Cover sheet for submission of



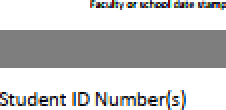
work for assessment



Unit n ame Managing IT Ptejects

unit coae swe3o3 to Assi me«t « o. Task ose oue a «Ie N/A Name or feet direr/t eacher Dr. Thomas Hang

Tutor/marker’s name Dr Thomas Hang





**Name:**

**Student ID:**

**Task 05P: Software Quality and Definition of Done**

**PROJECT PROPOSAL**

**BUILDING AN INNOVATIVE B2B2C E-COMMERCE PLATFORM**

1. **QUALITY MANAGEMENT**
   1. **What is Quality?**

In the context of this E-commerce project, a ‘quality’ platform is considered as a final platform that can satisfy all requirements collected from the clients. These requirements are summarized below:

* **Technical requirements**
  + Appropriate framework and the hosting server
  + Responsive and friendly user interface
  + Support payment gateways and all related management (order, shipping, inventory, etc.)
  + Report and analytics tool is integrated
* **Functional Requirements**

All core functionalities of an e-commerce platform can work efficiently and do not have any severe errors that corrupt the platform. Core functionalities include:

* + User registration and account management
  + Product management and displaying
  + Order management and tracking
  + Review/Rating system
  + Feedback mechanism
  + Notification and Alert settings
* **Non-functional Requirements**
  + Have user-friendly UX/UI
  + Integrate appropriate security techniques and comply with data protection regulations
  + Integrate with third-party payment gateways, shipping vendors, and analytics tools and have the API to support them
  + High scalability

# How to measure quality?

To assess the quality of e-commerce platform development, we can use a Definition of Done (DoD) checklist. This checklist consists of a condition list that must be satisfied before a software development is considered 'Done'. Some of the items that appear on the DoD checklist before the release include:

* Core functionalities development: verify that all core functionalities of the platform specified in the project proposal are built correctly. The testing phase can be conducted through manual tests or automated test cases.
  + Unit Testing: a set of unit test cases needs to be written and executed to ensure each component of the e-commerce platform can pass successfully.
  + Code Quality and Conventions: check whether the code follows the standard of code structure, naming conventions, readability, and maintainability.
  + UX/UI Testing: ensure the user interface is user-friendly, responsive, and has visual consistency. It should meet all design requirements specified in the proposal.
  + Usability and Scalability: Evaluate the performance of the software under the workloads, such as response time, and resource consumption. Check if the platform can handle when the workloads increase unexpectedly.
  + Security: Check if the platform applied the required authentication and data protection regulations. This process involves performing security testing, protocol review, and IT guidelines/document review.
  + Documentation: The quality management process needs more than just IT testing. It also involves recording and documenting. We need to check the user manuals and API documentation to support users and third-party vendors.

When it comes to quality measurement, there are several approaches and metrics to evaluate the quality. Therefore, many quality models are designed to standardize the measurement procedure. Some popular models are ISO25010, ISO/IEC 9126, COSMIC, and IEEE 1061. Regardless of the chosen model for quality evaluation, the evaluation criteria need to be clearly written, easily understandable, and proper in the DoD Checklist. To enhance the quality of the DoD, we can apply S.M.A.R.T principles.

# DEFINITION OF DONE CHECKLIST

The following checklist is prepared based on the ISO25010 Model with 8 main characteristics to measure the ‘Quality’ of the e-commerce platform.

|  |  |  |  |
| --- | --- | --- | --- |
| **No** | **(Sub) Characteristics** | **Description Y/N** | |
| **1. Functionality Suitability** | | |  |
| 1.1 | Functional Appropriateness | Deploy the platform on the cloud-based hosting server | ☐ |
| 1.2 | Functional Appropriateness | 95% of test cases for user/business account registration and confirmation via CAPTCHA are successfully processed | ☐ |
| 1.3 | Functional Appropriateness | The platform allows user creation for 2 account types: business account and user account | ☐ |
| 1.4 | Functional Appropriateness | The platform allows users to switch between different languages. All content, messages, and alternative text are translated correctly | ☐ |
| 1.5 | Functional Appropriateness | The platform allows users to switch between different currency | ☐ |
| 1.6 | Functional Appropriateness | The data analytics tool can generate reports with meaningful insights from the data in .pdf format | ☐ |
| 1.7 | Functional Appropriateness | The platform's core functionalities can integrate with external systems, such as payment gateways, CRM, shipping, and inventory management systems with less than 5% of API conflicts | ☐ |
| 1.8 | Functional Appropriateness | A product review or rating must be completed within a maximum duration of 10 minutes. Otherwise, it will be discarded. | ☐ |

|  |  |  |  |
| --- | --- | --- | --- |
| 1.10 | Functional Appropriateness | The data analytics tool is integrated with the feedback mechanism to analyze customer feedback | ☐ |
| 1.11 | Functional Correctness | The number of errors of each core functionality of the e-commerce platform should be less than 5% of total automated test cases | ☐ |
| 1.12 | Functional Correctness | During the account registration process, important fields such as name, password, and email are required | ☐ |
| 1.12 | Functional Correctness | Product listings and catalog updates correctly appear on the screen | ☐ |
| 1.13 | Functional Correctness | The searching and filtering options return the correct result | ☐ |
| 1.14 | Functional Correctness | When a new product is added to the shopping cart, the total price is correctly recalculated | ☐ |
| 1.15 | Functional Correctness | Different test cases are performed to check whether all payment scenarios, such as wrong card numbers, and no money in cards… work properly | ☐ |
| 1.16 | Functional Correctness | 99% of the checkout process are completed and recorded in the databases when clicking on the ‘Confirm’ button | ☐ |
| 1.17 | Functional Completeness | The platform can display well-organized product pages with categories, sub-categories, filtering/searching options | ☐ |
| 1.18 | Functional Completeness | The platform can send notifications via email to remind users of the products in their shopping cart after the product is added for 2 days | ☐ |
| 1.19 | Functional Completeness | The inventory management tool can record and update the available stock of a product correctly | ☐ |
| 1.20 | Functional Completeness | The platform can estimate the shipping cost based on the shipping option and the destination of the order | ☐ |
| 1.21 | Functional Completeness | Perform a test to check if the platform sends a confirmation email (with the order details, payment, and shipping methods) when the order is confirmed | ☐ |
| **2. Performance Efficiency** | | |  |
| 2.1 | Time Behaviour | The following actions have a response time of less than 2 seconds:   * Return searching results after the users submit the keywords * Display a list of products * Response to ‘Add’ button to add the items to the shopping cart * Response to the ‘Confirm’ button to complete the checkout * Generate and display an order confirmation page * Record customer feedback submission * Login/Logout | ☐ |
| 2.2 | Time Behaviour | The loading time of pages is less than 5 seconds | ☐ |
| 2.3 | Resource Utilization | The CPU utilization when running the platform is below 80% and memory consumption is below 70% | ☐ |
| 2.4 | Capacity | A simulated workload of 20,000 customers is used for testing. Could the platform handle this simulated workload without reducing the performance by at least 30% | ☐ |

|  |  |  |  |
| --- | --- | --- | --- |
| **3. Compatibility** | | |  |
| 3.1 | Co-existence | The platform can operate simultaneously with at least 2 other software in an operating system without conflicting | ☐ |
| 3.2 | Interoperability | 95% of orders can reflect the payment methods, and shipping information correctly on the third-party system | ☐ |
| 3.3 | Interoperability | The operational data of the platform can be displayed on the data analytics tool in real-time with a maximum delay of 1 minute. | ☐ |
| **4. Usability** | | |  |
| 4.1 | Appropriate recognizability | Users complete the following tasks in less than 3 minutes:   * Register a new account * Submit a product review/rating * Find a product (search by keyword, see the product list, browse through the list, apply filtering/sorting options) * Complete the checkout process (enter the information, select shipping vendor, select payment methods…) | ☐ |
| 4.2 | Learnability | 90% of users (business staff and customers) can complete basic tasks to serve their needs on the first day of the testing period | ☐ |
| 4.3 | Operability | The average number of errors that test users encountered during a 15- minute testing session is less than 2 errors | ☐ |
| 4.4 | User interface aesthetics | The platform is responsive for mobile and desktop | ☐ |
| 4.5 | User interface aesthetics | This condition is evaluated based on a feedback survey  The minimum average satisfaction score of 20 users about the UX/UI of the platform, including webpage structure, text font, color palette, languages, navigation, product display, buttons, etc. is 8 out of 10 | ☐ |
| 4.6 | User error protection | 90% of test users can understand the content of the confirmation dialog box displayed when they:   * Delete/edit their review/rating * Cancel/Confirm their orders * Change the platform settings | ☐ |
| 4.7 | Accessibility | 80% of non-text content (video, images) have alternative text decriptions | ☐ |
| **5. Reliability** | | |  |
| 5.1 | Maturity | Mean Time Between Failures (MTBF) metrics of the platform is at least 30 days | ☐ |
| 5.2 | Fault Tolerance | Mean Time to Recover (MTTR) takes less than 2 hours to recover platform data from errors or failures | ☐ |
| 5.3 | Recoverability | All functionalities of the platform can be fully restored from the backup within 5 hours after the failure | ☐ |
| 5.4 | Availability | The platform is accessible for 99.9% of operating hours | ☐ |

|  |  |  |  |
| --- | --- | --- | --- |
| **6. Security** | | |  |
| 6.1 | Confidentiality | Block at least 95% of unauthorized access attempts and send alerts to users | ☐ |
| 6.2 | Confidentiality | Data encryption and secure authentication protocols are integrated | ☐ |
| 6.3 | Confidentiality | Payment card industry (PCI) compliance is applied | ☐ |
| 6.4 | Integrity | 100% of data is stored and transmitted correctly. Data in the databases is reflected correctly on the user interface and data analytics tools | ☐ |
| 6.5 | Integrity | The platform follows data protection regulations | ☐ |
| 6.6 | Non-repudiation | Implement non-repudiation measures for at least 95% of user actions | ☐ |
| 6.7 | Accountability | 100% of user interactions on the platform and the platform events are recorded in the logbook | ☐ |
| 6.8 | Authenticity | 95% of user authentication processes are successful | ☐ |
| **7. Maintainability** | | |  |
| 7.1 | Analyzability | Reduce time taken to diagnose and resolve reported issues to less than 1 hour 30 minutes | ☐ |
| 7.2 | Modifiability | New change/enhancement is integrated into the platform 1 week after the change request is made | ☐ |
| 7.3 | Testability | Maintain a code coverage of at least 80% through automated test cases | ☐ |
| 7.4 | Modularity | The platform architecture is designed with at least 12 modules | ☐ |
| 7.5 | Reusability | All codes have naming conventions, and comments to modify when needed without affecting other parts (Low coupling – High cohesion) | ☐ |
| 7.6 | Reusability | A minimum of 80% code is reusable | ☐ |
| **8. Portability** | | |  |
| 8.1 | Adaptability | The time to deploy the platform on a new hosting server (on-premises or cloud server) is less than 3 days | ☐ |
| 8.2 | Installability | The time to complete the installation and configuration of all platform components in an environment is less than 5 hours | ☐ |
| 8.3 | Replaceability | Could we replace a component of the platform with a migration time of less than 1 week without impact on other components? | ☐ |