GitHub Username: proevan

Jeanz

Description

Jeanz is your assistant to give you sale update for your favorite jeans brands and models. You can easily subscribe brands and models to get the sale notification through the app and email. And you can also browsing the lastest sale items on this app. Start to buy your jeans in the cheapest way!

Intended User

Myself! And also the customers who have favorite jeans brands and even a particular jeans model and wanting to buy their jeans only on a sale!

Features

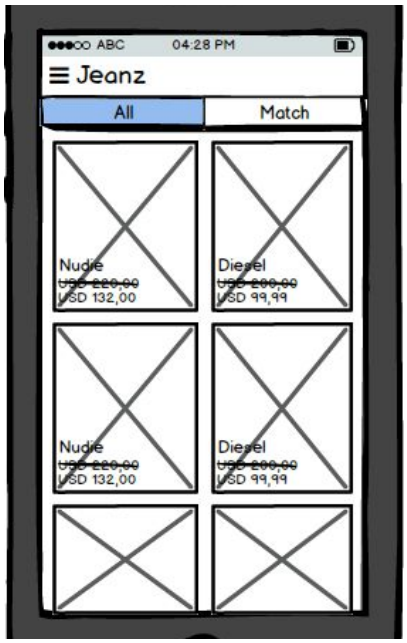
List the main features of your app. For example:

- Show sale jeans items
- Filter brands, model, waist, length and discount percentage
- Login system
- Subscribe sale notification
- Push notification

User Interface Mocks

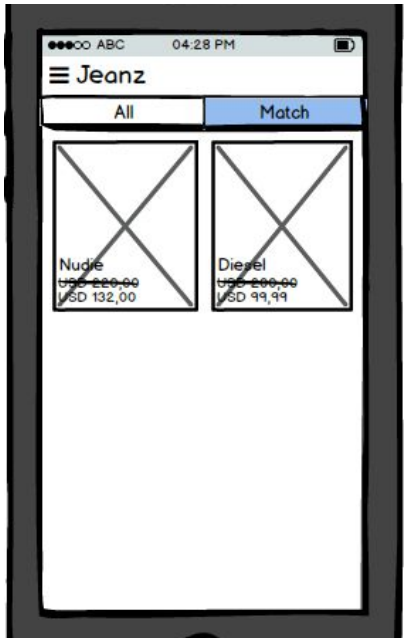
These can be created by hand (take a photo of your drawings and insert them in this flow), or using a program like Photoshop or Balsamiq.

Screen 1



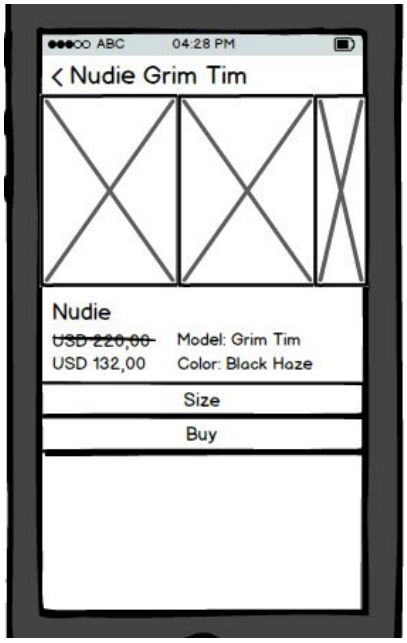
Item list page
it's app's launch page. Provide a navigation drawer.
Tab 'All' to list out all the items.

Screen 2



Item list page
Tab ‘Match’ to list out the matched items which fit user’s preference.

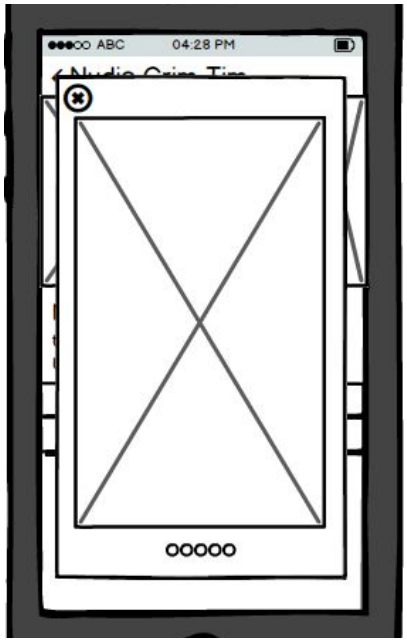
Screen 3



Item detail page
Show the item pictures and details.
Click size button to see the available item sizes.

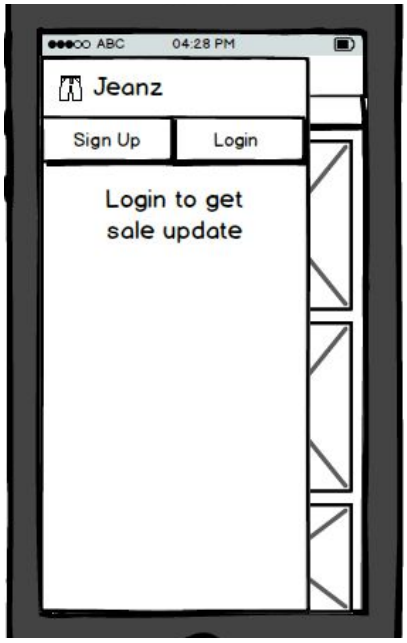
Click buy button to open original selling webpage using web browser.

Screen 4



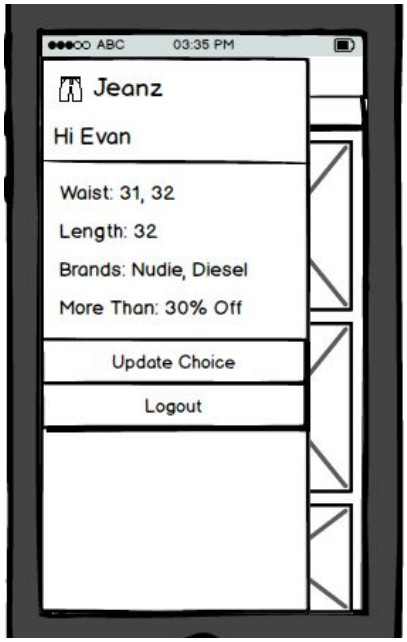
Item picture popout dialog
Swipe to browse the item pictures.

Screen 5



Nagivation drawer
Provide login and signup.

Screen 6



Nagivation drawer
If user already logined, show the preference user selected.
Provide a button to update item preference, and a logout button.

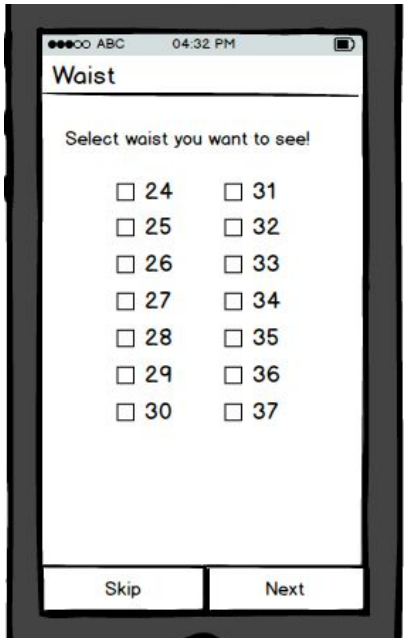
Screen 7

Login page
Provide basic login feature, forgot password button and sign up button.

Screen 8

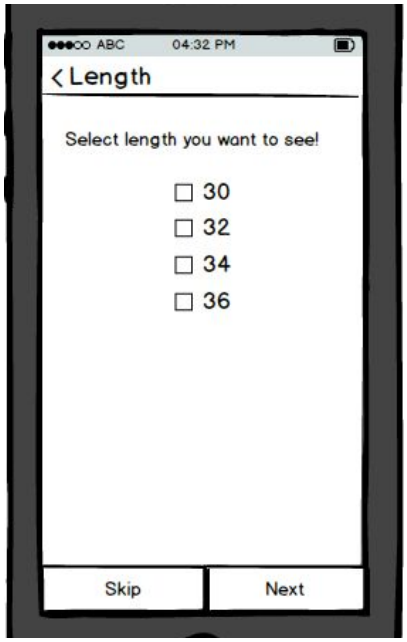
Sign up page
Fill name, email and password to sign up a account

Screen 9



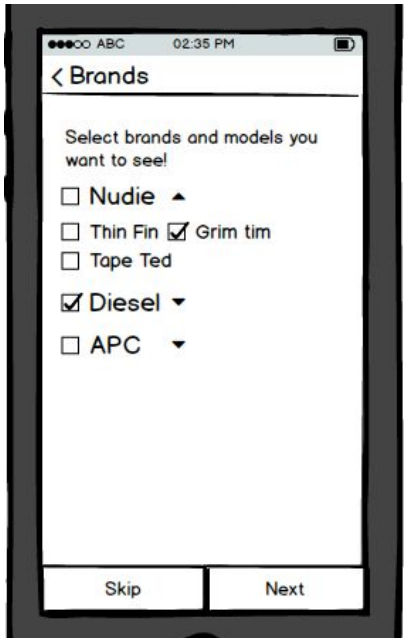
Select jeans waist preference page
Select waist preference or press skip to ignore

Screen 10



Select jeans length preference page
Select length preference or press skip to ignore

Screen 11



Select jeans brands preference page
Select brands and jeans models preference or press skip to ignore
Provide descriptive text for each screen

Screen 12

Select item discount preference page
Select item discount preference or press skip to ignore

Add as many screens as you need to portray your app's UI flow.

Key Considerations

How will your app handle data persistence?

Describe how your app with handle data. (For example, will you build a Content Provider or connect to an existing one?)
Not necessary but will use content provider to cache the sale items data in local db and load data locally when there's no internet connection in order to fit project specification.

Describe any corner cases in the UX.

When user click a sale item push notification, app will open and navigate to the item detail page automatically.

Describe any libraries you'll be using and share your reasoning for including them.

glide, for image loading
retrofit, for restful http client
gson, to parse json data
butterknife, to inject views easier
org.parceler:parceler, to pass data between activities easier
com.orhanobut:logger, for logging
google analytics, for analytics
crashlytics, for crash report

Next Steps: Required Tasks

This is the section where you can take the main features of your app (declared above) and decompose them into tangible technical tasks that you can complete incrementally until you have a finished app.

Task 1: Project Setup

- Set up android project
- Configure libraries

Task 2: Implement UI for Each Activity and Fragment

- Build UI for MainActivity
- Build UI for ItemActivity
- Build UI for LoginActivity
- Build UI for SignUpActivity
- Build UI for PicturesDialogFragment
- Build UI for SelectWaistActivity
- Build UI for SelectLengthActivity
- Build UI for SelectBrandsActivity
- Build UI for SelectSaleRateActivity

Task 3: Get and display items

- Get items from server api
- Display data to a recyclerview

Task 4: Display items detail

- Display image and information to a item detail page

Task 5: Implement Login system

- Integrate sign up api between server
- Integrate login api between server
- Integrate reset password api between server
- Handle api token

Task 6: Implement subscribe

- Implement subscribe flow after user sign up
- Implement feature to modify subscribe item conditions
- Show matched items filter by for brands, model, waist, length and discount percentage

Task 7: Implement push notification

- Integrate parse.com for push notification service
- Handle clicking and auto navigate to sale item detail page

Task 8: Implement data cache for item list (to meet project specifications)

- Create db schema
- Implement simple dao service
- Implement data cache for item list data

Task 9: Implement a widget (to meet project specifications)

- Implement a widget showing a latest sale item user subscribed