

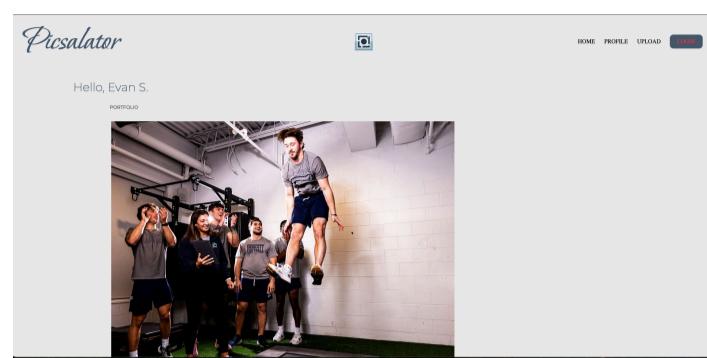
# Picsalator



HOME PROFILE UPLOAD LOGIN

Frontend and backend additions:

- Add a settings page or panel that allows users to logout, check sign in, compression settings, and color theme.
- The current sqlite3 storage for larger image files is limited, so I will figure out a better way to do that. I may have to use mongoDB or something.
- Sharing capabilities for any photo the user has access to, they can email it or share it with other users in local storage.
- There will be a drop down on the main home page that allows users to switch views between their photos, shared photos, or a combination of both.
- Allow users to click on any image at any time and have it be full screen full quality.
- make any photo downloadable.
- Delete images, comment under images in local storage.
- User "blobs" database



## Profile Page

Frontend: user will see their name, their uploaded content, content shared with them, and compressed files (specific for IG posts).

Backend: profile.php that communicates with picsalator.db > images table

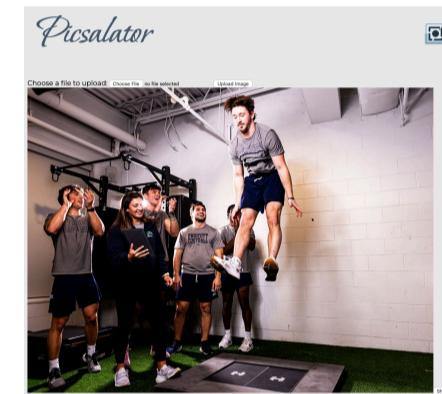
Text



## User Sign up / Login

Frontend: users decides to login or sign up, and then follows the prompts to do which ever they go with. They are able to login once they sign up.

Backend: saves username, password in sqlite with the images, and will use authentication and hashing.



## Upload Page

Frontend: User will be able to press upload button, select an image from their hard drive, and it will show up on the page.

Backend: upload.php, and communicates with the computer's hard drive and the picsalator.db sqlite3 database and uploads.db

api.php, client.html, profile.php, upload.php, login.php, authenticate.php, settings.php