

Home Loan – Salaried Flow - Test Cases

Project: Home Loan - Salaried Flow

Platform: ZooP Orchestrator

Version: 1.0

Tester Name: Anagha S

Date: 6/10/2025

1. QR Code Scan Entry

Test Scenario:

As a user, I would like to enter the loan application journey by scanning the QR code at the builder's end.

Verify QR code scanning using different devices and apps.

Ensure the correct loan application flow loads with pre-filled builder and property details.

Test invalid or expired QR codes

Positive Test case

Verify if the user is able to scan the QR code

Verify if there are any warnings or errors when the user has network issue

Verify if the QR code scanned is only valid for sometime

Verify if the QR code is landing the user on the right page to the user

Verify if the user is able to access the QR code from different networks and devices at different point of time

Verify if scanning the QR code is taking a lot of time

Verify if the user is able to see the property details in the link opened

Verify if the user can use the camera app to open the link

Verify if the user is being redirected to the app once the link loads and if the user is having the app pre-installed

Verify if the user is being asked to download the app, if the application is not installed

Negative Test case

Verify if the URL throw any error during the peak hours or when the user has network issue

Verify if the QR code can be scanned that is already expired but the user is able to open it successfully

Verify if the valid QR code is throwing any error

Verify if there are more than 1 property details when the user scans the QR code

Verify if the user can also scan the fraudulent QR codes from the app

Edge Cases

Verify if the user can scan the QR code and open the link using any UPI app

Verify if the user can scan any QR code (irrelevant to the loan) and successfully open the link

2. Builder Link Entry

As a user, I would like to enter the loan application journey through a unique builder link.

- Validate unique builder links redirect to the correct application flow.
- Ensure pre-filled builder and property details are accurate.
- Handle invalid or expired links.

Positive Test case

Verify if the user is being redirected to the right application he scanned for

Verify if the mapping of the QR code is correct with details/application added regarding the property

Verify if the user gets the prefilled property details

Verify if the details prefilled are accurate

Verify if the app throws the error for the expired links or QR codes

Negative Test case

Verify if the user can still see the expired or blocked property details

Verify if the user can see more than one property details in the link

Edge Cases

Verify if the user can see other links which redirects the user to any other properties or any other fraudulent properties

3. Sign-up/Login with OTP

As a user, I would like to sign up or log in using my phone number and OTP.

- Validate phone number format (10 digits).
- Ensure OTP generation and delivery.
- Verify OTP expiry and error handling.
- Test scenarios for new and existing users.

Positive Test case

Verify if the user can enter 10 digits

Verify if the user is getting OTP on the valid ph no. entered

Verify if the user clicks on Get OTP, OTP is generated

Verify if the user gets OTP is valid for the 15mins

Verify if the user is able to login once the OTP is enters

Verify if the user is able to ask for resending the OTP
Verify if the user gets error message when the user is giving incorrect OTP
Verify if the user gets error message on entering the expired OTP
Verify if there are errors thrown to the user when there is a network failure
Verify if the user will be getting error though one number of the actual vs expected OTP differs

Negative Test case

Verify if the user can enter alphabets or special characters in the phone number field
Verify if there are any errors if the user enters less than or more than 10 digits
Verify if the user will be able to login if the user enters expired OTP
Verify if the user is able to login if he uses incorrect OTP
Verify if the user will be able to input special characters or alphabets in the OTP field

Edge Cases

Verify if the app throws error message if the user has already asked for resending the OTP, but enters the old OTP
Verify if the user can ask for the resending the OTP, before getting the OTP or before clicking on the OTP button

4. Test Scenarios for New Users

Verify that a new user is able to enter a valid 10-digit mobile number and receive an OTP.
Verify that the system redirects the new user to the sign-up flow after successful OTP verification.
Verify the behavior when a new user enters a valid OTP but doesn't complete the sign-up process.
Verify error handling when a new user enters an invalid or expired OTP.
Verify that if a new user's mobile number already exists in the DB, they are redirected to login instead of sign-up.

Test Scenarios for Existing Users

Verify that an existing user can log in using the correct mobile number and OTP.
Verify that an existing user is shown appropriate error messages when entering the wrong OTP.
Verify the login success when the correct OTP is entered within the validity period.
Verify that OTP resend functionality works for existing users.
Verify session persistence for existing users after login.
Verify that existing users can't sign up again with the same phone number.

5. User Data Collection & Verification

As a user, I will share my personal details for verification.

- Validate field-level inputs (e.g., PAN, Aadhar, email, DOB).
- Ensure API integrations for Aadhar and PAN verification.
- Verify error messages for invalid data.

Positive Test case

Verify that the PAN field accepts values only in the correct format (e.g., 5 letters, 4 digits, 1 letter).

Verify that the Aadhaar number is accepted only if it is a 12-digit numeric value.

Verify that the DOB field allows valid date input and format (e.g., DD-MM-YYYY).

Verify if the fields throw validation errors when the input is not as per regex

Verify if the user's data is being verified once submitted

Verify if the user is getting the exact message or notification where the user was not able to pass the verification

Verify that valid personal details trigger successful API verification.

Verify that successful submission gives appropriate confirmation/next step message.

Verify that each field shows validation feedback immediately (client-side validation).

Verify that once valid data is entered and verified, it's correctly saved in the backend.

Negative Test case

Verify if the user can submit the application or save it though the users have entered the details which is not as per the regex

Verify that the form cannot be submitted if any mandatory fields are left blank.

Verify that invalid PAN format (e.g., numbers only, too short) shows an error.

Verify that invalid Aadhaar (e.g., <12 digits, letters included) shows an error.

Verify that the system throws an error if fake or tampered data is entered.

Verify that mismatched data (like PAN name vs Aadhaar name) leads to verification failure.

Edge Cases

Verify behavior when user submits a name with special characters (e.g., Dr. Anagha, M. Kumar@).

Verify how the system handles minor typos between Aadhaar and PAN name (e.g., "Rohan Kumar" vs "Rohan K."), and whether there's a fuzzy match.

Verify if the user can proceed if Aadhaar and PAN are of two different people but both are technically valid.

Verify if verification fails during API downtime or timeout.

Verify behavior when user pastes content vs types it manually (if applicable).

6. Document Upload

As a user, I will upload documents like salary slips and bank statements.

- Validate supported file formats and size limits.
- Ensure successful uploads and error handling.
- Test integration with net banking for fetching bank statements

Positive Test case

Verify if the error is thrown if the user uploads a file which is 1GB

Verify if the error is thrown if any file format except .pdf is uploaded

Verify if there is an error or a warning message thrown when there is a slow network during file upload

Verify if there is an error/ warning thrown when the user is trying to leave the site when the upload is in progress

Verify if the user gets a successful message when the upload is complete

Verify if the user's data is being saved and mapped to the right user

Verify if the user is able to upload the file by dragging and dropping

Verify if the user is able to upload the file by browsing

Verify if the user is able to upload not more than 5 docs

Verify if the user is able to successfully save it before uploading all the docs but not able to submit it before uploading all the docs

Verify if the user is able to select multiple docs while he browses for the files in the system

Verify if the progress is shown on the screen while the docs get uploaded

Verify if the user is able to successfully fetch bank statements via netbanking after entering valid credentials.

Verify if the fetched statements are correctly mapped to the user profile and available for review.

Negative Test case

Verify if the user is able to submit the application or form even without uploading any doc

Verify if the user is getting error when there is a format mismatch

Verify if the user is getting error when there is a size mismatch, small sized files or large sized files

Verify that an appropriate error message is displayed if the user enters incorrect netbanking credentials.

Verify the timeout or failure behavior when the bank server does not respond.

Edge Cases

Verify if the user is able to upload huge no. of docs

Verify if an error is thrown when the user can upload an empty file

Verify if the user is able to upload again if the previously uploaded docs were errored

Verify system behavior if partial bank data is fetched (e.g., 2 out of 6 months) due to an API issue.

Verify if retry/resume is available after failure.

7. Employment & Property Details

As a user, I will enter my employment and property details.

- Ensure accurate data capture for employment and property fields.
- Validate field-level inputs and boundary values.

Positive Test case

Verify if the user is filling in all the mandatory fields before submission

Validate if all the field inputs are following the regex

Verify if data gets saved correctly in DB after submission.

Verify if dropdown fields (like employment type, property type) populate correctly and allow selection.

Verify if numeric fields (like income or property value) allow decimal inputs if required.

Negative Test case

Verify if the user can submit without filling in all the mandatory fields

Verify if the user is able to submit irrelevant data in the fields

Verify that special characters are not accepted where only alphabets are expected (e.g., Employer Name).

Verify submission fails when entering future date for employment start date or unrealistic property possession date.

Edge Cases

Verify the system behavior when maximum field length is reached (e.g., entering 100 characters in address).

Verify if employment data from long company names or unusual job titles is handled properly.

Verify if the user can edit/update the employment/property details after saving but before final submission.

Verify behavior when optional fields are left blank

7. Loan Offers Generation

As a user, I will receive loan offers based on my data.

- Verify loan offer generation and accuracy.
- Confirm proper calculation of interest rates, EMIs, and tenures.
- Test scenarios with no offers available.

Positive Test case

Verify if the user is getting the loan offer as per the data

Verify if the user is getting the right amount of loan offered

Verify if the user is not getting any loan offer
Verify if the user can reject or accept any loan offer
Verify if the user can compare the loan offers
Verify that the system does not display unrealistic tenure options (e.g., 100 years)
Verify if the user the user is getting all the details and the right details of the bank and contact person
Verify if the user can revoke the offer after rejecting
Verify if the user is getting the correct final amount based on the ROI and tenure provided

Negative Test case

Verify if the user can get the offer even after the bank is blocked from user's end
Verify if the user can get the offer even after rejecting the offer for more than 3 times
Verify if users can force-submit an offer selection when no offers are available.
Verify if the system prevents offer selection if user's profile is marked as fraud or restricted.

Edge Cases

Verify if the user is getting the loan more than the property value
Verify behavior when EMI exceeds 90% of user's monthly income (should get rejected or warning).
Verify if the user can accept the offer from 2 banks

8. Co-applicant Addition

As a user, I can add a co-applicant for a better offer.

- Ensure co-applicant data is captured correctly.
- Validate improved loan offers.
- Handle incomplete or invalid co-applicant details.

Positive Test case

Verify if the user has added a valid co-applicant
Verify if the user has added only one co-applicant
Verify if the user is able to get better loan offers after adding the co-applicant
Verify if the user is getting offers based on both applicant and co-applicant profile
Verify if the user is able to add the co applicant
Verify if the user can edit/update co-applicant details after saving.
Verify if the co-applicant receives a notification/email

Negative Test case

Verify if the user gets a error when he tries to add more than 1 co-applicant
Verify if the system throws error when the user adds invalid co applicant details

Verify if the system throws an error if required fields for co-applicant (e.g., PAN/Aadhar) are left blank.

Verify error handling when invalid regex or characters are entered in the co-applicant fields.

Verify if an error is shown when a blacklisted or flagged person is added as co-applicant.

Edge Cases

Verify if the system prevents the primary applicant from being added again as a co-applicant.

Verify the behavior of the system when the user adds himself as the co-applicant

Verify the behavior of the system when the user added incomplete details of the co-applicant but still getting offers related to co-applicant profile

9. Offer Selection

As a user, I will select a loan offer.

- Verify successful offer selection.
- Ensure accurate data transmission to lenders.
- Test expired or withdrawn offers.

Positive Test case

Verify if the user is able select a loan offer is eligible for successfully

Verify if the lender is getting notification of the user accepting loan offer

Verify if the user is getting error on accepting the expired offer

Verify if the user is getting a email or notification of accepting the loan offer

Verify if the system throws error when the user tries to select the loan offer is not eligible for

Verify if the user can select more than one loan offer

Verify once the user selects the loan offer, its saved in the database

Verify if there is a notification sent to bank once the user accepts the load offer

Verify if the user can withdraw the loan offer within a month

Verify if the user can check for other offers even after accepting one loan offer

Verify if the user can accept more than one loan offer

Verify if the user can accept the already accepted loan offer

Negative Test case

Verify if the system blocks the user from selecting an expired/withdrawn loan offer.

Verify if error is shown when DB fails to save the selected offer.

Verify system behavior if there's a session timeout while selecting an offer.

Verify that users without pre-approved offers cannot proceed with selection.

Verify if users from a blocked geography/IP are prevented from selection (security check).

Edge Cases

Verify if the user is able to select a second loan offer

Verify if the system throws error when the user tries to accept the already withdrawn loan offer

10. Lender Integration

As a user, I will complete the lender integration process.

- Validate lender data requests and document uploads.
- Ensure successful login fee payments.
- Verify e-signature integration.

Positive Test case

Verify if the user is able to login

Verify if the user is able to make the fee payment

Verify if the payment is reflected in the DB

Verify if the user is able to pay through all the modes of transaction available

Verify if the payment is going through the payment gateway successfully

Verify if the user is able to cancel the payment in between

Verify if the lender has got the login fees as per the pre decided percentage for successful transaction

Verify if the app is getting the login fee payments for successful transaction

Verify if the amount debits from the user bank account

Verify if the user is using the valid credentials like account no. or UPI ID for the transaction

Verify if the user can upload the signature doc

Verify if the user can upload the doc in .jpg or doc format

Verify if the user can take a picture from the app to upload the doc

Verify if the user can make a e-signature in the app

Verify if the lender can send messages asking for the documents

Verify if the app will throw error or warning to the lender if there are docs related to it already

Verify if the app is throwing error only regarding the profile lender is checking

Verify if payment confirmation email or SMS is sent to the user.

Verify backend sync with payment gateway logs.

Negative Test case

Verify if the app can starts throwing error on slow networks

Verify if the app throws error when there are too many users using

Verify if the user gets the error on transaction when there is a network issue

Verify if the user can add some irrelevant or a heavy doc(1GB)

Verify if the app shows successful message even when the lender's message is not reaching the user

Verify if user can manually refresh application status.

Edge Cases

Verify if the non logged in or non signed up users are also able to make the payment

Verify if the user gets the error when there is no required balance in the account

Verify if the user can make the second attempt of payment from different mode or different account

Verify if the user can make multiple payments using the same logged in account

11. Loan Application Status Tracking

As a user, I will track my loan application status.

- Validate real-time status updates.
- Ensure notifications are triggered correctly.
- Test various application states (e.g., In Progress, Approved, Rejected).

Positive Test case

Verify if the user gets the success message once the application is approved

Verify if the user gets the failure message once the application is rejected

Verify if the user is getting update notifications when there is a change in application status like start to inprogress and in progress to pending

Verify if the user is getting the notification on time

Verify if the user is getting no updates in notification section when there are no notifications

Verify if the user can scroll to see all the notifications from Day1

Negative Test case

Verify if the user is getting no updates though there are notifications generated

Edge Cases

verify if the user is getting his own notifications and not getting other login notification

Verify if the notification will not pop up when the user has logged out

Verify if the user gets the notification pop up but there are no notifications when he visits the app

Verify if the notification is being erased after sometime or after user logging out

Session Management

As a returning user, I will resume my loan application.

- Test resuming incomplete journeys.
- Validate session persistence.

Positive Test case

Verify if the user's are able to continue from where they left off

Verify if the app is saving the data after few changes made by the user or after sometime

Verify if the data autosaved is being shown when the user logs in again

Verify if the data is being autosaved when the user click on back button or home button

Negative Test case

Verify if the user gets error on saving the data

Edge Cases

Verify if the data which was autosaved is only available and not other people's data

Test resuming from multiple devices
