CMPG323 PROJECT 2

POSTALOT APP DOCUMENTATION

L ANTHONY	**32969694**
JRG KLERCK	**28999146**

SYSTEM DESIGN AND OVERVIEW

Thesis submitted for the degree BSc in Information Technology at the North-West University

Supervisor:	Prof JT Janse van Rensburg
Co-supervisor:	Mr. Zander Boonzaaier

Graduation: 2021

`

TABLE OF CONTENTS

`

1.1 LIST OF FIGURES

`

1.2 INTRODUCTION AND SYSTEM OVERVIEW

The project brief was to create an image sharing platform where users could upload images, share images with other users and edit the associated metadata. We determined that the best way to accomplish this was by developing a front-end for the users to interact with, and an API server to handle data requests.

1.3 TECHNOLOGY STACK

Our system is built with Node.js with a React based frontend and a Loopback framework-based API server for the backend. Both the front and backend components are hosted on Heroku. Our PostgreSQL database is hosted on Amazon Web Services and provisioned by Heroku. Our image files are saved to Cloudinary.

1.4 TEAM DYNAMICS

1.4.1 ROLES AND TEAM FUNCTIONING

We split the group into two groups, and Llewellyn built the React Frontend, while John built the Loopback Backend and setup the database. When one member had a problem that they could not solve on their own, a meeting was called to help each other out.

1.4.2 COMMUNICATION

Communication was done primarily over Discord through regular meetings in the TheBlankCompany Discord channel. During these meetings, planning was done, along with resolving issues with deployment along with any other issues that might have arose during development.

1.4.3 SOURCE CONTROL

GitHub was used as Source Control. Because the Front and backend were hosted and developed apart from one-another, two GitHub repositories were used. A Dev branch was used for development, and successful changes were merged into the Main branches. The Main branches were linked to Heroku by means of Automatic Deployment, so that the latest successful changes were always live.

1.4.4 CONTRIBUTIONS/TASKS PER MEMBER

Llewellyn focused primarily on building the React Front-end that the users interact with. The front-end captures the user’s data, formulates an API-request and makes this request to the server.

John focused primarily on building the Loopback API-server that would interact with the database for data storage and data-manipulation. It would execute requests and return the right data to the front-end.

Both members contributed to the creation of the Cloudinary account for saving the image files.

1.4.5 BENEFITS/CHALLENGES OF THE TEAM APPROACH

The main benefit to the team approach was the ability to divide the labour between members so that each member could focus on their strengths. The main challenge was executing proper planning. Because team members were not able to meet up due to the Covid-19 pandemic, planning the project took longer than we anticipated.

`

1.5 ENTITY RELATIONSHIP DIAGRAM

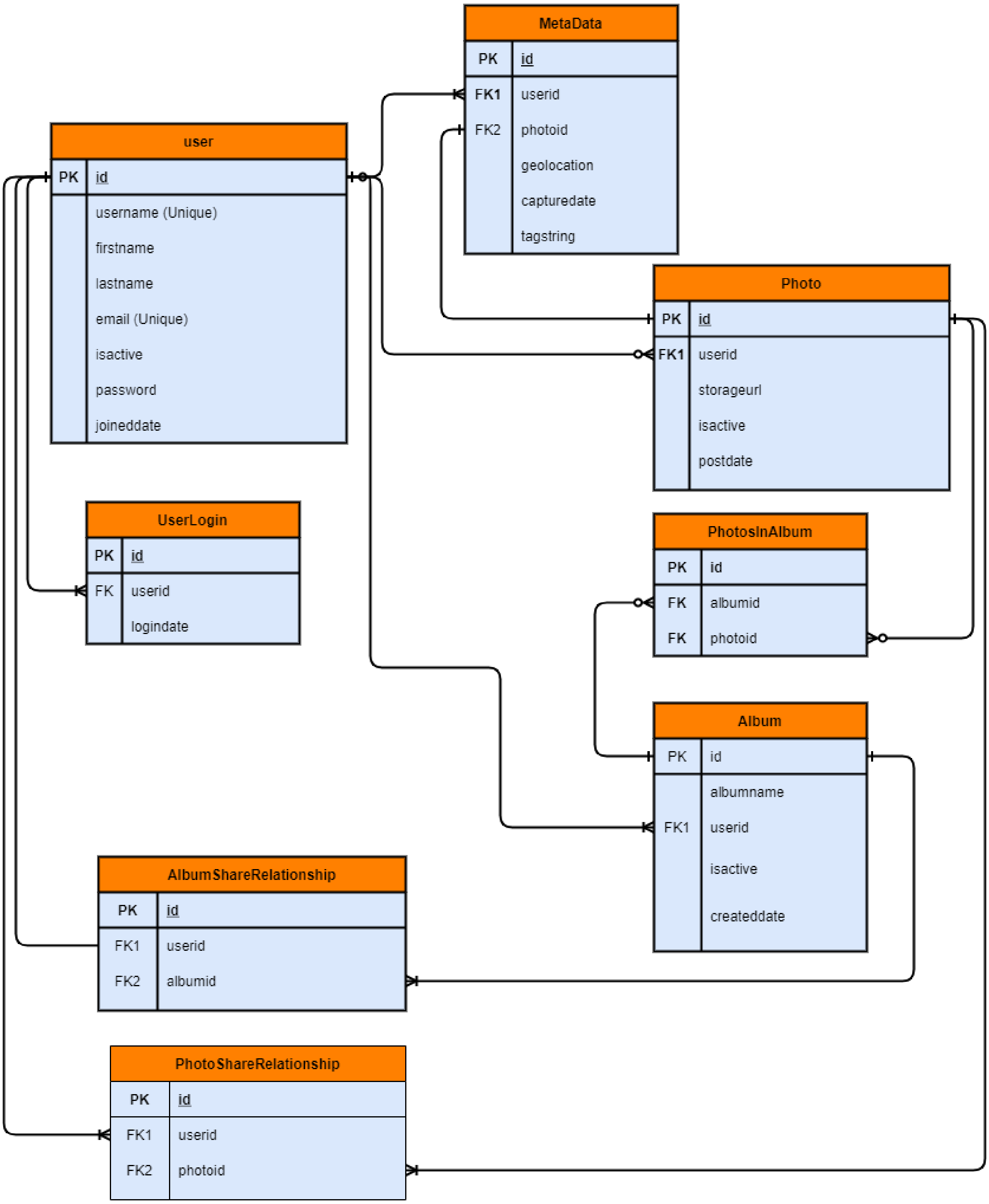


Figure 1-1: Entity Relationship Diagram

`

1.6 USE-CASE DIAGRAMS

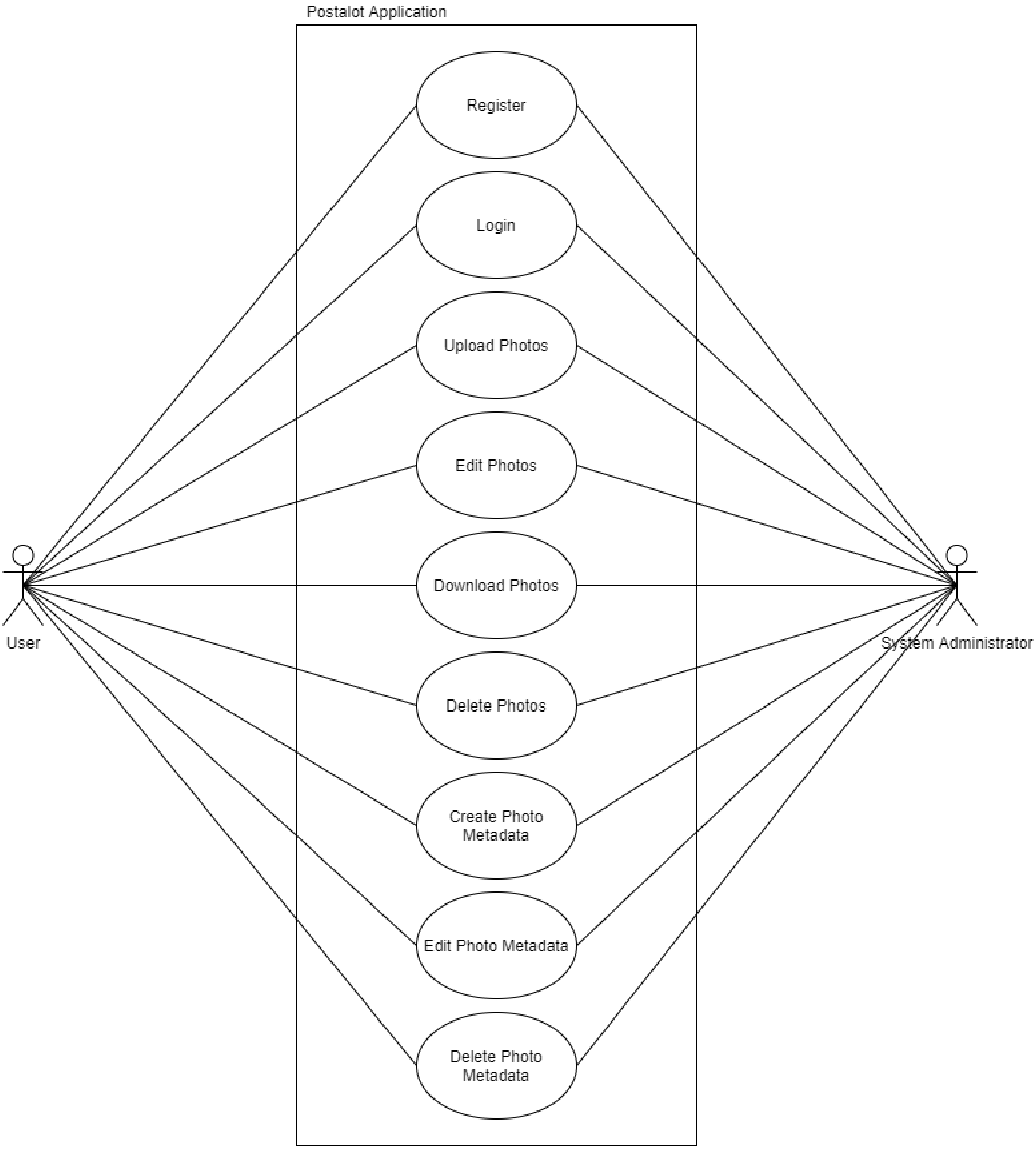
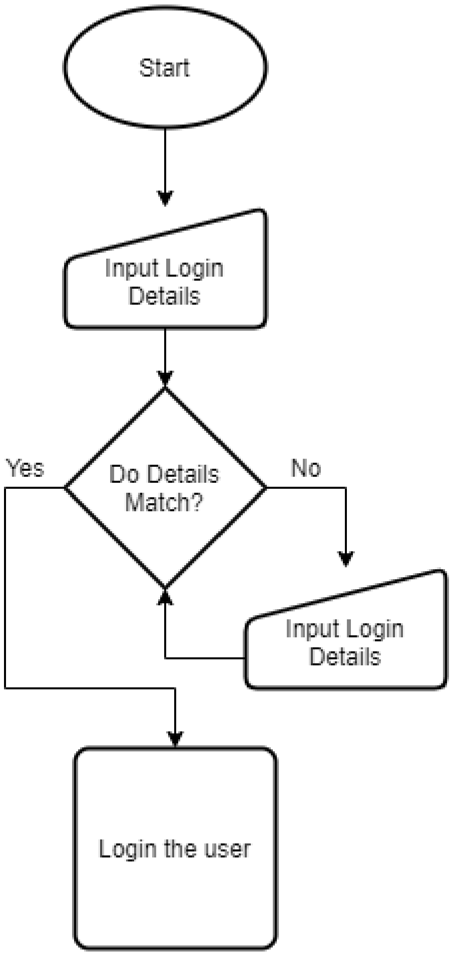


Figure 1-2: Use Case Diagram

`

1.7 DATA FLOW DIAGRAMS



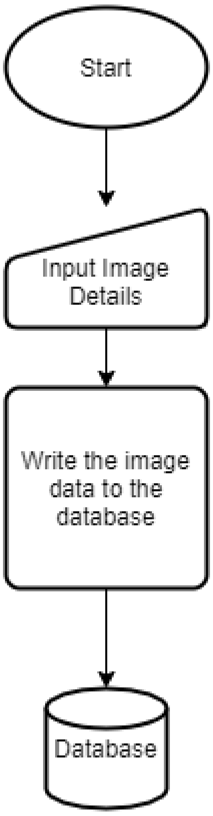


Figure 1-3:Upload Image Data Flow Diagram

Figure 1-4: Login to System Data Flow Diagram

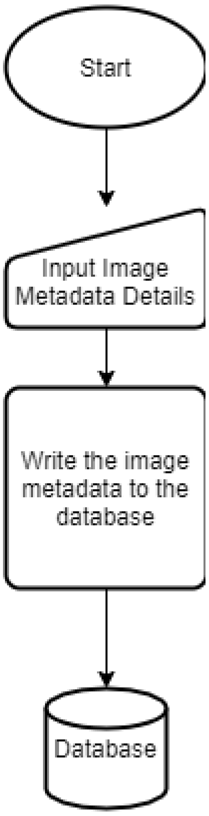

Start
Input Image Metadata Details
Write the image metadata to the database
Database

Figure 1-7: Upload Metadata Data Flow Diagram

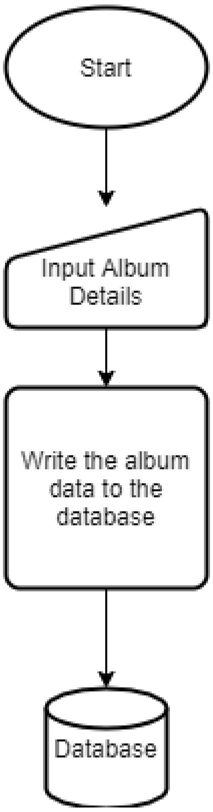


Figure 1-6: Create Album Data Flow Diagram

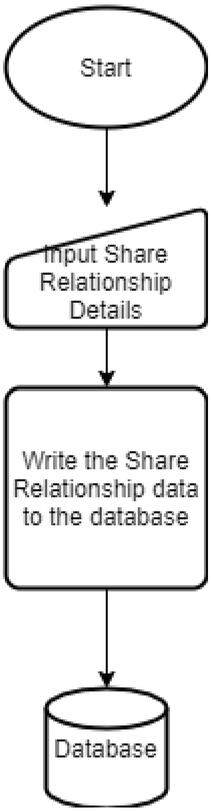

Start
Input Share Relationship Details
Write the Share Relationship data to the database
Database

Figure 1-5: Create Share Relationship Data Flow Diagram

`

1.8 USER MANUAL

In your web browser, navigate to postalot-client.herokuapp.com. You will see the login page. If you have an account, use these details to log into your account. If not, select the “Not a user? Register Here” option showed in the figure below.

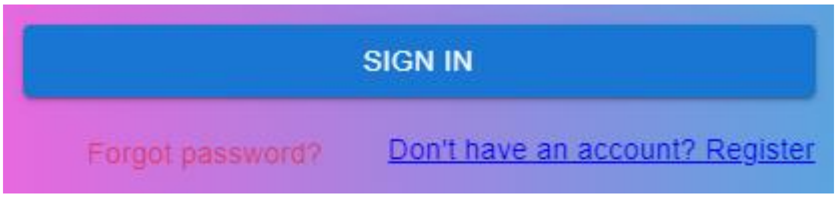


Figure 1-8: Sign in Button and Register Here Option

You will see images shared with you on your home page. To upload a photo, select the ‘Choose File’ button on the homepage. Select your image and Input the Metadata you would like to attach to the image, including tags you would like to use to search for. Hit the upload button, and your image will be saved!

Planned Functionality:

To share this photo with your friends, select the “paper plane” showed below. Search for the user you would like to share the image with, and press share. They will now be able to view your image and edit its metadata.

Figure 1-9: Share Button

To edit the metadata of an image on your home screen, select the three-dot button, and select the “Edit Metadata” option. Enter the new data you would like to save and hit the save button.

To create an album, head over to the ‘album’ button shown below. Enter the name for your album and select the images you would like to save to the album. If you would like to share this album with friends, select the share button, search for your friend and select the save button. Now they too can view your newly created album!

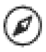


Figure 1-10: Album Button

`

1.9 REFERENCES

-

1.10 Links

- Server-side GitHub Repo: https://bit.ly/PostalotServerGit
- Client-side GitHub Repo: https://bit.ly/PostalotClientGit
- Old GitHub Repo(used from 12 October – 26 October): https://bit.ly/PostalotOldGit
- Postalot Live Version: https://bit.ly/PostalotClient
- Postalot Video Demo: https://bit.ly/PostalotVideoDemo