I want help with drafting test scenarios

###### ChatGPT said:

Of course! Could you provide more details about the context of your test scenarios? For example:

**Type of testing**: Are these for functional, UI, API, performance, or security testing?

**Application details**: Is it a web app, mobile app, or desktop app? Any specific modules or features you’re focusing on?

**Objective**: Are you aiming for basic validation, edge cases, regression testing, or exploratory testing?

**Tools**: Will you be automating these scenarios, or are these for manual testing?

With this information, I can tailor the test scenarios to fit your needs!

##### You said:

For now Manual, functional for Web

###### ChatGPT said:

Got it! Let's draft some functional test scenarios for manual testing of a web application. I'll keep these general, but feel free to let me know if there are specific features or modules you want to focus on.

### ****Test Scenarios for Web Application (Functional Testing - Manual)****

#### ****1. User Authentication****

**Scenario 1**: Verify that the user can successfully log in with valid credentials.

**Scenario 2**: Verify that the appropriate error message is displayed when logging in with invalid credentials.

**Scenario 3**: Verify that the "Forgot Password" functionality sends a reset link to the registered email.

**Scenario 4**: Verify the user is logged out successfully when the "Logout" button is clicked.

#### ****2. Navigation****

**Scenario 5**: Verify that the

##### You said:

no, Let me help you with my application. It is an LMS application which has content given to cms. Learner consume programs which contains courses, which contain consumable called Assets. These assets are placed under structural combinations called Modules-> Sessions ->segments. Where Assets are under segments. There are multiple asset types. Pdf Recall Online Editor Assessment Super Asset Assessment Assignment Class Opinion Hands-on Project Now once there Assets and the structural combinations are added to the program. The program is published and the published version is V1 and when this version is publish, there will be cohorts created and then learners for this cohort will be enrolled. Once the learners are enrolled, they can login to the LMS and start consuming the assets. So they can complete their circullum/program. Now when the learner is part of the jounrey the Content users can come and either Upgrade the asset which will upgrade to the learner but no updating the progress calculation or completed status of the asset to the learner. Also the they can update the program with changing some configurations like due dates. I want testscenarios for the upgrade cases both for Asset and Program, together and seperately. Lets start with this, I have more to add

###### ChatGPT said:

Thanks for the detailed context! Based on the description of your LMS application, here’s a structured set of **test scenarios for the "Upgrade" cases**, covering both **Assets** and **Programs**. I'll separate them into asset-specific and program-specific scenarios and also include a combined section.

### ****Test Scenarios for Asset Upgrade (Manual - Functional)****

#### ****1. Upgrading Assets for Learners in Progress****

**Scenario 1**: Verify that when an asset is upgraded (e.g., PDF replaced, new assessment questions added), the updated content is reflected for the learner but their progress/completion status remains unchanged.

**Scenario 2**: Verify that if a learner has completed an asset before the upgrade, the "completed" status remains intact after the upgrade.

**Scenario 3**: Verify that for learners who have not yet started the asset, they only see the updated version after the upgrade.

**Scenario 4**: Verify that different asset types (e.g., PDF, Hands-on, Assignment) are upgraded successfully without affecting the structural combinations (Modules, Sessions, Segments).

**Scenario 5**: Verify that upgrading an asset for a cohort does not impact other cohorts using the same program.

**Scenario 6**: Verify the rollback mechanism (if applicable) for an asset upgrade to restore the previous version of the asset.

#### ****2. Notifications & Logs****

**Scenario 7**: Verify that learners are notified (if applicable) about the asset upgrade in their cohort.

**Scenario 8**: Verify that upgrade actions are properly logged for audit purposes (e.g., who upgraded the asset, timestamp, and changes made).

#### ****3. Edge Cases for Asset Upgrades****

**Scenario 9**: Verify that an asset upgrade with invalid content (e.g., corrupt file, invalid assessment configuration) does not impact learners and triggers appropriate error messages.

**Scenario 10**: Verify that simultaneous upgrades to multiple assets in the same segment do not cause any system inconsistencies or display issues for learners.

### ****Test Scenarios for Program Upgrade (Manual - Functional)****

#### ****1. Changing Program Configurations****

**Scenario 11**: Verify that updating a program’s due dates reflects correctly for all enrolled learners in the respective cohort.

**Scenario 12**: Verify that learners with completed or in-progress statuses for the program are not impacted when configurations are updated.

**Scenario 13**: Verify that any configuration changes (e.g., module order, prerequisites) apply only to future progress tracking and not to the progress already recorded.

**Scenario 14**: Verify that changes to program configurations are reflected across all modules, sessions, and segments as intended.

#### ****2. Notifications & Logs****

**Scenario 15**: Verify that learners are notified (if applicable) about program updates (e.g., due date changes).

**Scenario 16**: Verify that program update actions are logged for audit purposes.

#### ****3. Edge Cases for Program Upgrades****

**Scenario 17**: Verify that simultaneous updates to program configurations (e.g., by two admins) result in proper conflict resolution or system validation.

**Scenario 18**: Verify that updates made to one cohort's program configuration do not affect other cohorts.

### ****Test Scenarios for Combined Upgrades (Assets and Programs)****

#### ****1. Cohort-Level Impact****

**Scenario 19**: Verify that upgrading an asset within a program and simultaneously updating program configurations (e.g., due dates) does not cause conflicts or data inconsistencies.

**Scenario 20**: Verify that a learner consuming a program is seamlessly transitioned to the updated asset and program configurations without errors or data loss.

**Scenario 21**: Verify the system behavior when multiple assets and program configurations are upgraded in quick succession (e.g., ensure no learner-facing interruptions).

#### ****2. Notifications****

**Scenario 22**: Verify that learners receive clear and accurate notifications when both an asset and the program are upgraded for their cohort.

#### ****3. Edge Cases****

**Scenario 23**: Verify that when an asset is upgraded and later rolled back while program configurations are updated, the learner experience remains consistent with the latest valid data.

**Scenario 24**: Verify the behavior when an upgraded asset is part of multiple programs and one program is updated simultaneously.

### ****Additional Notes****

**Dependency Testing**: Ensure that structural dependencies (Module → Session → Segment → Asset) are maintained during upgrades.

**Cross-Cohort Testing**: If learners in multiple cohorts use the same program, verify isolation between cohorts during upgrades.

**Regression Testing**: Include regression tests for learner progress tracking, notifications, and reporting dashboards.

Let me know if you want me to add more scenarios or focus on specific areas like asset types or learner progress handling!

##### You said:

remove roll back related and notfication related scenarios

###### ChatGPT said:

Here’s the refined set of test scenarios with rollback and notification-related scenarios removed as per your request:

### ****Test Scenarios for Asset Upgrade (Manual - Functional)****

#### ****1. Upgrading Assets for Learners in Progress****

**Scenario 1**: Verify that when an asset is upgraded (e.g., PDF replaced, new assessment questions added), the updated content is reflected for the learner but their progress/completion status remains unchanged.

**Scenario 2**: Verify that if a learner has completed an asset before the upgrade, the "completed" status remains intact after the upgrade.

**Scenario 3**: Verify that for learners who have not yet started the asset, they only see the updated version after the upgrade.

**Scenario 4**: Verify that different asset types (e.g., PDF, Hands-on, Assignment) are upgraded successfully without affecting the structural combinations (Modules, Sessions, Segments).

**Scenario 5**: Verify that upgrading an asset for a cohort does not impact other cohorts using the same program.

#### ****2. Edge Cases for Asset Upgrades****

**Scenario 6**: Verify that an asset upgrade with invalid content (e.g., corrupt file, invalid assessment configuration) does not impact learners and triggers appropriate error messages.

**Scenario 7**: Verify that simultaneous upgrades to multiple assets in the same segment do not cause any system inconsistencies or display issues for learners.

### ****Test Scenarios for Program Upgrade (Manual - Functional)****

#### ****1. Changing Program Configurations****

**Scenario 8**: Verify that updating a program’s due dates reflects correctly for all enrolled learners in the respective cohort.

**Scenario 9**: Verify that learners with completed or in-progress statuses for the program are not impacted when configurations are updated.

**Scenario 10**: Verify that any configuration changes (e.g., module order, prerequisites) apply only to future progress tracking and not to the progress already recorded.

**Scenario 11**: Verify that changes to program configurations are reflected across all modules, sessions, and segments as intended.

#### ****2. Edge Cases for Program Upgrades****

**Scenario 12**: Verify that simultaneous updates to program configurations (e.g., by two admins) result in proper conflict resolution or system validation.

**Scenario 13**: Verify that updates made to one cohort's program configuration do not affect other cohorts.

### ****Test Scenarios for Combined Upgrades (Assets and Programs)****

#### ****1. Cohort-Level Impact****

**Scenario 14**: Verify that upgrading an asset within a program and simultaneously updating program configurations (e.g., due dates) does not cause conflicts or data inconsistencies.

**Scenario 15**: Verify that a learner consuming a program is seamlessly transitioned to the updated asset and program configurations without errors or data loss.

**Scenario 16**: Verify the system behavior when multiple assets and program configurations are upgraded in quick succession (e.g., ensure no learner-facing interruptions).

#### ****2. Edge Cases****

**Scenario 17**: Verify the behavior when an upgraded asset is part of multiple programs and one program is updated simultaneously.

This streamlined list focuses purely on functional aspects of upgrading assets and programs without rollback or notification-related scenarios. Let me know if you'd like adjustments or more detail!

##### You said:

Make a sheet with this

###### ChatGPT said: