational content. - **Content Writers:** - Develop educational content, including text, multimedia, and interactive elements. - Ensure content aligns with curriculum standards and educational objectives.	Pragathi Rao Pragathi Rao

Roles:	Responsibility By:
6. Customer Support: - Support Representatives: - Provide assistance to users, addressing queries, and troubleshooting issues. - Gather user feedback and communicate it to the development team for continuous improvement.	Pragathi Rao

Outcomes:

CO1: Comprehend process models.

CO2: Describe software planning and management.

CO3: Demonstrate requirements, modeling and design of a system.

Conclusion: (Conclusion to be based on the outcomes achieved)

Thus with the help of this experiment we successfully learned how to Comprehend process models and Describe software planning and management and Demonstrate requirements, modeling and design of a system.

References:

https://www.slideshare.net/YashrajNigam/e-learning-project-report-yashraj-nigam

https://www.overleaf.com/learn/latex/Commands

Books/ Journals/ Websites:

1. Roger S. Pressman, Software Engineering: A practitioners Approach, 7th Edition, McGraw Hill, 2010

2. https://www.sharelatex.com/