

Make UI of Website Developed in Flutter Be Aesthetically Pleasing

MyCatalog.bz

Project Specifications

Version 1.0

May 5, 2022

Overview

A website has been developed for the country of Belize where businesses can upload their products and create a basic online catalog of the products they sell.

The website has a front end component developed in Flutter and a back end component accessible via APIs.

The following major aspects of the website have been developed and completed:

- The entire back end API component
- The entire functionality of the front end component

There is one remaining aspect that needs to be updated. The user interface of the front end is very basic. It needs to be improved; specifically it needs to be aesthetically pleasing.

The user interface must be updated in a manner so that it is mobile responsive, intuitive, clean, clear, easy to navigate, and easy to use. The user interface must not come across as clunky, cluttered, and complex.

Requirement

The requirement of this project is to update the user interface of the front end to make it aesthetically pleasing.

Nothing in terms of functionality would need to be developed. All of that has already been done.

The desired end result of how each page should look is also being provided. The developer must update the UI and change it from the current basic interface to the end result being provided. The update does not have to be identical to the end result being provided, but it must follow it to a large extent.

Due to the nature of the project, the developer must have an in-depth knowledge of how widgets work in Flutter. The following widgets have been used in this project: Row, Column, ListViewBuilder, GridViewBuilder, Container, BoxDecoration, TextButton, ElevatedButton, AlertDialog, TextFormField, SingleChildScrollView, Scaffold, AppBar, TextStyle, etc.

Specifications

There are a total of 17 pages whose user interface needs to be updated. A listing of each page is provided in the below table.

In order to get an accurate idea of the scope of work to be done, the developer can see how the current UI of a page is and how the same UI is being requested to look. For some pages there are also additional requirements listed.

The domain of the pages related to the current UI is dev.tanatechnologies.com. The domain of the pages related to the intended UI is dev2.tanatechnologies.com.

For example: the current UI of the login page can be seen at dev.tanatechnologies.com/login. The end result of the UI after the update to the user interface should look like dev2.tanatechnologies.com/login.

Where a sign in with an account is requested, the developer can use the email mikadostore@gmail.com and password test.

Page ID	Page Name	Dart file in codebase	Current design URL	Intended design URL	Notes
1	Register	register	/register	/register	
2	Login	login	/login	/login	
3	Forgot password	forgot_password	/forgot-password	/password/reset	
4	Reset password	reset_password	/reset-password	/password/reset/{code}?email={email}	Trigger a password reset to see this page
5	Admin home	home	/	/user	Account sign in required.
6	Add item	add_item	/add-item	/item/create	Account sign in required.
7	Change password	change_password	/change-password	/user/edit/password	Account sign in required.
8	Edit item	edit_item	/edit-item	/item/1316/edit, /item/1385/edit	Account sign in required.
9	Edit profile	edit_profile	/edit-profile	/user/edit	Account sign in required.
10	Edit variation	edit_variation	/edit-variation	/variation/4387/edit	Account sign in required.
11	Manage categories	user_categories	/user-categories	/usercategories	Account sign in required.
12	Add or edit category	user_category_detail	/user-category-detail	/usercategory/create	Account sign in required.
13	Manage items	user_items	/user-items	/items	Account sign in required.

14	Catalog home	catalog_home	/u/14	/u/14	
15	View category in catalog	view_category	/u/14/c/51	/u/14/c/51	
16	View item in catalog	view_item	/u/14/i/1316, /u/14/i/567	/u/14/i/1316, /u/14/i/567	
17	Search item	search_item	/u/14/s/Oxford	/u/14/i/s?q=Oxford	

The below additional requirements for the following pages must also be met.

Forgot password

- There is a message shown after the button is clicked. The style of that message must be updated.

Add item

- This page has 2 states. Both must be updated.
 - The first state is when the Select pictures and Upload buttons are displayed
 - The second state is after the upload has occurred
 - The design of both states must be updated
 - There is a label dialog that comes up when pressing the Label, Update price for selected, and Update categories for selected buttons. That dialog must be updated.
- The list does not scroll when many items are added and all do not fit on the page

Manage items

- The manage items page should not list the items in grid view.
 - It should list them in tabular mode; for example as it looks in this [picture](#).
- When the list has no more records to show, the text “There are no more records to show” appears

- When the user scrolls back up, this text remains at the bottom of the screen
 - But this text should not always be visible at the bottom of the screen.
 - It should be a part of the scroll view so when the user scrolls back up, this text should not be visible
- The search box should be removed from the top of the screen and made visible only upon clicking a search button on the app bar

For all pages

- Update font of error dialog box to have the color of red
 - The error dialog is always populated via the showTanaDialog function in the helper.dart file
- Standardize the font style used across all browsers
 - In Safari the default font style used is Times New Roman
 - But in Chrome and other browsers another default font style is used
 - Use the font used in Chrome
- Make the size of the thumbnail images be mobile responsive
 - Right now the thumbnail of all images has been set to a constant height and width of 200
 - Make the thumbnail size be responsive on 2 images fit horizontally on a small screen size, 4 images fit on a medium screen size, and 6 images fit on a large screen size
 - At all times the width of the thumbnail must be based on the screen size, and the height of the thumbnail should always be equal to the width of the thumbnail
 - The pages that have thumbnails are: add item, catalog home, search item, edit item, manage items, view category in catalog, and view item in catalog
- The design of the pages (for example padding between widgets) vary slightly depending on whether it is being seen from a desktop or mobile device.
 - Standardize the design across the desktop or mobile device

Admin home

- When the user first visits the page after creating an account, make the design of the welcome banner similar to the one in this [picture](#)
- When the user clicks the Start managing button on this welcome banner, make the design of the welcome instructions be similar to the one in this [picture](#)

- Make the design of the share button be similar to the one in this [picture](#)

View category

- When the list has no more records to show, the text “There are no more records to show” appears
 - However, when the user scrolls back up, this text remains at the bottom of the page
 - But this text should scroll along with the page and not always be visible at the bottom of the page
 - When the user scrolls back up, this text should not be visible
- The category name at the top is always visible
 - But this text should scroll along with the page and not always be visible at the top of the page
 - When the user scrolls back down, the text should not be visible

Catalog home

- On a mobile device, I cannot perform the scroll action on the area of the phone that contains images. I can only scroll on the area that contains the buttons at the top
- The search box should be removed from the top of the screen and made visible only upon clicking a search button on the app bar

View item

- On a mobile device, when the user scrolls down to the bottom of the page, he cannot scroll back up.
 - I think it is because of the List View at the bottom of the page

Budget

The estimated budget of this project is expected to be in the range of \$425 USD.

The was determined using the following numbers:

- There are a total of 17 pages
- Each page will take 1 hour to update
- The charge per hour will be \$25 USD
- The total cost of all pages will be \$425 USD (17x1x25)

The payment plan of the project will be as follows:

- 34% will be paid upon signing of contract
- 33% will be due when the first 9 pages are completed
 - The first 8 pages to complete are page ID 1, 2, 5, 6, 8, 11, 12, 14, and 17
- 33% will be due when the final 8 pages are completed
 - The final 8 pages to complete are page ID 3, 4, 7, 9, 10, 13, 15, and 16

Project Conditions

All developers interested in this project must submit the following:

- Examples of past development done in Flutter
- It is preferable the examples be concentrated in the area of user interface design
- Be prepared to perform a basic demo of these example applications
- Estimated completion date
- Cost

This project has been developed on Flutter 2.10. This is the version on which all development must be done.

Access to the github repository containing the code base will be provided. All changes must be done on a branch of the repository in Github.

Upon approval, the developer must make himself available via Zoom so a detailed walkthrough of the changes to be done on each page can be provided.

Long Term Potential

The long term vision of this project is to grow this website into Belize's first eCommerce platform.

There is much more front end development that needs to occur on this codebase.

Provided this project is successful, the developer will naturally receive higher preference for other work related to this project.

This is expected to be an ongoing project and serve as a steady income stream on a monthly basis for the developer as long as the website is successful.