**Pixture project documentation**

Team name: Aksuallit

Team members: Amanuel Ayezabu, Ville Liekari, Daniel Finnerman

Introduction

Pixture is a picture sharing application, where all users can browse pictures and registered users can post and share them. Registered users can also like the pictures. The theme of the pictures is meant to be artistic. The main target group is users who want to share their artistic photos, for example from the wildlife or art.

Users can stay anonymous behind the nickname of their choice. The website is made adaptive for both mobile and desktop use.

User stories for Pixture app:

1. As a photographer, I want to upload my photos online with to share my hobby and interest with others
2. As a travelling person, I want to share my photos and go back to memorizing the moments
3. As an artist, I want to share my paintings with others and get ideas for future paintings and drawings
4. As a photographer, I want to share my neatest photos anytime and everywhere while staying anonymous behind my nickname
5. As an artist, I want to be able to swiftly access my uploaded photos with both mobile and desktop

Requirements:

When a user opens the website, there is an option to sign up, and sign in after the account is created. The account and its password will be saved to mysql database and password will be crypted with bcrypt as a hash.

Users can browse the pictures on the website and upload them if they are signed in. The likes are saved to the pictures mysql table. All users can like the pictures. The pictures are uploaded to the database and rescaled to smaller thumbnails so they won’t take too much space from the server storage. The picture names are changed to something else with multer.

Pictures are uploaded to the website with information from their title, description and owner. The owner is the username of the user. All these attributes are saved on a database after the user presses upload button.

The website has a modern UI, which is made adaptive with CSS for mobile and desktop usage, as seen on the mock up down below.

The website is deployed to a node express server on a remote linux machine. The node.js app is making it run continously. The website is deployed on HTTPS, which is the secure way to send data between the server and browser.

Mock-ups from the website: 