Quality Plan

Table of Contents

Contents

Purpose and Scope

Purpose

The purpose of this Software Quality Assurance (SQA) Plan is to establish the goals, processes, and responsibilities required to implement effective quality assurance functions for the Automated Note-Taking and Summarisation project. The Software Quality Assurance Plan provides the framework necessary to ensure a consistent approach to software quality assurance throughout the project life cycle. It defines the approach that will be used by the QAM and Software Quality (SQ) personnel to monitor and assess software development processes and products to provide objective insight into the maturity and quality of the software. The systematic monitoring of products, processes, and services will be evaluated to ensure they meet requirements and comply with policies, standards, and procedures, as well as applicable Institute of Electrical and Electronic Engineers (IEEE) and ISO standards.

Scope

The purpose of SQA is to ensure that the software developed does not deviate from the original intended product. SQA is also concerned to identify any errors, omissions, inconsistencies, and alternatives, enhancements or improvements that can be made at any stage of development.

The project, spearheaded by Magnificent 7, aims to revolutionize the traditional note-taking process in academic lectures by introducing an Automated Note-Taking System. This innovative system leverages advanced speech-to-text technologies provided by Whisper API for accurate transcription of lectures and employs ChatGPT API for the generation of summaries and interactive chatbot functionalities, enhancing the learning experience. The core software components include a sophisticated PostgreSQL database for efficient transcript management, and the system is designed to facilitate automated transcription, searchable and editable transcripts, AI-generated summaries tailored to user preferences, and interactive chatbots for personalized learning support. The intended use of this system is to improve the efficiency, reliability, and accessibility of lecture notes for students, thereby addressing the challenges associated with manual note-taking and its associated costs and limitations.

Reference Documents

- IEEE STD 730-2002, IEEE Standard for Software Quality Assurance Plans (http://standards.ieee.org/reading/ieee/std_public/description/se/730-2002_desc.html)
- ISO IEC 90003:2004 Software Standard (http://praxiom.com/iso-90003.htm)
- Project Plan
- System Requirement Specifications

Management

This section describes the management organizational structure, its roles and responsibilities, and the software quality tasks to be performed.

Management Organisation

The implementation of quality assurance system is the responsibility of the Quality Assurance Manager (QAM).

Project Management

The Project Manager will be responsible for approving:-

- The system requirement specification document
- The overall time scale for the project
- The choice of system development life cycle
- The choice of software development tools and techniques utilised
- The selection of project teams
- The training of project teams

Assurance Management

The QAM provides Project Management with visibility into the processes being used by the software development teams and the quality of the products being built. The QAM maintains a level of independence from the project and the software developers. In support of software quality assurance activities, the QAM has assigned and secured Software Quality personnel from the pool of available SQ trainees to coordinate and conduct the SQ activities for the project and report back results and issues.

Tasks

This section summarizes the tasks (product and process assessments) to be performed during the development of software. These tasks are selected based on the developer's Project Plan and planned deliverables, and identified reviews.

Product Assessments

The following product assessments will be conducted by SQ personnel:

- Automated Note-Taking Process
- Database Utilization: Search Functionality and Insert Functionality
- Summary Generation
- Interactive Chatbots

Process Assessments

The following process assessments will be conducted by SQ personnel:

- Requirement management process
- Change management process

Roles and Responsibilities

This section describes the roles and responsibilities for each assurance person assigned to the Project.

QAM

Responsibilities include, but are not limited to:

- Secure and manage SQ personnel resource levels
- Ensure that SQ personnel have office space and the appropriate tools to conduct SQ activities
- Provide general guidance and direction to the SQ personnel responsible for conducting software quality activities and assessments
- Assist SQ personnel in the resolution of any issues/concerns and/or risks identified as a result of software quality activities
- Escalate any issues/concerns/risks to project management

Software Quality Personnel

Responsibilities include, but are not limited to:

- Develop and maintain the project software quality assurance plan
- Generate and maintain a schedule of software quality assurance activities
- Conduct process and product assessments, as described within this plan
- Identify/report findings, observations, and risks from all software assurance related activities to the QAM

Documents

Purpose

This section identifies the minimum documentation governing the requirements, development, verification, validation, and maintenance of software that falls within the scope of this software quality plan. Each document below shall be assessed (reviewed) by SQ personnel.

Minimum Document Requirements

- System Requirement Specifications

Standards, Practices, Conventions and Metrics

Purpose

This section highlights the standards, practices, quality requirements, and metrics to be applied to ensure a successful software quality program.

Software Quality Programme

These practices and conventions are tools used to ensure a consistent approach to software quality for all programs/projects.

Usability

The system should be user-friendly, with an intuitive interface and easy navigation. Metrics for measuring usability may include user satisfaction surveys, task completion times, and error rates during user interactions.

Reliability

The system must perform consistently and accurately, especially in speech-to-text conversion and chatbot responses. Reliability metrics could encompass system uptime, transcript accuracy rates, and the system's resilience to failures.

Maintainability

Ease of system modification for enhancements or fixes is crucial, given the project's reliance on evolving AI technologies. Maintainability can be gauged through metrics like the time required for updates, frequency of maintenance releases, and adaptability to new requirements.

Efficiency

The system should operate efficiently, with optimized resource use, particularly for real-time processing tasks. Efficiency metrics might include response times for user queries, resource utilization during operations, and bandwidth usage for data uploads and processing.

Standard Metrics

The following standard metrics are the minimum planned metrics that will be collected, reported, and maintained in the area of software quality assurance:

- **Code Quality Metrics**
  - Number of bugs per thousand lines of code: This metric helps in understanding the overall quality of the codebase by measuring the density of bugs.
  - Cyclomatic Complexity: Measures the complexity of the software's structure, helping to assess the risk of modifying the code.
- **Testing Metrics**
  - Test Coverage Percentage: Indicates the percentage of the codebase that is covered by automated tests, ensuring that most functionalities are tested.
  - Defect Detection Percentage: The ratio of defects found during testing to the total number of defects found, indicating the effectiveness of the testing process.
- **Performance Metrics**
  - Response Time: Measures the time taken for the system to respond to a request under a particular load, ensuring the system meets performance requirements.
  - System Downtime: Tracks the total time the system is unavailable or not operational, aiming to minimize downtime in production.
- **User Satisfaction Metrics**
  - Net Promoter Score (NPS): Gauges user satisfaction and loyalty by measuring the likelihood of users to recommend the system to others.
  - Customer Support Tickets: The number of tickets or issues reported by users, indicating areas that may need improvement or are causing problems for users.
- **Maintenance Metrics**
  - Mean Time to Repair (MTTR): The average time taken to fix a fault or bug, indicating the efficiency of the maintenance team.
  - Change Request Turnaround Time: The time taken from receiving a change request to its implementation, measuring the agility of the development process in accommodating changes.
- **Project Management Metrics**
  - Sprint Burn-Down Rate: Measures the completion rate of tasks in a sprint against the planned work, indicating the efficiency and pace of the development team.

Software Reviews

Purpose

This section identifies the number and type of system/subsystem reviews and engineering peer reviews that will be supported by the SQ Personnel. The project milestone chart, and the SQ Personnel resource levels determine the reviews that are supported.

Minimum Software Reviews

For each review, SQ will assess the review products to assure that review packages are being developed according to the specified criteria, the review content is complete, accurate, and of sufficient detail, and Requests for Action are captured, reviewed, and tracked to closure. In addition, SQ will assess the processes used to conduct the reviews to determine if appropriate personnel are in attendance, correct information is presented, entry and exit criteria are met, and appropriate documents are identified for update. The following software reviews will be assessed by SQ:

- Project Plan Review
- Requirements Analysis Review
- Software Design Review
- Test Plan Review
- Acceptance Review

Test

SQ personnel will assure that the test management processes and products are being implemented per Test Plan. This includes all types of testing of software system components as described in the test plan, specifically during integration testing (verification) and acceptance testing (validation). SQ personnel will monitor testing efforts to assure that test schedules are adhered to and maintained to reflect an accurate progression of the testing activities. SQ will assure that tests are conducted using approved test procedures and appropriate test tools, and that test anomalies are identified, documented, addressed, and tracked to closure. In addition, SQ will assure that assumptions, constraints, and test results are accurately recorded to substantiate the requirements verification/validation status. SQ personnel will review post-test execution related artifacts including test reports, test results, problem reports, updated requirements verification matrices, etc.

Problem Reporting and Corrective Action

Software Quality (SQ) personnel are responsible for the generation, tracking, and analysis of assessment findings and observations. This process is facilitated through a centralized Reporting and Corrective Action System, ensuring efficient management and resolution of issues.

Centralized System Location

The system is hosted on WikiMedia. This platform is accessible at http://155.69.100.30/s22324SCELMagnificent7/index.php/Documentation, ensuring that all stakeholders have real-time access to the data.

Communication Protocol

Assessment data and the status of corrective actions are communicated to the Quality Assurance Manager (QAM) and the Project Manager through a structured protocol:

-
  - Weekly Reports: Summarized findings and corrective action progress are compiled into weekly reports, distributed via email.
  - Dashboard Updates: The centralized system includes a dashboard feature, providing real-time updates on problem status and resolution progress. Key personnel have access to this dashboard for an at-a-glance view of quality assurance metrics.
  - Regular Meetings: Bi-weekly meetings are held to discuss outstanding issues, review corrective action plans, and update on problem resolution progress. These meetings ensure that both the QAM and Project Manager are fully informed and can provide input on priority actions.

Action Tracking

Each reported problem is assigned a unique identifier and logged into the system with details including the date reported, problem description, severity level, assigned personnel for resolution, targeted resolution date, and current status. This structured approach ensures that every issue is monitored from identification through to resolution, with accountability and transparency at every stage.

Feedback Loop

A feedback loop is established to review the effectiveness of resolved actions during subsequent assessments. This process ensures continuous improvement and helps prevent the recurrence of similar issues.

Tools, Techniques and Methodologies

SQ personnel will require access to the following:

Software Quality Tools

- Github Link for software assessment
- Wikimedia Link
- Database in Vercel

Media Control

SQ deliverables will be documented in WikiMedia. Deliverables will be in soft copy, with the exception of completed checklists from process and product assessments. See Section 12 for additional details on the collection and retention of key records. Software Quality personnel will request space on the project's secured server for SQ records. This server is password protected and backed up nightly.

Supplier Control

Not applicable for this project

Record Collection, Maintenance, and Retention

SQ personnel will maintain records that document assessments performed on the project. Maintaining these records will provide objective evidence and traceability of assessments performed throughout the project's life cycle. There are two types of records that will be maintained: Hardcopy and Electronic. SQ personnel will maintain electronic or hard copies of all assessment reports and findings. SQ Project folders will contain hardcopies of the assessment work products such as completed checklists, supporting objective evidence, and notes. The table below identifies the record types that will be collected, as well as the Record Custodian and Retention period

Record Title	Record Custodian	Record Retention
SQA Assessments	SQ Personnel	Six Months
SQA Checklists	SQ Personnel	Six Months
Deliverable Defects	SQ Personnel	Six Months

Training

SQ personnel have fundamental knowledge in the following areas through prior experience, training, or certification in methodologies, processes, and standards:

- Audits and Reviews (Assessments)
- Risk Management
- Software Assurance
- Configuration Management
- Software Engineering
- ISO 9001, ISO 9000-3
- CMMI
- Verification and Validation

== Risk Management SQ personnel will assess the project's risk management process and participate in weekly risk management meetings and report any software risks to the QAM and the project manager.

SQA Plan Change Procedure and History

SQ personnel are responsible for the maintenance of this plan. It is expected that this plan will be updated throughout the life cycle to reflect any changes in support levels and SQ activities. Proposed changes shall be submitted to the Quality Assurance Manager (QAM), along with supportive material justifying the proposed change.

Retrieved from "http://155.69.100.30/s22324SCELMagnificent7/index.php?title=Quality_Plan&oldid=128"

This page was last edited on 28 February 2024, at 22:35.