

**ACKO Website Testing**

**Test Strategy for**

**Revision History**

|  |  |  |  |
| --- | --- | --- | --- |
| Date | Version | Author | Description |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |

**Table of Contents:**

**1.** Scope

**2.** Test Approach

**3.** Test Environment

**4.** Sprint Plan

**5.** Testing Tools

**6.** Release Control

**7.** Risk Analysis

**8.** Review and Approvals

**1. Scope:**

* The testing process will follow a structured, phase-wise approach based on the Waterfall model.
* The scope includes validating all key functionalities of Acko.com, such as policy purchase, renewal, payment processing, and claim management.
* **Test Strategy will be reviewed** by Test Lead / QA Manager.
* **Test Strategy will be approved** by Project Manager / Test Manager.
* **Testing activities will be done as per the planned timelines** for each phase.
* **Testing will finish as per schedule,** and any areas not included in this plan will not be tested

**2. Test Approach:**

**1. Process of Testing**

**Requirements Test Planning Test Case Design Test Execution Defect Management Test Closure & Sign-Off**

**2. Testing Levels**

* **Unit Integration System User Acceptance Testing (UAT)**

**3. Roles & Responsibilities**

* **Test Manger – Approves plan and timelines.**
* **Test Lead – Reviews strategy and guides testers.**
* **Tester – Executes test, log and re-test defects.**
* **Developers – Fix defects and support testing.**

**4. Types of Testing**

* **Functional, Security, Performance/Load, Compatibility, Regression**

**5. Testing Approach & Tools**

* **Manual testing for main flows**
* **Automation for regression using Selenium if needed**
* **Defect tracking in JIRA or similar tool.**

**6. Defects Managements & Sign-Off**

* **Log defects , re-test after fixes, run regression**
* **Conduct defect triage to prioritize issue.**
* **Final sign-off after all major defects are resolve**

**3. Test Environments:**

**We need 5 number of required Environments:**

**1. Development- Local Docker setup, Code branching strategy, Mock APIs**

**2. Testing/QA- Dedicated test servers, Automated test tools (Selenium/Testing), CI integration**

**3. Staging/UAT- Production, like infra, masked real data, UAT tracking**

**4. Production- Scalable cloud infra (AWS/GCP), Load balancers & SSL, Backup & compliance (IRDAI)**

**5. Performance- Isolated infra, Load monitoring, Alerts/thresholds**

**Test Data Backup:**

**1. Frequency: Daily incremental, weekly full backups**

**2. Data: Synthetic/masked test data (e.g., policies, claims, user roles)**

**3. Storage: Encrypted cloud storage (e.g., AWS S3)**

**4. Security: Mask personal data (PII), restrict access**

**5. Automation: Use scripts or CI/CD tools (e.g., Jenkins)**

**Restore Strategy:**

**1. When: Before test cycles or after corruption**

**2. How: Automated restore scripts or DB import**

**3. Validation: Run sample test cases after restore**

**4. Isolation: Restore only in test/UAT environments**

**5. Rollback: Snapshot before restore for quick rollback**

**4.Sprint Plan**

| **Sprint** | **Focus Areas** |
| --- | --- |
| Sprint 1 | User Login/Signup Web UI |
| Sprint 2 | Products Insurance Policies |
| Sprint 3 | Resources Claim Status |
| Sprint 4 | Check Pending Challans Create ABHA Cards |

**5. Automation Tools:**

1. Selenium – For website UI testing

2. Postman – For API testing

3. Jenkins – To run tests automatically

4. TestNG – To organize and run test scripts.

**Test Management Tools:**

1. JIRA – For bug tracking

2. TestRail – For managing test cases

3. Allure – For test reports

**Open Source Tools (Free):**

1. Selenium – UI automation (5–10 QA users)

2. Postman (Free) – API testing (3–5 users)

3. Jenkins – CI/CD automation (2–3 users)

4. TestNG / Allure – Test execution & reporting (5–10 users)

**Open Source Tools (Paid):**

1. JIRA – Bug tracking (15–20 users)

2. TestRail – Test case management (5–10 QA users)

3. Browser Stack – Cross-browser/device testing (2–3 users/session)

**6. Release Control:**

**1. Sequential Phases**: Follows waterfall model—requirements → design → development → testing → deployment.

**2. Version History**: Uses versioning format (vX.Y.Z) to track major, minor, and patch updates.

**3. Release Phases**:

* Formal requirement and design approval.
* Complete development and full testing.
* Sign-off from QA, Product, and stakeholders.

**4. Test Coverage**:

* Unit, integration, regression, and end-to-end tests.
* Full test reports required before release.

**5. Release Criteria**: No release without passing all test phases and formal approvals.

**6. Patch Management**: Hotfixes handled via separate patch versions with quick regression checks.

**7. Documentation**: Maintains release notes and change logs for every version.

**7. Risk Analysis:**

**1.Major Risks Identified:**

* Changing or incomplete requirements
* Regulatory changes (IRDAI, DPDP)
* Technical constraints during design (e.g., cloud, AI integration)
* AI module inaccuracies
* Late discovery of bugs or performance issues
* Cloud service failures

**2. Risk Mitigation Strategies:**

* Conduct stakeholder workshops and freeze requirements
* Use threat modelling and DevSecOps in design
* Prototype critical components early
* Execute performance and load testing in realistic environments
* Roll out in pilot phases with user training and support readiness

**3. Contingency Plans:**

* Handle scope changes via controlled mini-iterations
* Activate manual fallback for AI-related failures
* Use alternate cloud zones in case of outages
* Launch escalation and response protocols for customer complaints or breaches.

**8. Review and Approval:**

* The Test Strategy document will first be reviewed by the Test Lead / QA Manager.
* After review, it will be approved and signed off by the Project Manager, Business Team, and Development Team.
* Each Waterfall phase will start only after the previous phase is reviewed and approved.
* All review activities will be signed off by the business team, project management, and development team to ensure agreement.
* A summary of all review changes will be kept at the start of the document, showing the approved date, reviewer name, and comments.
* Final sign‑off will be given after all tests are completed and critical defects are fixed.