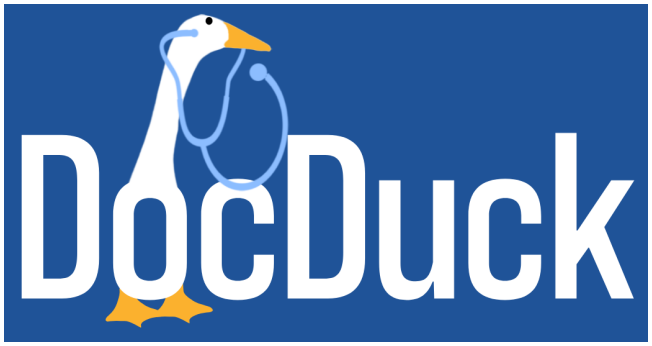

DocDuck Quality Assurance Manual

SWEng Group 1

November 30, 2023

Contents

Functional Spec Version	Release Date	Changes	Contributors
0.1.0	14/11/23	Initial Version. Created Sections: 1, 2, 4, 5,	wab515 jrb617 nc1196 ttc515 zm926 htsm500 js2140 rw1834 lw2380
0.2.0	30/11/23	Section 3 Added. Table format standardised across roles. clarification of manager QA metrics. unnecessary vision statement removed, company profile adjusted to be in line with QA manual standards. Modified the design phase to match an agile method apposed to waterfall. Cleaned up structure.**Write here if you change something**	jrb617 htsm500 js2140 lw2380

1 Introduction

1.1 Company Profile

The company was formed to provide assistance to companies with a maintenance engineer division. the major business focus for the company is to improve overall clarity for the maintenance engineering sector. Our companies emphasis is on affordability for the growing business as well as continued support with additional future features.

Our group is a team of experienced developers with backgrounds in many different avenues combining together to provide a high quality product with a variety of features and accessible to all skill levels.

2 Roles and Responsibilities

2.1 Organisational Structure

To ensure the product can be delivered at a high quality, on time and on budget. The company structure has been divided into a series of roles for the different members. These roles break down into 2 areas, organisation and software. The organisational roles lead the running of the company and design of the product. The software roles lead the implementation and testing.
Each role has at least one deputy assigned to it. They are to be familiar with the responsibilities of the role and able to take over the duties should the main person be unable to through illness etc. They also take up smaller tasks and assist with larger ones to distribute labour more evenly amongst the company.
Communication between team members is achieved through weekly team meetings and reviews. Additional progress reviews are also conducted prior to deliverable submission.
An illustration of the roles and the interactions between them is shown below:

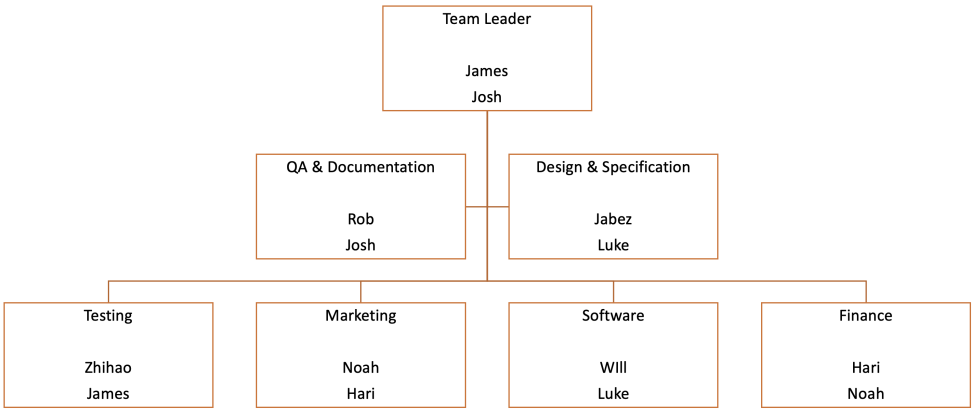

Figure 1: Role Hierarchy

2.2 Project Manager

Group member: James Stevenson

2.2.1 Role Description

The project manager is responsible for the overall project, Ensuring the project hits its desired goals in its desired time frames as well as ensuring that all members of the group have adequate roles and contributions to the project as a whole. The project manager should be in regular contact with each role to ensure that their progress is smooth, any issues are dealt with as well as ensuring that any assets required can be provided. Accomplishing this involves several tasks including

- Arranging regular meetings as well as extra meetings if required.
- Making sure the agenda is known and set for the future meetings.
- Assigning roles that complement each individual members skill set.
- Attaching secondary roles for each member to ensure that the role can still be handled in-case of emergency and that the primary role member stays on track.
- Ensuring a healthy working environment where each member feels listened to.
- Making sure that each member feels they have made an appropriate contribution to the project as a whole.
- Ensuring contingencies are in-place in-case unexpected emergencies occur during development.
- Having appropriate backup solutions to problems.
- Organise a realistic project plan and time schedule.
- Liaise with the customer to ensure the product meets with their expectations.
- Perform the final check on deliverables before approval.
- Communicate with the finance manager to keep within budget for the project.

2.2.2 Risk Management

Risk Description	Likelihood (1-5)	Impact (1-5)	Priority (1-25)	Mitigation
Team Members clashing	2	4	8	Team members feel comfortable to voice any and all concerns at meetings and solution is found
Missing internal deadlines	3	5	15	have constant communication with members of each section of the project and find solutions to problems before deadline is reached such as swapping members of the team at fault or adding a new team member to increase productivity
Team Members prolonged absence	4	2	8	Each role has a backup member capable of picking up where the other team mate left off, if absence lasts even longer multiple members may be assigned to that role.
Failure to implement feature	3	4	12	Ensure a rollback state is created before each new features implementation, if the feature is mandatory check if there is a simpler alternative feature that could provide similar results.

Table 1: Project Manager potential risks and mitigations

2.2.3 QA Metrics

Responsibility	Metric	Purpose
Customer communication:	Satisfies customer standards	Consistent reviews and feedback from customer over product and documentation. ask for customer to prove ratings out of 5 stars and customer reviews
project tracking:	Smooth Project Pace	Set realistic internal deadlines and gantt charts, review and monitor how close we lineup to the goals set.
healthy workspace	Developers are not crunched	Spread work out evenly over time given and ensure devs follow the timetable set.
Workspread	Workload is portioned evenly	Have consistent meetings to listen to developers thoughts and feelings on their workload and if its feasible in a given time.

Table 2: Project Manager QA Metrics

2.3 QA and Documentation Manager

Group member: Rob Walker

2.3.1 Role Description

Quality Assurance is a set of procedures that ensures the software created by the company meets the requirements needed and to an acceptable quality. The QA and Documentation manager is responsible for creating these processes are adhered to for the life cycle of the software to ensure quality, monitoring these processes, creating a document management system and reviewing documents to ensure they adhere to the requirements. Ensuring the processes created are adhered to requires regular communication and review with all members of the team.

Communication with members of the team will be done through group meetings, review meetings and documents created for team members to help them follow the requirements for all documents through the company. These documents will be designed to allow the QA and Document manager to communicate and check that requirements are followed. If change is needed in and QA documents or processes, approval of the changes will be required approval by the QA and Documentation manger and the Project Manager. An updated document/process will be created by the QA and Documentation manager and will be sent out to all team members with the change clearly communicated.

The tasks for the QA and Documentation manager are:

- Create QA processes and make clear to all members what should be followed, who should follow it, and the quality needed.
- Define the quality expected from documents in the company alongside the Project Manager.
- Create QA metrics for quality to be measured from throughout development.
- Review the use of the QA processes throughout development and carry out review meetings to ensure quality.
- Specify deadlines and how different types of documents should be handled in the company.
- Create document guidelines to be followed by all members of the team.
- Ensure the document guidelines are followed so the standard of documentation is up to high.
- Archive all documents throughout the project.
- Track all updates on documents and releases.
- Monitor deadlines for documents to make sure they are delivered.
- Ensure the latest versions of documents are being used.
- Ensure all meetings are appropriately held and documented.

2.3.2 Risk Management

Risk Description	Likelihood (1-5)	Impact (1-5)	Priority (1-25)	Mitigation
Failure to carry out a QA process	2	2	4	Have regular review meetings with team members and keep QA processes appropriate and up to date.
Missing or lost documents	1	5	5	Keep online and offline backups of all documents when they are created or updated
Module or Document clashing/incompatibility	2	4	8	Make sure the most up to date documents are used and all team members are aware of any changes
QA metric not met	2	2	4	Monitor and review all team members and their adherence to the QA metrics and processes.

Table 3: QA and Documentation potential Risks and Mitigations

2.3.3 QA Metrics

Responsibility	Metric	Purpose
Documentation:	Documents delivered	The number of documents delivered against the number of documents expected.
Deadlines:	Deadlines met	The number of submissions against the number of deadlines set.
Meeting QAs:	QA Metric Collection	The number of metrics met against the number of metrics expected.

Table 4: QA and Documentation QA Metrics

2.4 Design and Specification Manager

Group member: Jabez Cheung

2.4.1 Role Description

The design & specification manager is responsible for providing a design of the product with specifications and features fulfilling the customer's needs. Seamless communication with managers is of paramount importance to keep track of the progress with all of the designed/requested features included with high-quality standards, meeting client's expectations.

- Read through the client's requirement statement and design a solution within the given budget and time.
- Produce the product specification by communicating with stakeholders including the client and the project team.
- Clearly define functionalities/performance criteria.
- Analyze the test results report and make sure all of the designed features meet the criteria and quality standard fulfilling client and end user's expectations.
- Communicate with other managers to keep track of the progress.
- Identify risks associated with the design or specification of the product and avoid over budget or time.
- Document the design and specification processes providing details of updates/changes made in each iteration.
- In association with the test manager, ensure the features requested by the client are in good working order and meet high-quality standards. Allocate group members to solve challenges and improve the quality of the product if needed.
- In association with the GUI design manager, ensure the product is easy to use and the services are easy to access, following Human Interface Guidelines [1].
- Seek feedback from the client for each version released identifying room for improvement.

2.4.2 Risk Management

Risk Description	**Likelihood (1-5)**	**Impact (1-5)**	**Priority (1-25)**	**Mitigation**
Delay in finalizing the product design	2	5	10	Identify the reason for the delay. Consider reordering by prioritizing the tasks, evaluate the time lost and report to the project manager.
Unfulfillment of quality standard & specification	2	4	8	Work with the test manager to identify the reason for unfulfillment and the challenges involved, allocate members to improve the corresponding part of the product, evaluate the time needed and report to the project manager.

Table 5: Design & specification potential risks and solutions

2.4.3 QA Metrics

Responsibility	Metric	Purpose
Resource Management:	Cost of adding certain features to the product	resources required to complete a certain task, including time, human power, tools, etc.
Specification Management	Suitability of design	Is the design fulfilling the features and specifications requested by the client meeting high-quality standards without major defects.
Client Feedback	Feedback from the client	Amount of elements being approved or to be improved.

Table 6: Design & specification QA metrics and measurement

2.5 Testing and Integration Manager

Group member: Zhihao Ma

2.5.1 Role Description

The Testing and Integration Manager leads the efforts to maintain high quality and reliability of software releases. This role encompasses strategic planning and execution of testing and integration activities. Responsibilities include:

- Strategizing and executing a comprehensive test plan to ensure the robustness and performance of software before deployment.
- Managing the integration of new software with existing systems, ensuring minimal disruption and maintaining system integrity.
- Overseeing the creation, maintenance, and evaluation of automated test frameworks to facilitate continuous integration and deployment processes.
- Collaborating with cross-functional teams, including development, operations, and product management, to align testing strategies with business objectives.
- Ensuring that all testing and integration activities are conducted in accordance with regulatory standards and company policies.
- Conducting risk assessments for testing and integration processes, implementing mitigation plans to address identified risks.
- Providing leadership and guidance to the testing team, fostering a culture of quality and continuous improvement.
- Analyzing test results, reporting to stakeholders on software quality, and making recommendations for improvements.
- Keeping abreast of new testing tools and strategies, integrating innovative technologies and practices to enhance the testing and integration life-cycle.

2.5.2 Risk Management

Risk Description	Likelihood (1-5)	Impact (1-5)	Priority (1-25)	Mitigation
Inadequate test coverage	2	4	8	Implement test coverage tools and regularly review test cases.
Integration delays	3	4	12	Promote continuous integration and set clear milestones.

Table 7: Testing and Integration Potential Risks and Mitigations

2.5.3 QA Metrics

Responsibility	Metric	Purpose
Integration:	Integration success rate	Percentage of successful integrations per release cycle.
Release Errors:	Defect escape rate	Number of issues found post-release versus pre-release.

Table 8: Testing and Integration QA Metrics

2.6 Finance Manager

Group member: Hari Mamman

2.6.1 Role Description

The finance manager is responsible for handling all financial and accounting affairs. They should be in regular contact with the project manager to ensure that the group is kept on budget and expenses are being reduced where possible. Also, they should be in communication with the marketing leader in order to help produce sale demonstrations. The finance manager tasks should include:

- Managing the company's financial accounting, monitoring and reporting systems
- managing the company's budget
- providing and interpreting financial information
- work with the marketing manager to ensure there are appropriate contracts across groups
- work with the project manager to make sure the company receives financial backing
- formulate strategic and long-term business plans
- produce accurate financial reports to specific deadlines

2.6.2 Risk Management

Risk Description	**Likelihood (1-5)**	**Impact (1-5)**	**Priority (1-25)**	**Mitigation**
Poor Budgeting	2	4	8	Monitor at regular intervals and adjust as needed.
risks due to insufficient liquid assets	2	2	4	Monitor and stick to an accurate budget, ensuring that there are plans in place for sudden shocks.
Ineffective communication of financial information	1	2	2	Ensure clear and transparent communication and provide accurate and timely finance reports.

Table 9: Finance Potential Risks and Mitigations

2.6.3 QA Metrics

Responsibility	**Metric**	**Purpose**
Reporting Accuracy:	Percentage of error-free financial reports	Ensure accuracy of financial information when being presented to financial backers.
Budgeting:	Percentage variance between budgeted and actual expenses	Monitor and control budgetary performance, identifying places where costs can be saved.

Table 10: Finance Manager QA Metrics

2.7 Software Manager

Group Member: William Betteridge

2.7.1 Role Description

The software manager is responsible for planning, designing and coordinating the implementation of the software components of the project. They should have a full understanding of the specifications and the requirements of the project in order to develop the desired product. Also, they should be in regular communication with the QA and Documentation manager to ensure the product meets the specification and the quality of the software design is to the requirements. They should also be in communication with the Testing and Integration Manager to ensure that the product is thoroughly tested and is suitably integrated into any design flows and project timelines that are being worked on. The Software Manager tasks should include:

- Develop a clear overall software design based on the specifications and the customer requirements.
- Analyse the functional specification to determine necessary specific features to be programmed in Java, and split it up into multiple sections to be distributed to other group members.
  - Identify and decide on the interfaces, libraries, objects, data encapsulation and hierarchy that will be used.
- Work with the Design and Specification Manager to decide on the methods and techniques to be used in the design phase.
- Monitor the process of the coding and its implementation.
- Ensure that the design standards are followed in order to remain consistent between different sections of code written amongst the project group.

2.7.2 Risk Management

Risk Description	**Likelihood (1-5)**	**Impact (1-5)**	**Priority (1-25)**	**Mitigation**
Code failing to work	5	3	15	Implement test driven development to reduce unsolvable errors.
Code from group members not compatible	3	3	9	Make sure there is good quality version control and all group members agree on standards when writing code
Delay when finishing modules of code to be integrated	3	4	12	Keep track of all tasks and agree on temporary deadlines prior to final deadlines to allow for time to finish sections

Table 11: Software Potential Risks and Mitigations

2.7.3 QA Metrics

Responsibility	Metric	Purpose
Time Management	Time spent coding vs time planned	Ensure delivery of software is within plan by keeping track of total time spent on work
Good coding practice	Well written code with comments and efficient design	Consider lines of comments against lines of code. Use of standardised coding practices
Errors	Compilation errors and uncaught exceptions	Produce a compilation report to indicate the number of errors during compilation and whether code is error-free and tested.

Table 12: Software QA Metrics

2.8 Marketing Manager

Group Member: Noah Carter

2.8.1 Role Description

The role of the marketing manager in engineering application development is imperative to the success of the business. This can be accomplished in many ways, but the main idea is to stay ahead of the curve and fill any gaps in the market by having a dynamic marketing plan. They should be in regular contact with all managers in the group to ensure all marketing materials and campaigns can be kept up to date with regards to the application development.

Some of the key roles of the marketing manager include:

- Market Research: conducting market research to identify any new potential consumers and businesses by understanding their needs and preferences. The marketing manager should stay informed on current industry trends and conduct market studies to identify any gaps in the market. Market Research also includes accessibility by making sure thorough research is completed in this department and ensuring our app is made as accessible as possible.
- Defined Target Audience: the target audience should be clearly defined for the product.
- Collaborating closely with the rest of the team to oversee and be aware of any emerging developments.
- Overseeing and creating strategic launch campaigns for the product to maximise market impact.
- Ensuring the implementation of effective branded marketing communications across various platforms including the company website, any print communications, and advertising.
- Overseeing the management of media and marketing personnel as well as navigate relationships with any external PR agencies.
- Evaluating the effectiveness of any marketing strategies executed.
- Working with the finance manager to explore appropriate business models included product pricing.

2.8.2 Risk Management

Risk Description	Likelihood (1-5)	Impact (1-5)	Priority (1-25)	Mitigation
Market Access	4	4	16	Gain authorisation from certification bodies as required. Check whether end users are those with purchasing authority, model selling strategy accordingly.
Pricing issues related to core products	3	2	6	Find trusted sources with appropriate pricing.
Competition	5	4	20	Ensure quality of product. Produce appropriate advertising campaign. Provide excellent after-sales service.

Table 13: Marketing Potential Risks and Mitigations

2.8.3 QA Metrics

Responsibility	Metric	Purpose
User Satisfaction:	Collect user feedback and satisfaction surveys.	Evaluate the overall satisfaction of end-users with the visual design and user interface. Feedback can provide insights into the effectiveness of the design in meeting user expectations.
Brand Awareness:	Market survey alongside customer feedback.	Evaluate results from market survey, as well as asking consumers how they discovered our product to test campaign effectiveness.
Share of the market:	Market research.	Research the current market and evaluate the products standing in comparison to the competitive and ever-changing landscape of product development.

Table 14: Marketing Potential Risks and Mitigations

2.9 Communications Manager

Group Member: Josh Bradley

2.9.1 Role Description

Effective communication amongst a group is paramount to success. The Communications Manager is responsible for handling the overall communication amongst and outside the group. They coordinate communications amongst the other Managers to enable the fluid transfer of information. They liaise with other groups to enable effective cooperation in Project Wide Standards and inter-group contracts.

The key tasks of the Communications Manager include:

- Overseeing internal communications within the group to ensure effective communication strategies.
- Ensuring all members of the group are up to date in all required areas: deliverables, expectations, due dates, etc.
- Overseeing all inter-group communications. Working with the Software Manager and Project Manager to develop project wide standards and inter-group contracts.
- Booking rooms for team meetings and notifying group members of their time and location.
- Disseminating meeting notes to members not at that meeting
- Ensuring all required information is readily available to group members who need it.

2.9.2 Risk Management

Risk Description	Likelihood (1-5)	Impact (1-5)	Priority (1-25)	Mitigation
Ineffective internal communications	3	2	6	Ask for clarification if unclear. Ensure that all members are communicating their progress via meetings and reviews.
Members unsure of their required work	3	4	12	Work with QA and Documentation Manager to create standardised and readily available documentation such as Gantt charts and meeting minutes
Groups unable to agree on project wide standard	2	5	10	Mediate with Project Manager, Software Manager and their external counterparts to reach a solution which will work for all parties
Groups not fulfilling their contracts	2	5	10	Mediate with Project Manager their external counterpart to identify the issue and how to solve it and adjust time scheduling appropriately
Group Members unable to make meeting	4	2	8	Ensure Deputy is able to fill in if required, move meeting format to hybrid if appropriate

Table 15: Communications Potential Risks and Mitigations

2.9.3 QA Metrics

Responsibility	Metric	Purpose
Group Communication	Group Members attending meetings and completing appropriate work	Evaluate internal communication, whether meeting and work information is being passed on appropriately
Dissemination of responsibilities	Quality of Gantt chart, meeting minutes Monday.com	Ensure work expectations are available and obvious to group members
Inter-Group communication	Meetings with other groups	Ensuring that appropriate information is passed between groups to enable affective cooperation

Table 16: Communications QA Metrics

2.10 Graphic Designer & GUI Developer

Group Member: Luke Warbey

2.10.1 Role Description

The Graphic Designer & GUI Developer in the project plays a pivotal role in shaping the visual identity and user interface (UI) of the software. This role requires a unique blend of artistic creativity and technical proficiency. Key responsibilities include:

- **User-Centric Design:** Develop a deep understanding of user personas and incorporate user-centered design principles to create interfaces that are not only aesthetically pleasing but also highly intuitive and user-friendly.
- **Wireframing and Mockups:** Collaborate closely with the team to translate conceptual ideas into tangible wireframes and design mockups. These artifacts serve as the foundation for the development of the software's graphical elements.
- **UI Prototyping:** Utilize prototyping tools to create interactive prototypes, allowing stakeholders and end-users to visualize the flow and functionality of the UI. Iterate on prototypes based on feedback received during design reviews.
- **Visual Asset Creation:** Craft and optimize graphical assets, including icons, images, and other visual elements, ensuring they align with the overall design vision and enhance the user experience.
- **Design Consistency:** Maintain consistency in design elements throughout the software to create a cohesive and unified look and feel. This includes adherence to established design patterns and style guides.
- **Responsive Design:** Implement responsive design principles to ensure the software's UI is accessible and visually appealing across various devices and screen sizes.
- **Collaboration with Development:** Work closely with the development team to ensure the feasibility of design implementations. Provide necessary assets and guidance to facilitate a smooth integration of graphical elements into the software.
- **Accessibility & Coding Standard Considerations:** Integrate accessibility features into the design process, ensuring that the UI is inclusive and complies with accessibility standards and is formatted professionally [2].
- **Stay Updated on Design Trends:** Continuously stay informed about industry trends and emerging design technologies to bring innovative and modern design solutions to the project.
- **Documentation:** Maintain comprehensive documentation of design decisions, rationale, and guidelines to facilitate knowledge transfer and collaboration within the team.

2.10.2 Risk Management

Risk Description	Likelihood (1-5)	Impact (1-5)	Priority (1-25)	Mitigation
Compatibility Issues: The graphical elements created may not integrate seamlessly with the GUI, leading to compatibility issues.	2	4	8	Collaborate with Software Manager and conduct regular integration testing to identify and address compatibility issues early in the development cycle.
Performances Bottlenecks: Complex graphical elements can contribute to performance bottlenecks in GUI.	2	2	4	Optimize graphical assets for performance. Implement caching strategies and conduct performance testing to identify and resolve bottlenecks. Keep a balance between visual richness and system performance.
Maintainability Challenges: Poorly structured code for GUI development can result in difficulties in maintaining and updating the software.	3	4	12	Enforce coding standards and best practices for GUI development. Document the codebase thoroughly to facilitate easier maintenance. Conduct code reviews to ensure adherence to established guidelines.
Security Vulnerabilities: Inadequate handling of graphical elements in the GUI code may introduce security vulnerabilities.	3	5	15	Enforce coding standards and best practices for GUI development. Document the codebase thoroughly to facilitate easier maintenance. Conduct code reviews to ensure adherence to established guidelines.
Misalignment with Brand Identity: There is a risk of the graphic designer creating visual elements that do not align with the established brand identity and become inconsistent with GUI design. leading to inconsistency.	2	4	4	Conduct regular brand guideline reviews, ensuring the graphic/GUI designer is familiar with and adheres to the brand's visual standards. Schedule periodic meetings with the team to clarify any ambiguities.

Table 17: Graphic Designer & GUI Developer Potential Risks and Mitigations

2.10.3 QA Metrics

Responsibility	Metric	Purpose
User Satisfaction:	Collect user feedback and satisfaction surveys.	Evaluate the overall satisfaction of end-users with the visual design and user interface. Feedback can provide insights into the effectiveness of the design in meeting user expectations.
Usability Testing Results:	Analyze results from usability testing sessions.	Ensure that the graphic design aligns with established design guidelines and branding standards. Identify and rectify any deviations.
GUI Related Bug Reports:	Monitor and analyze design-related bug reports.	Track the number and severity of bugs related to graphical elements. Address and resolve issues promptly to enhance the overall quality of the user interface.
Timeline Adherence:	track the adherence of design tasks to the project timeline.	Ensure timely delivery of graphical assets, preventing delays in the development and testing phases.
Prototype Effectiveness:	Analyze feedback on interactive prototypes.	Evaluate the effectiveness of interactive prototypes in communicating the intended flow and behavior of the user interface. Use feedback for continuous improvement.
Documentation Completeness:	Review the comprehensiveness of design documentation.	Ensure that design decisions, rationale, and guidelines are well-documented for knowledge transfer and collaboration within the team.

Table 18: Graphic Designer & GUI Developer QA Metrics

3 Deliverables

Deliverable	Producer*	Recipient*	Due
Technical Requirements Document	Client	Project Manager, Design and Specification Manager, QA and Documentation Manager	Start of Project
Functional Specification	All Managers	Client	End of Design Phase
Quality Assurance Manual	All Managers	Client	End of Design Phase
Project Wide Standard (PWS)	Project Manager, Software Manager, Communications Manager	Client	End of Design Phase
Design Documentation	QA and Documentation Manager	Client	End of Design Phase
User Manual	All Managers	Client	With initial release and updated with subsequent releases
Inter-Group Contracts	Project Manager, Software Manager, Communications Manager	Other group Project Managers, Client	End of Design Phase
Test and Integration Plans	Testing and Integration Manager	Testing Team	Before Implementation Phase
Test and Integration Reports	Testing Team	Testing and Implementation Manager	Throughout the Project
Financial Business Plan	Financial Manager	Client	End of Design Phase
Source Code	Software Manager	Client	With Each Release
Code	All Software Developers	Software Manager	Specific Milestones
Meeting Minutes	QA and Documentation Manager	Project Manager	After Each Meeting
Example Multimedia Presentation	All Managers	Client	End of Design Phase
HTML Product Tour	All Managers	Client	End of Design Phase
Weekly Contribution Reports	All Members	Project Manager	Throughout the Project
Weekly Time Sheets	All Members	Finance Manager	Throughout the Project
Budget Analysis Reports (Costs, expenses and resources)	Finance Manager	Project Manager	Throughout the Project

Table 19: Project Deliverables

*Where a specific Manager is responsible, their deputy is likewise responsible

4 Project Management Methodology

4.1 Requirements and Specifications Phase

The project starts with a statement by the customer laying out what is required from the product. The group then analyses the requirements and generates a series of ideas for the software to produce. This goes through a series of development meetings to refine the concepts and narrow down the most effective and practicable. These are then discussed with the customer to ensure that they meet their requirements and feasibility. The individual members whose ideas these are then finally develop them to present to the group for deliberation. The group then votes on which idea will be moved forward with as the final product and elect the project manager who reports this to the customer. This is then further developed by creating the final feature list and separating it into individual releases. To summarise:

- The customer provides the requirements for what the project must do
- The group generates ideas to meet this requirements
- The group meets with the customer to discuss and narrow down the ideas
- The group votes on the final idea and Project Manager.
- The Project Manager reports to the customer and discusses any changes required.
- The Project Manager, Design and Specification Manager and QA and Documentation Manager, then generate the final feature and release list.

4.2 Design Phase

For the DocDuck project, the agile software development life cycle (SDLC) is critical to the Design Phase. Agile takes an iterative approach, constantly refining the design throughout each sprint, as opposed to the waterfall design. Our objective is to convert given specifications into a scalable, adaptable design that may change as demands do. Our first priority is developing a software solution that is efficient and responsive to both new and evolving functional and non-functional requirements. We can include user feedback throughout this agile design phase, adapt to shifting priorities, and iteratively optimise the design to achieve project objectives.

4.2.1 Design Objectives

The primary objectives of the Design Phase are as follows:

1. Requirements Translation: Translate the documented requirements into a detailed design that serves as a blueprint for implementation.
2. Scalability and Performance: Ensure the design supports scalability requirements and addresses performance considerations for optimal execution.
3. Modularity and Reusability: Promote modularity and reusability of code components to enhance maintainability and reduce development time.
4. Security Considerations: Integrate security features and considerations into the design to safeguard sensitive information and ensure data integrity.
5. User Experience (UX) Design: Collaborate with UX and Graphic Design teams to create an intuitive and visually appealing user interface.
6. Compliance with Standards: Adhere to Java SE-11 standards and best practices to ensure code quality and compatibility.

4.2.2 Architectural Design

In this activity, the system architecture is defined, detailing the structure and organization of components. Key considerations include:

- Component Identification: Identify and define the main components/modules of the system.
- Interaction Design: Specify the interactions between different components and modules.
- Data Architecture: Design the data model and database schema.
- Technology Stack: Determine the technologies and frameworks to be used.

4.2.3 Detailed Design

The detailed design activity involves expanding on the architectural design, providing a more granular view of the system:

- Class and Interface Design: Define classes, interfaces, and their relationships.
- Algorithm Design: Specify algorithms and data structures.
- Database Design: Develop detailed database schemas and relationships.
- Security Design: Incorporate security mechanisms and access controls.

4.2.4 User Interface (UI) Design

Collaborating with the Graphic Design and UX teams, the UI design activity focuses on creating an engaging and user-friendly interface:

- Wireframing: Create wireframes to outline the structure and layout of the user interface.
- Prototyping: Develop interactive prototypes for user feedback and validation.
- Graphic Design Integration: Integrate graphical elements and branding into the UI design.
- Accessibility: Ensure the user interface complies with accessibility standards.

4.2.5 Quality Assurance in the Design Phase

Quality Assurance activities during the Design Phase are crucial to ensure that the design meets the specified requirements and aligns with established standards:

4.2.6 Design Reviews

Conduct regular design reviews to assess the completeness, correctness, and consistency of the design artifacts. Key stakeholders, including developers, testers, and project managers, should participate in these reviews.

4.2.7 Compliance Checks

Verify the design's compliance with Java SE-11 standards, coding guidelines, and organizational best practices. Automated tools and manual inspections should be employed to identify and address any non-compliance issues.

4.2.8 Risk Assessment

Identify potential design-related risks and assess their impact on project goals. Develop mitigation strategies and contingency plans to address these risks effectively.

4.2.9 Documentation Standards

Enforce documentation standards for design artifacts, including architecture documents, detailed design specifications, and UI design documentation. Well-documented designs facilitate knowledge transfer and future maintenance.

4.2.10 Design Phase Deliverables

The Design Phase results in several key deliverables that serve as the foundation for subsequent development and testing activities:

- Architectural Design Document: Describes the system architecture, component interactions, and data architecture.
- Detailed Design Specifications: Provide detailed information on class structures, algorithms, and database designs.
- User Interface Design Documents: Include wireframes, prototypes, and graphic design specifications.
- Compliance and Standards Reports: Document the results of compliance checks and adherence to Java SE-11 standards.
- Risk Log: Lists identified risks, their impact, likelihood, and mitigation strategies.

4.3 Implementation Phase

By analysing and following the design specifications, the software manager can consider and decide upon appropriate time schedules and resource requirements for each of the sections within the specification. This will allow for tasks to be assigned to members of the group to work on their own sections. There will be frequent meetings to ensure that each section of software is compliant. The testing and integration manager will make sure there is consistent version control as well as frequent and succinct testing that thoroughly confirms that each section of code is well made and works as intended to the design.

During the implementation phase, the software manager will be responsible for the control of the whole implementation whilst communicating with the Design and Specification manager and the Project manager to make sure progress is smooth and meeting the required specification in due course. The Finance manager will monitor the budget and communicate with the Software manager to track the progress and time spent working across the team. This will be essential for tracking and updating the records for the overall cost of the project, which in turn will allow a detailed product cost to be calculated for clients.

Upon the completion of each individual component, the software manager is responsible for testing the component and delivering it to the Testing and Integration manager to formally report all outcomes and make sure each component is ready for integrating into a final product.

Below is an overview of the tasks included:

- Software team works on implementing the design specifications given by the software manager
- Software manager overlooks production and keeps track of progress
- Design and Specification manager and Project manager communication with Software manager
- Testing and Integration manager to receive finalised components to the code and report outcomes from testing
- Finance manager is responsible for monitoring the budget spent and time taken for the team to complete the implementation

4.4 Testing and Integration Phase

4.4.1 Introduction

The Testing and Integration Phase is pivotal in the software development lifecycle, especially for Java projects in Eclipse. This phase is managed by the Testing and Integration Manager, who ensures that all testing and integration processes are systematically carried out.

4.4.2 Component Testing

- Individual Java components (classes and methods) are rigorously tested.
- White box testing is employed for an in-depth examination of the internal software logic.
- Final testing of each component should be conducted by team members not involved in its initial development to ensure objective quality assessment.

4.4.3 Integration Strategy

- A bottom-up approach to integration is recommended, beginning with smaller units and progressing to larger integrated systems.
- Integration testing ensures that combined Java components work together seamlessly.

4.4.4 Testing Environment

- Software simulation is initially conducted within the Eclipse IDE.
- Sideways integration is used for integrating modules at the same hierarchical level.

4.4.5 Error Management and Reporting

- All defects and issues are systematically logged and tracked.
- Regression testing is crucial following any changes to ensure the stability of existing functionalities.
- Comprehensive reports on testing outcomes are prepared by the Testing and Integration Manager.

4.4.6 Schedule and Collaboration

- Strict adherence to the testing and integration schedule is essential for all team members.
- Immediate reporting of any issues encountered during testing is required.

4.4.7 Conclusion

- The software undergoes final verification in simulated environments, ensuring its robustness and reliability.
- Continuous feedback and iterative improvements are critical for the software's development.

4.5 Quality Auditing Reviews

Producing a high-quality product necessitates an equally high-quality level of production process; the methods utilized during the design and implementation phases are specifically crafted to reach these standards. This is also achieved through careful supervision of the QA metric collections. It is imperative that a review session takes place when wanting to progress onto the next phase of the project, as passing a review session is the only way to progress. These are often conducted by peers, stakeholders, or experts who have the experience and knowledge to provide feedback, suggestions and constructive criticism, however these members are a must:

- Supplier: The individual whose product is to be examined
- QA and Documentation Manager: Establishes and oversees the overall review process, documents the results of the review session
- Inspectors: Typically 3 members: one from the previous phase, one from the same team as the supplier, and one from the next phase, to ensure the product meets the necessary requirements for upcoming tasks.
- Customer: Not directly necessary, but is often the case in particular when the deliverable is directly related to the final product

Within the review session, the team will make use of numerous resources, including the QA metrics linked to the product, in order to look for any errors or inconsistencies. The QA and Documentation manager will then create the Quality Auditing Report and submit it to the Project Manager. If the review session is failed, the product is unable to progress to the next phase until the errors are fixed.

5 To Be Completed

- ~~Standardise Table Format~~
- ~~Label Tables/Figures Appropriately~~
- Update Role Diagram with GUI and Communications
- ~~Add deliverables~~
- General formatting/GrammAr
- Stuart Relevant feedback

References

[1] "Apple human interface guidelines." [Online]. Available: https://developer.apple.com/design/human-interface-guidelines/

[2] A. Korunović and S. Vlajić, "An example of integration of java gui desktop technologies using the abstract factory pattern for education purposes," ETF Journal of Electrical Engineering, vol. 29 No. 1, 2023. [Online]. Available: https://jee.ucg.ac.me/index.php/files/article/view/241