Exploratory Testing

Feature:

Custom loan repayments for business loans

Assumptions:

User can have more than one existing loan.

Only logged in / authenticated users can make custom repayments

User is shown a list of all existing loans and has the option to select

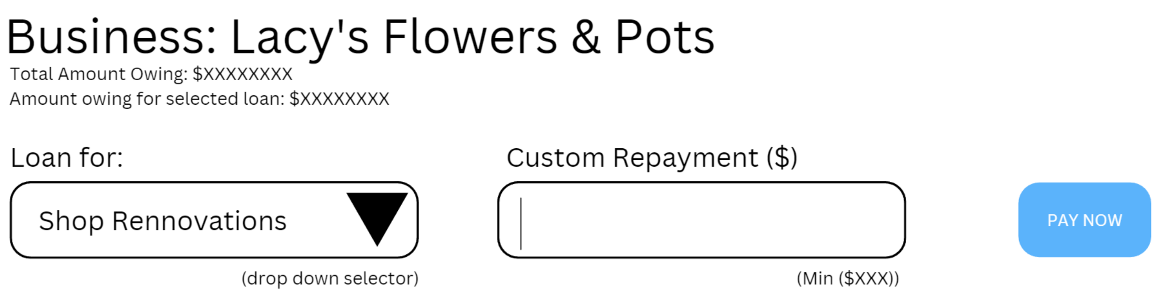
Custom payments requests are sent when the user presses a “Pay” button

User can input a custom integer amount to pay towards the loan.

There is a minimum amount that the user must contribute (e.g $10 minimum / custom repayment).

The user cannot contribute more than the total amount owed.

Mock user interface



Business: Lacy's Flowers & Pots

Total Amount Owing: $XXXXXXXX

Amount owing for selected loan: $XXXXXXXX

Loan for: Shop Rennovations

Custom Repayment ($):

Approach:

In this exploratory testing session I first explored the form/the main ui that the user would use to make any custom repayments, testing invalid and valid inputs and searching for any errors / popups that may appear. I also tested for any invalid api requests, when I would submit the form with these invalid parameters to see if the API would accept / reject the request. I also explored the custom repayments feature from an un-authenticated user, checking that they cannot access the page, and if they some how do access it while authenticated, checking the response of an unauthenticated users payment request. I then check for valid requests made by authenticated users with valid tokens. Looking for behaviour and how the page/application updates in response to these valid requests.

Scenarios:

1. Explore incorrect input handling & invalid api requests
    a. Explore form inputs loan selector
        i. No loan selected & form submit, check error popups & handling on the user interface
            1. Look for any api requests / failures upon pressing the “pay now” button
        ii. Loan selected but no custom repayment inputted, check error popups & handling on the user interface
            1. Look for any api requests / failures upon pressing the “pay now” button
    b. Explore Custom repayments input
        i. Explore invalid input (non integer characters) & check for any errors on the form (when inputting into the form and submitting)
            1. Look for any api requests / failures upon pressing the “pay now” button
        ii. Explore invalid input (negative values & values less than the minimum) & check for any errors (when inputting into the form and submitting)
            1. Look for any api requests / failures upon pressing the “pay now” button
        iii. Explore invalid input (Amounts greater than the amount for this loan) & check for any errors (when inputting into the form and submitting)
            1. Look for any api requests / failures upon pressing the “pay now” button
2. Explore invalid api requests (un-authenticated user, invalid user tokens)

   a. Log user out and try to access the custom loan repayments page
   b. Invalidate the user token and try to make a custom repayment
      i. Analyse any errors/api requests

3. Explore any valid api requests (get, post, put, del)
   a. Valid loan selected &
      i. Minimum custom amount paid
         1. Check how the application responds to repayments (does it update the outstanding loan balance for both total and individual loans?, does it provide the user with a popup?)
         2. Look for successful api requests & status codes (does it respond with code 2xx?)
      ii. Maximum remaining loan paid
         1. Check how application handles the loan being paid off (does it remove the loan from the list of outstanding loans? Does it update the total amount owing balance? does it provide the user with a pop up?
         2. Look for successful api requests & status codes (does it respond with code 2xx?)