buyDeis

Final Presentation

Team 6 - Fatumata Jalloh, Yalda Mauj, Sophia Wang

buyDeis Team

Fatumata Jalloh

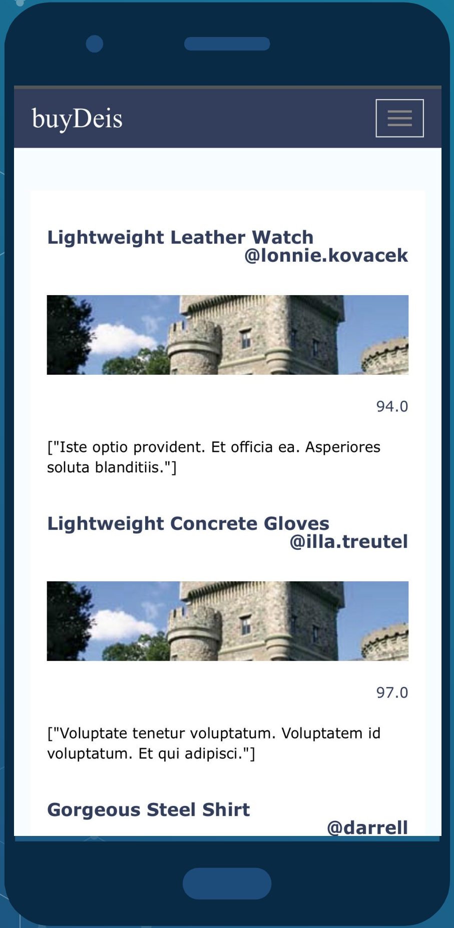
Yalda Mauj

Sophia Wang

1

What is buyDeis?

Project overview



buyDeis

Lightweight Leather Watch
@lonnie.kovacek

94.0

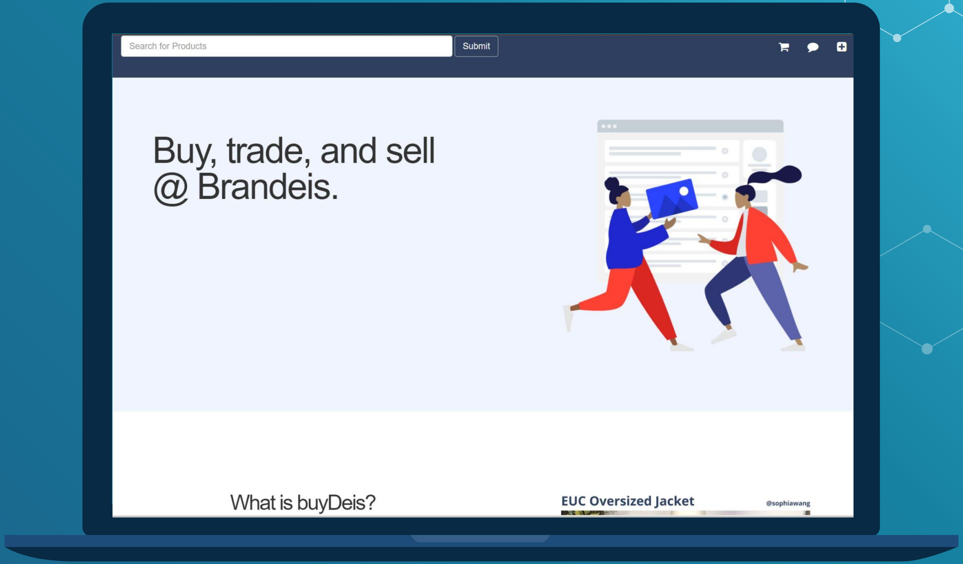
["Iste optio provident. Et officia ea. Asperiores soluta blanditiis."]

Lightweight Concrete Gloves
@illa.treutel

97.0

["Voluptate tenetur voluptatum. Voluptatem id voluptatum. Et qui adipisci."]

Gorgeous Steel Shirt
@darrell



Buy, trade, and sell
@ Brandeis.

What is buyDeis?

EUC Oversized Jacket
@sophiawang

buyDeis provides a platform for Brandeis students to practice sustainability by selling goods that they don’t use/need

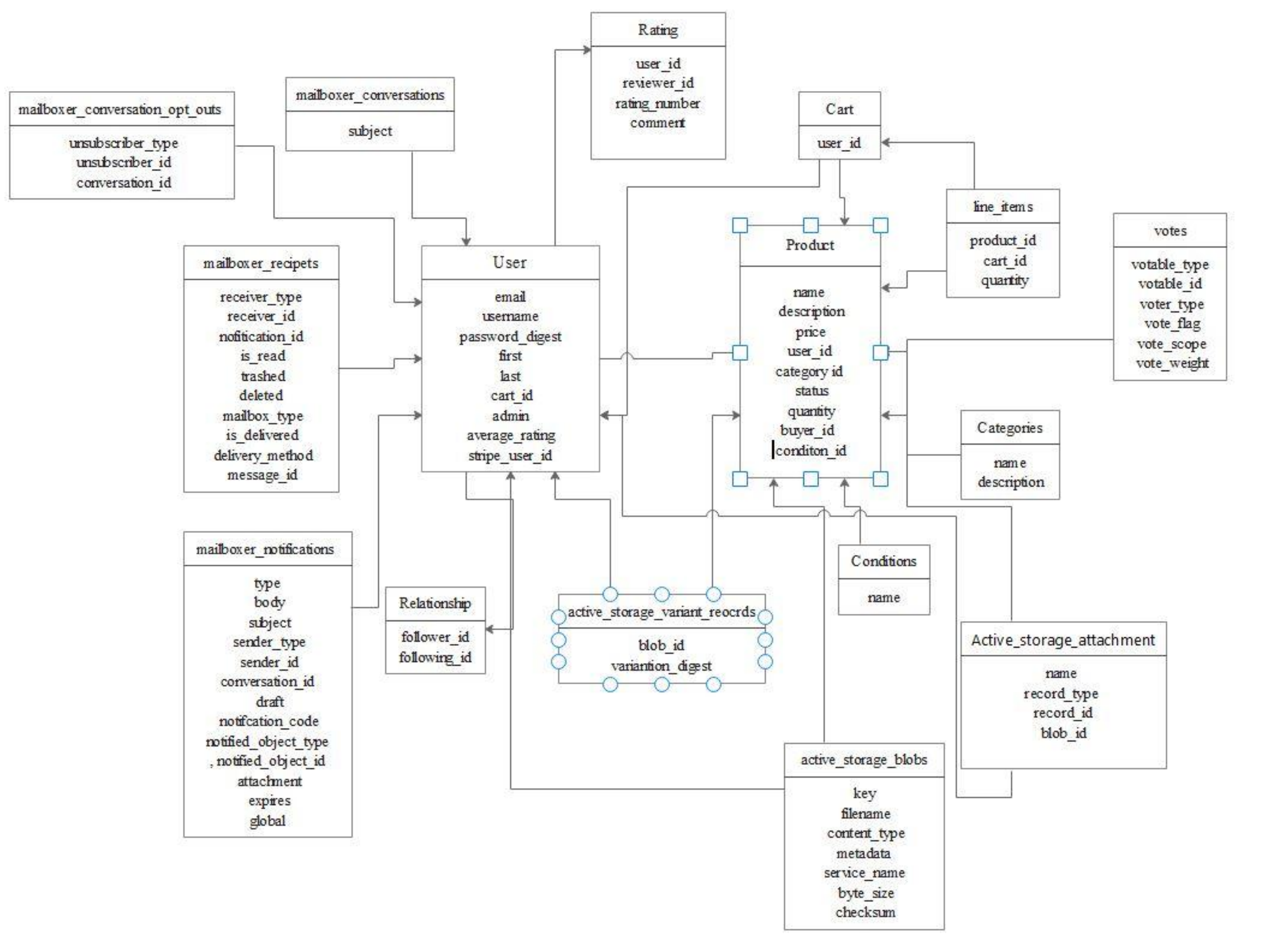
Functionalities

- Allows users to create posts and see the feed for items that are available for other users to purchase
- Gives users the option to search for specific products and filter products on their feed
- Allow users to follow each other
- Allows users to leave reviews for other users that they have purchased from
- Allows users to like/unlike products
- Allows users to message other users about their products
- Allows users to have a shopping cart and perform transactions

Demo

2

What is the technical architecture of our application?



mailboxer_conversation_opt_outs
unsubscriber_type
unsubscriber_id
conversation_id

mailboxer_conversations
subject

Rating
user_id
reviewer_id
rating_number
comment

Cart
user_id

line_items
product_id
cart_id
quantity

votes
votable_type
votable_id
voter_type
vote_flag
vote_scope
vote_weight

mailboxer_recipets
receiver_type
receiver_id
nofitication_id
is_read
trashed
deleted
mailbox_type
is_delivered
delivery_method
message_id

User
email
username
password_digest
first
last
cart_id
admin
average_rating
stripe_user_id

Product
name
description
price
user_id
category id
status
quantity
buyer_id
conditon_id

Categories
name
description

mailboxer_notifications
type
body
subject
sender_type
sender_id
conversation_id
draft
notifcation_code
notified_object_type
, notified_object_id
attachment
expires
global

Relationship
follower_id
following_id

active_storage_variant_reocrds
blob_id
variantion_digest

Conditions
name

Active_storage_attachment
name
record_type
record_id
blob_id

active_storage_blobs
key
filename
content_type
metadata
service_name
byte_size
checksum

3

Technology: Gems, Integrations

Gemfile

- Mailboxer
  - Mailboxer allows any object created with a acts_as_messageable property to have a mailbox. This means they can send and receive messages. They can also have conversations with other users. This is implemented with the messages and conversations controllers and views
- Active storage
  - Active storage allows us to upload files, in our case images, and store the data of the files. By default, it stores the data locally, which why we used google cloud to store it on the cloud.
- Google cloud storage
  - We used google cloud storage and created a bucket to store all images uploaded on the site. We added this to our storage.yml file with the specific encryption data given by google so all images stored in the site are saved.
- Bcrypt
  - Allows us to encrypt user passwords and ensures all user password created are correct

Gemfile

- Stripe, omniauth-stripe-connect, oauth2
  - These gems allowed us to connect with the stripe UI to integrate transactions. The oauth2/omniauth-stripe-connect allowed for authorization with these transactions
- Searchkick (with Elasticsearch)
  - Searchkick allows for more advanced searching and filtering. It accounts for misspelled search input and incomplete input. It also allows for compound filtering with the products
- Acts_as_votable
  - Allows users to like/unlike a product and helps track number of likes and dislikes
- Cancancan, rails-admin
  - Allowed us to verify views. Users cannot see restricted views

4 Testing

Testing

- minitests for products and users
- Model testing
- TDD

5

What are we proud of the most?

Integrating Stripe and Searchkick

- Stripe
  - Integrating Stripe UI and getting the functionality to have secure transactions on the website
- Searchkick
  - We used elasticsearch to integrate searchkick which allowed us to have search functionality that allows us to search for products even with misspelling errors or incomplete search input. This also allows us to filter the products by category, condition, price, and date. The filtering was also compounded.

6

Where can our product be improved or expanded?

Features we could add in the future

- In app notification
  - New messages, follows and likes
- Email notification
  - When a new product is purchased
- Comments on posts
- Resetting password
- Using only Brandeis emails to sign up
- Trading

8 Group Structure

Deployment, Integration, Teamwork

- We deployed multiple times a week.
- We met every sunday and made sure we all had the same version of the app on our individual laptops.
- Used trello to track progress
- We met with Julian every Wednesday to track our progress

THANKS!