Stuy N’ Large
Software Development
Period 9
Edward Luo, Ryan SIu, Max Korsun, Andrew Wong
Project Manager: Edward

Stuy N’ Buy

Stuy n’ Buy is a website for members of the Stuyvesant community to buy and sell items they no longer need. Members will create accounts using their stuy.edu email and password. From there, members can list items for sale, accompanied by a picture, description, and price. They can also search for items and see if any relevant items are being sold, and if not, indicate items they are “looking for”. A buyer and seller may contact each other to agree on a price and place for exchange. Stuy N’ Buy is not responsible for any transactions that may or may not occur between a buyer and seller. We are also not responsible for emotional or physical damages incurred by meeting with super seniors or having a teacher spot you juuling.

COMPONENTS

Back end

- api.py
    - Handles API usage and parsing JSON data
- auth.py
    - Handles authentication (logging in, creating account, etc.)
    - Make sure that user has a stuy.edu email
- db.py
    - Contains methods to interact with the database, app.db
- gmail.py
    - Handles gmail related functions (creating and sending messages to users, changing names and passwords, setting admins, etc.)
- main.py
    - Runs the application on localhost
    - Contains the views of the application
- Items.py
    - Handles item related functions (adding, changing, and removing items, status, etc.)
- pictures.py
    - Handles picture related functions (adding and removing pictures)
- search.py
    - Helps with search filtering
- users.py
    - Handles user related functions (creating and authorizing users,
- client_secret.json
    - Contains secret API information for the Gmail API
-

Front end (UI)

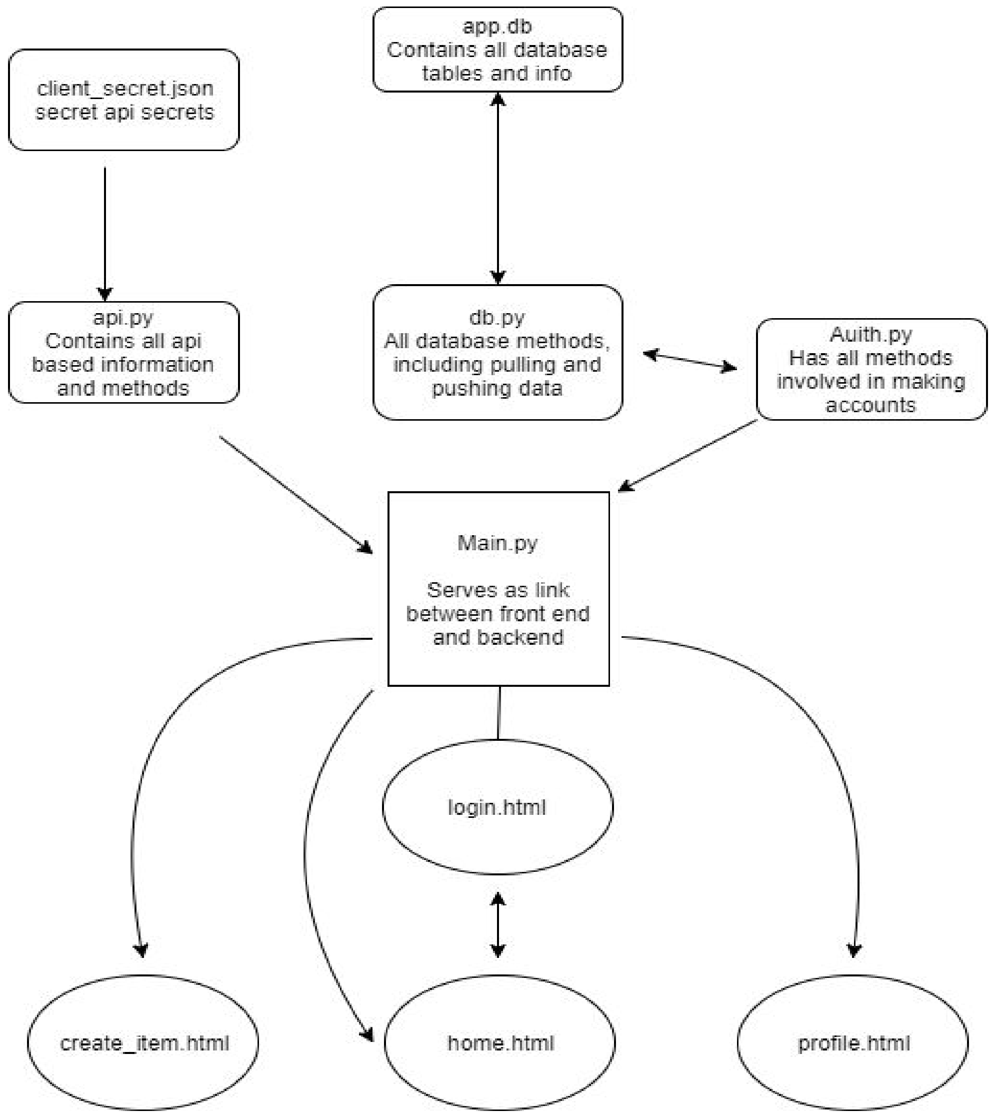
- CSS styling: Foundation
    - Navbar and footer in the base
- Javascript
    - Listed items update automatically, refresh listings button (AJAX)

Views

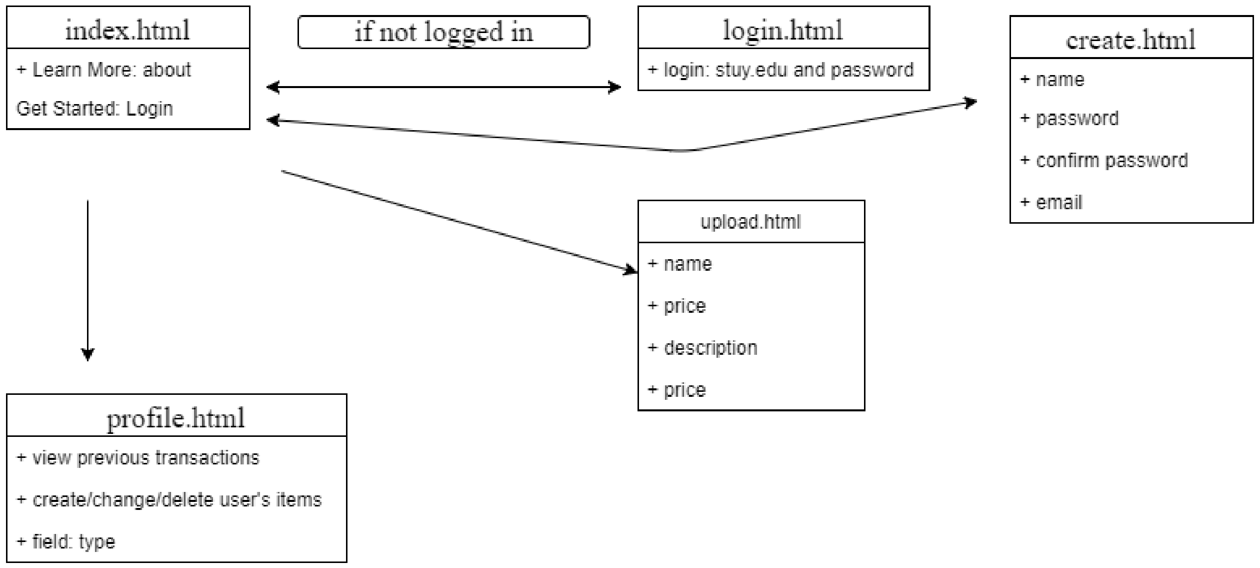
- Login/create
- Index
    - Public: about the website, links to login/create
    - Logged in: listed items
        - Search and filters (price, alphabetical)
        - Link to create item
- View Item
    - Item price, seller, description, pictures, meetup button

    - Modal prompting for email message
- Create Item
    - Name, description, price
    - Upload pictures - file form field and stored locally
- Profile
    - View user information
    - View user-created items and previous transactions
    - Links to delete, create, change status of user-owned items
- Admin interfaces
    - View, edit, delete (moderate) items
    - View, delete users

COMPONENT MAP


client_secret.json
secret api secrets
app.db
Contains all database tables and info
api.py
Contains all api based information and methods
db.py
All database methods, including pulling and pushing data
Auith.py
Has all methods involved in making accounts
Main.py
Serves as link between front end and backend
login.html
create_item.html
home.html
profile.html

SITE MAP


index.html
+ Learn More: about
Get Started: Login
if not logged in
login.html
+ login: stuy.edu and password
create.html
+ name
+ password
+ confirm password
+ email
upload.html
+ name
+ price
+ description
+ price
profile.html
+ view previous transactions
+ create/change/delete user's items
+ field: type

DB SCHEMA
User
user_id (unique identifier) | password (hashed) | admin | name | email (hashed)

user_id	password	admin	name	email
5	“df98fhsofj4ioc”	True		

Item
item_id (unique identifier) | item_name | price | description | status | is_selling | user_id

item_id	item_name	price	description	status	is_selling	user_id
1	“MetroCard”	39.40	“Sold”	2*	True	5
0	“Set of gym clothes”	10.29	“Want new”	1*	False	5
15	“iPhone”	650	“On sale”	0*	True	5

* status: {0: up for sale, 1: meeting arranged, 2: completed}

Picture
picture_id (unique identifier) | item_id | path

picture_id	item_id	path
3	1	“/item_img/3.jpg”

APIs
Description

- Gmail API - https://developers.google.com/gmail/api/guides/
    - Used to send people emails alerting them of a sale of their item as well as details of the sale, like where to meet up and when.

Usage (how we’ll be using them)

We will be using the Gmail API to interact with the users of the website. When a buyer wants to buy something from a seller, the Gmail API will act as an intermediary, sending the seller an email message.

PLAN
MVP

Task	**Assigned To**
Set up project directory	Ryan
Login/create users ● Login and create views, templates ● Authentication methods ● DB methods	Andrew/Edward
Foundation/CSS ● Base template	Ryan/Edward
Homepage ● Public view + template ● Logged in view + template ○ Refresh button	Edward/Max
Upload items ● Upload view, template (form) ● DB methods ● Modify homepage to show uploaded items	Andrew
Purchasing process + meetup system ● Item-specific view, template ● Items listed in a page → click leads to the item page ● Item page: button for meetup request → modal for email message ● Message is sent via Gmail API to the seller ● Seller/buyer can mark an item as bought/found once the item is bought → archives it	Ryan
User Profile ● Profile template, view ● View user-created items and previous transactions ● Links to delete, create, change status of user-owned items ● User information	Max
Search and Filter ● Modify homepage to allow for search,	Ryan

filters ● Modify user profile to allow for search	
Admin Interfaces ● View, edit, delete (moderate) items ● View, delete users	Edward

Stretch

- Imgur API for storing pictures
- Rating system, view comments on profile (reviews for items sold, +rep)
- Editing items
- Bidding system (choose set price or bid)
- Maps of school, user can click on a spot on the map to indicate meetup location
    - Choose in modal
- Online payment (Stripe API - https://stripe.com/docs/api)
- Price check items (Amazon Product Advertising API - http://docs.aws.amazon.com/AWSECommerceService/latest/DG/EX_RetrievingPriceInformation.html)
- Adding item requests