

GameTrade

Work_with_db

(gsread, google-python-api, sqlalchemy):

- Первые строки для создания и использования google-таблице с ключами
- Order - класс заказа - (id, time, total, client_id - внешний ключ)
- ShopClient - класс клиента (у yoomoney уже есть Client) - (id, name, email, password (hashed)) - есть проверка и добавления пароля (check_password и set_password соответственно)
- Game - класс игра - (id, name, price, picture, описание (description)) - есть получение данных google-таблицы и получение+удаление ключа для выдачи покупателю.

Client_forms (wtforms):

- LoginForm - email, password, remember_me, submit
- RegisterForm - name, email, password, password_again, submit
- ResetPasswordForm - password, password_again, submit

db_session - подключение к, создание БД

Games-api

- delete_game (/api/delete_game?password={password}&id={needed_id}) - удаляет игру из бд по id.
- reset_game (/api/reset_game?password={password}&name={game name}&price={game price}&image={image name: must be in img dir}&info={game info}) - добавляет игру обратно в бд.

Static:

- img - картинки
- assets - дизайн (.css, bootstrap, .js)
- favicon

data:

- shop-market.db - база данных

Templates:

- 404 - обработка ошибки 404
- base - базовый (шаблон)
- cart - корзина
- forgot-password - страница восстановления пароля
- games - каталог
- index - главная страница
- letter - письмо, но сейчас уже не нужно
- login2 - страница входа
- pay - страница с ссылкой на оплату
- profile - страница профиля клиента с историей заказов
- register2 - страница регистрации
- reset-password - страница нового пароля (продолжение восстановления после получения письма на почту)

app.py

- load_user - flask_login - подгрузка пользователя из бд
- main_page(GET) - главная
- login_page(POST, GET) - вход
- logout - выход

- to_list - остался после проблем с ast на python.anywhere: преобразует список из cookie (записывается json.dump) в нормальный список.
- register_page(POST, GET) - регистрация
- forgotten_password_page(POST, GET) - начало восстановления пароля: отправляется письмо с ссылкой на почту для нового пароля
- reset_password(POST, GET) - новый пароль
- show_games(POST, GET) - каталог с сортировкой, поиском, добавлением в корзину.
- show_profile - профиль с историей заказов
- show_cart - корзина: оплата, удаление товара
- get_payment - страница, на которую клиент попадает после оплаты (ссылка прописана в uoomoney): проверяется оплата через историю переводов по label: выдаются ключи на почту/прямо на сайте.
- uoo_money - создание платежа на физлицо: вывод ссылки на оплату.
- handle_404 - обработка ошибки 404