Project Phase II

Topic:

QuARS

Submitted by:

Name	**Roll No**
Zainab Eman	22F-3738
Noor Fatima	22F-3634
Imama Kainat	22F-3661

Submitted to: **Date:**

Sir Sajid Anwar **20-10-2023**

Table of Contents

QuARS

The QuARS tool is designed to automate the assessment of NL requirements documents to ensure they are well-expressed, complete, and internally consistent. This automation can save time and reduce the potential for errors in the requirements, ultimately leading to more successful software development projects. It is a valuable asset in the field of requirements engineering, where clear and unambiguous requirements are crucial for the success of any software project.

Metrices for Evaluating Requirements used by QuARS

It appears that the QuARS tool is designed to automate the assessment of natural language (NL) requirements documents for clarity, completeness, and internal consistency. While I don't have specific information about the matrices used by QuARS, I can provide a general idea of the types of evaluation matrices that could be employed to assess requirements, including those for the Learnova system:

1.Clarity Matrix**:**

- **Readability:** Assess the document's readability, including sentence structure, grammar, and overall language clarity.

- **Consistency:** Check for consistent use of terminology, symbols, and abbreviations throughout the document.
- **Ambiguity:** Identify and eliminate ambiguous phrases or words that could lead to multiple interpretations.

2. Completeness Matrix:

- **Requirement Coverage:** Ensure that all necessary requirements are present and that there are no gaps in functionality.
- **Traceability:** Verify that each requirement is traceable to its source (e.g., stakeholder needs, design specifications, etc.).
- **Testability:** Assess whether the requirements are testable and can be validated in practice.

3. Consistency Matrix:

- **Internal Consistency:** Check for conflicts or contradictions between different requirements within the document.
- **External Consistency:** Ensure that the requirements align with external standards, regulations, or other related documents.
- **Cross-Reference Consistency:** Verify that references to other requirements, diagrams, or documents are correct and up to date.

4. Priority and Dependency Matrix:

- **Priority Assessment**: Assign priority levels to requirements to indicate their relative importance.
- **Dependency Evaluation:** Analyze dependencies between requirements and identify critical path requirements.
- **Impact Analysis:** Assess the impact of changes to one requirement on other related requirements.

5.Testability Matrix:

- **Test Case Identification:** Determine the specific test cases that need to be developed to validate each requirement.
- **Test Data Requirements**: Specify the data and conditions necessary for testing each requirement.
- **Acceptance Criteria:** Define the acceptance criteria for each requirement to declare it as met or not met.

6. Change Management Matrix:

- **Change Impact Analysis**: Assess how changes to one requirement may affect other parts of the project.
- **Change Approval Process:** Define the process for approving and implementing requirement changes.
- **Version Control:** Establish a version control mechanism to track changes and revisions to requirements.

These are some of the evaluation matrices that a tool like QuARS may use to assess the quality of requirements documents, ensuring that they are well-expressed, complete, and internally consistent. The tool can help automate these evaluations, thereby saving time and reducing the potential for errors in the requirements, which is essential for the success of software development projects like Learnova. The specific implementation and features of such matrices would depend on the tool's design and capabilities.

System Requirements using QuARS

To generate system requirements for Learnova using a tool like QuARS, you would typically follow a structured process. QuARS is designed to help assess and improve requirements documents, so it can assist in generating clear, complete, and consistent system requirements. Here's a step-by-step guide on how to use QuARS for generating system requirements for Learnova:

Functional Requirements using QuARS:

1. User Authentication and Authorization:

- Users must be able to create accounts and log in securely.
- Teachers should be able to create and manage classes.
- Students must be able to join classes and access learning materials.

2. Personalized Learning Paths:

- The system shall provide tools for teachers to create personalized learning paths for each student based on their needs and abilities.

- Students should be able to view their learning paths and progress.

3. Real-time Feedback:

- Teachers must be able to provide real-time feedback on students' assignments and assessments.
- Students should receive notifications and feedback in a timely manner.

4. Collaboration Tools:

- The system shall offer collaborative features, such as discussion boards and group projects.
- Users should be able to interact with each other and teachers through messaging and chat.

5. Content Accessibility:

- Learning materials, including documents, videos, and assignments, should be accessible from anywhere with an internet connection.
- Content should be organized and easy to search.

6. Assessment and Grading:

- The system must allow teachers to create quizzes and assessments.
- Students should receive automated grades and performance reports.

Non-Functional Requirements using QuARS:

1. Performance:

- The system must respond to user interactions within 2 seconds.
- It should support a concurrent user load of at least 1000 users.

2. Security:

- User data and learning materials must be stored and transmitted securely.
- The system should be resistant to common security threats such as SQL injection and cross-site scripting.

3. Scalability:

- The system must be scalable to accommodate future growth.
- It should handle an increasing number of users and courses without a significant drop in performance.

4. Reliability:

- The system should have a minimum uptime of 99.9%.
- Regular backups and disaster recovery procedures must be in place.

Quality Requirements using QuARS:

1. Usability:

- The user interface should be intuitive and user-friendly.
- User training should not take more than 30 minutes for teachers and students.

2. Compatibility:

- Learnova should be compatible with modern web browsers (Chrome, Firefox, Safari, and Edge).
- It should work on various devices, including desktops, tablets, and smartphones.

3. Maintainability:

- The system should be easy to maintain and update without causing disruptions.
- Code and database changes must be well-documented.

4. Compliance:

- The system should comply with relevant education and data protection regulations, such as FERPA and GDPR.
- Accessibility standards (e.g., WCAG) must be followed for inclusive learning.

QuARS can be a valuable tool for improving the quality of system requirements, ensuring that they meet the needs of the Learnova project, and helping to reduce errors and ambiguities. Remember that while QuARS can assist in the process, it's important to involve subject matter experts, stakeholders, and experienced requirements analysts to ensure the requirements' accuracy and relevance.