Design Doc: Team 2003 Honda Civic

Jack Schluger, Sebastian Cain, Noah Fichter, Chloe Delfau, Asher Lasday

Roles:

Noah - API Management

Jack - Database Management

Chloe - app.py and backend

Sebastian - frontend

Asher - Project Manager

Database Schema:

We will have one database: database.db. In this database.bd, there will be a few tables:
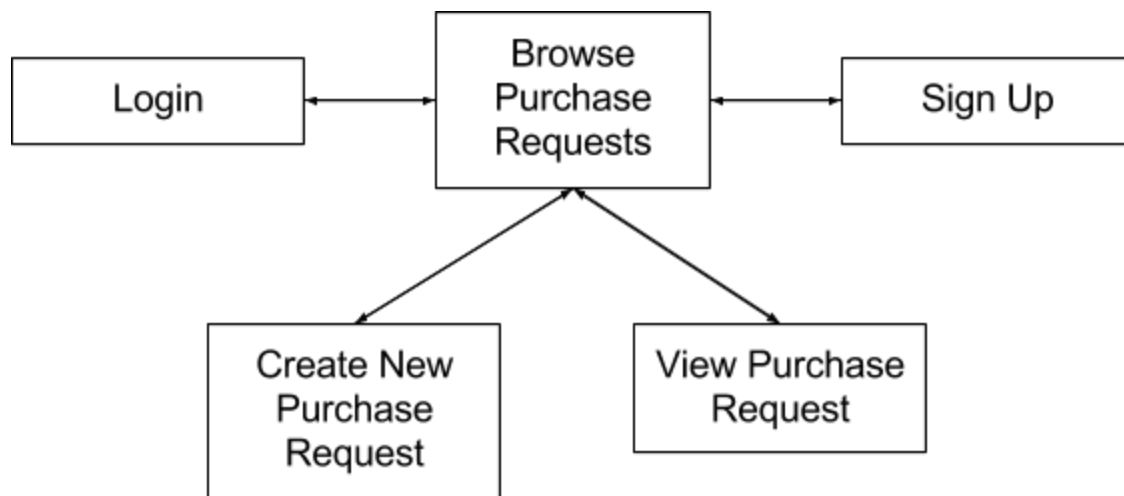
- users
    - username - TEXT
    - password - TEXT
    - userId - INTEGER
- posts
    - userId - INTEGER
    - postId - INTEGER
    - name - TEXT
    - startingPrice - INTEGER
    - expiration - date?
- bids
    - postId - INTEGER
    - bidId - INTEGER
    - userID - INTEGER
    - price - INTEGER
- sales
    - saleId - INTEGER
    - date - date?
    - price - INTEGER

I will use the file utils/dbManager.py to interact with the database. There will be the following methods in this file:

- login

- register
- post (post an item for to buy)
- bid
- logSale
- search
- Other Stuff that we need as we build the project

Site Map:

API Information:

APIs to be used: PayPal, Clarifai, Cloudinary

PayPal: used to create an invoice from the seller to the buyer

Functions: create invoice, check if invoice has been paid, create shipping label (extra feature)

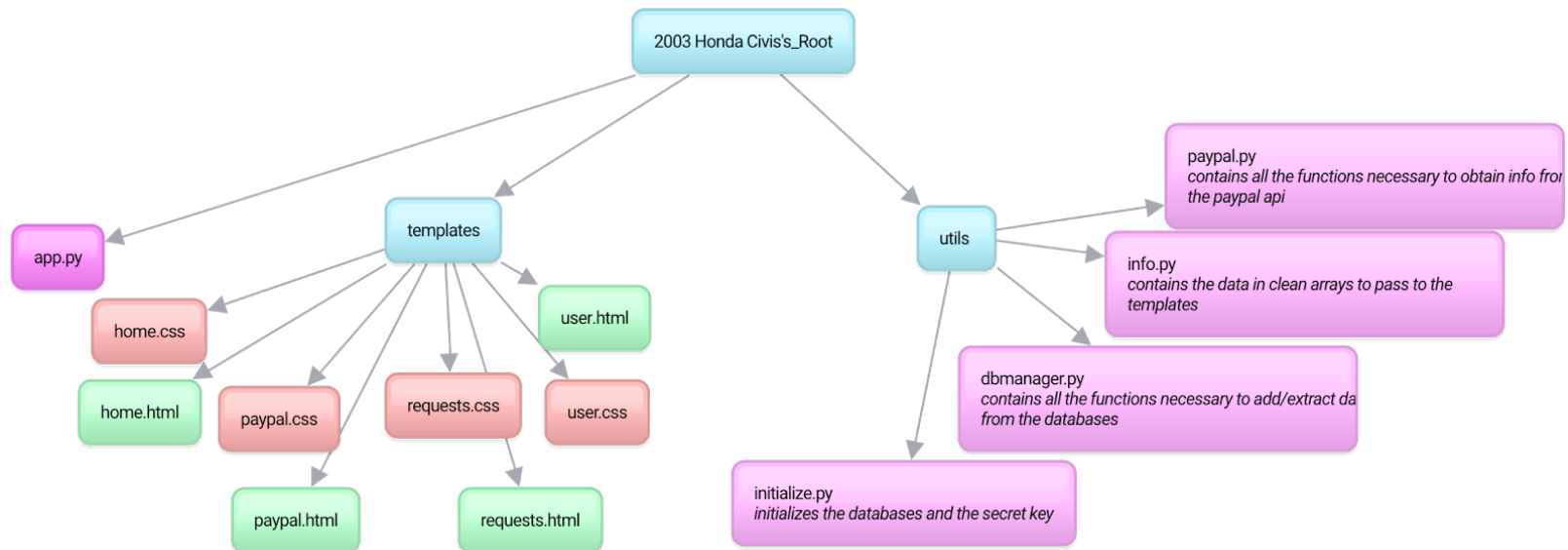Clarifai: used to tag pictures to allow users to find items better

Functions: tag an image

Cloudinary: used to host photos

Functions: upload an image, retrieve an image

API calls will be done with requests library

File Tree:


```
2003 Honda Civis's_Root
    ├── app.py
    ├── templates
    │   ├── home.css
    │   ├── home.html
    │   ├── paypal.css
    │   │   └── paypal.html
    │   ├── requests.css
    │   │   └── requests.html
    │   ├── user.html
    │   └── user.css
    └── utils
        ├── paypal.py
        │   contains all the functions necessary to obtain info from the paypal api
        ├── info.py
        │   contains the data in clean arrays to pass to the templates
        ├── dbmanager.py
        │   contains all the functions necessary to add/extract data from the databases
        └── initialize.py
            initializes the databases and the secret key
```

Examples of potential routes:
- Home page: @app.route("/")
  - Includes a login and registration form
  - These forms will be where a user is redirected to if (s)he tries to buy/sell something without being logged in
- User account: @app.route("/user")
  - Shows all account information
    - Items bought
    - Items sold
    - Buy requests
    - Sees items you've saved (potential save for later option)
- Buy requests: @app.route("/requests")
  - Shows all posts of requests people have made and prices they're willing to pay
- Paypal: @app.route("/paypal")
  - Shows all relevant information about your paypal account
    - All your transactions
    - How much money is on your account