**DAY 2 - Assignment 2 MRIGANKA PATRA**

**Tasks:**

2. Design a test plan that includes: - Project and module overview - Scope of testing - Roles and responsibilities - Entry and exit criteria - Types of testing covered (e.g., smoke, regression, exploratory) -Sprint-based test execution schedule.

* **Project Name:** E-commerce Web Application
* **Module:** Shopping Cart
* **Module Features:**
* Add items to cart
* Update item quantity
* Remove item from cart
* View cart summary
* Apply discount codes
* Proceed to checkout
* Scope of Testing
  + **In-Scope:**
    - Functional testing of Shopping Cart module
    - Integration testing with Product Catalog, User Login, and Payment modules
    - UI/UX validation across browsers/devices
    - Performance testing of add-to-cart operations
    - Security testing for cart manipulation
  + **Out-of-Scope:**
    - Testing of unrelated modules (e.g. user registration, order history)
    - Non-production environments (load testing on staging only)

|  |  |
| --- | --- |
| Role | Responsibility |
| Product Owner | Define user stories and acceptance criteria |
| Scrum Master | Facilitate sprint planning, daily standups, retrospectives |
| Developers | Develop and unit test features |
| QA Engineers / Testers | Design and execute test cases, automate regression suite, report defects |
| UX Designer | Ensure UI/UX standards compliance |
| DevOps Engineer | Manage test environments, CI/CD pipelines |

* Entry Criteria:
  + User story and acceptance criteria defined and approved
  + Code committed to Sprint branch
  + Unit tests passing with 90% coverage
  + Test environment ready
* Exit Criteria:
  + All planned test cases executed
  + No critical or major defects open
  + Regression suite passing
  + Test summary report reviewed and signed off
* Testing

|  |  |
| --- | --- |
| **Testing Type** | **Purpose** |
| Smoke Testing | Verify basic stability of build |
| Functional Testing | Verify cart functions as per user stories |
| Integration Testing | Verify cart works with product catalog, pricing, payment, inventory |
| Regression Testing | Ensure no existing functionality is broken by new changes |
| Exploratory Testing | Perform unscripted testing to find edge-case issues |
| Performance Testing | Validate add/remove/update actions under load |
| Security Testing | Ensure cart is protected from tampering, parameter injection, CSRF |

* Sprint based test execution

|  |  |  |
| --- | --- | --- |
| **Sprint** | **User Stories** | **Testing Activities** |
| Sprint 1 | Add to Cart, View Cart | Functional testing, Smoke testing |
| Sprint 2 | Update Quantity, Remove Item | Functional testing, Integration testing, Exploratory testing |
| Sprint 3 | Apply Discount Code, Checkout Integration | Functional testing, Regression testing, Security testing |
| Sprint 4 | Cross-browser Testing, Performance Testing | UI/UX validation, Load testing, Regression testing |

3. Define the defect management process.

The **Defect Management Process** is a systematic approach to identify, track, and resolve defects in software development.

1. **Defect identification** - Defects are identified through various testing activities, such as unit testing, integration testing, and user acceptance testing.
2. **Defect logging**- Defects are logged in a defect tracking system, along with details such as description, severity, and priority.
3. **Defect triage**- The triage process involves evaluating the defects to determine their priority and the resources required to resolve them.
4. **Defect assignment**- Defects are assigned to developers or testers for resolution, based on their expertise and availability.
5. **Defect resolution -** The assigned personnel work on resolving the defects by fixing the code, updating the documentation, or performing other necessary actions.
6. **Defect verification**- Once the defect is resolved, it is verified by the tester to ensure that it has been fixed correctly and does not introduce any new defects.
7. **Defect closure**- Once the defect has been verified, it is closed and the status is updated in the defect tracking system.
8. **Defect reporting**- Regular reports on the status of defects, including the number of open defects, the number of defects resolved, and the average time to resolve defects, are generated to provide visibility into the defect management process.

4. Include risk assessment and mitigation strategies.

|  |  |  |  |
| --- | --- | --- | --- |
| Risk | Impact | Likelihood | Mitigation Strategy |
| Late delivery of user stories | High | Medium | Prioritize testable stories early in sprint; test partial features when ready |
| Unstable test environment | High | High | Coordinate with DevOps to schedule regular environment validation |
| Cross-browser compatibility issues | Medium | High | Early integration of UI tests across supported browsers |
| Incomplete test coverage due to short sprints | High | Medium | Focus on high-risk areas first; use risk-based testing |
| Frequent UI changes during sprint | Medium | High | Maintain flexible and modular automation scripts; involve QA early in design reviews |