

FUNCTIONAL SPECIFICATION

SIMILAR CLOTHES SEARCH

[Dovydas Baranauskas 15336731]
SCHOOL OF COMPUTING, DCU

Functional Specification Contents

CONTENTS	
1. Introduction	2
2. General Description.....	3
3. Functional Requirements	8
4. System Architecture.....	11
5. High-Level Design.....	12
6. Preliminary Schedule.....	14
7. Appendices.....	15

1. INTRODUCTION

1.1 Overview

This will be a web application where a user can upload a photo of an article of clothing, and they will be recommended a similar article of clothing based on the image uploaded. The web application will then show you where to buy the similar clothes from. The system will use deep learning algorithms to train the model to recognise and rank the similar images.

1.2 Glossary

Studio photo – A photo that is taken in a studio in high definition and with a white background, having a high level of contrast.

API - a set of functions and procedures that allow the creation of applications which access the features or data of an operating system, application, or other service.

Web Application - An application program that is stored on a remote server and delivered over the Internet through a browser interface.

Euclidean distance – In mathematics, the Euclidean distance or Euclidean metric is the "ordinary" straight-line distance between two points in Euclidean space.

Transfer learning - Transfer learning is a research problem in machine learning that focuses on storing knowledge gained while solving one problem and applying it to a different but related problem.

Deep learning - Deep learning is part of a broader family of machine learning methods based on learning data representations, as opposed to task-specific algorithms.

Database - A database is an organized collection of data, generally stored and accessed electronically from a computer searches.

2. GENERAL DESCRIPTION

2.1 Product / System Functions

The system will use convolutional neural networks to analyse an uploaded image of an article of clothing and find similar articles in the database. The system will use a deep ranking search algorithm to detect fine grain similarity in the images. After the system has found the similar images, it will recommend them to you based on how close they are to the original image.

The user will upload an image they have taken or found of an article of clothing that they like and the web application will then analyse it and return the results. The user will be able to take a picture of clothing and upload the images to the web application and it will try to retrieve images similar to that image. This will eliminate the need for the user to extensively search multiple websites for just one similar picture, while the application will show multiple similar images in one place.

2.2 User Characteristics and Objectives

The user community will be any person with basic computing skills and an interest in finding something they want without needing to extensively search multiple websites and sources to find just one similar article, the web application will be able to display multiple similar articles in the same place. The user will just have to have a picture of the article of clothing that they want to find similarities to, or will have to download an Android application and take a photo to be uploaded to the web application.

Desirable Characteristics:

1. Upload any image of clothing to get recommendations.
2. Take an image with an android app and get recommendations.
3. Have a payments API to allow users to pay for items without needing to leave the app.
4. Have a recommender system to recommend clothes to users based on previous systems

Feasible solutions:

1. User uploaded images will be studio with a white background so there is high contrast.
2. Create a log in system for users to log in and see their uploaded images.
3. Using affiliate marketing so we could make profit if a user uses our link to buy items on redirected websites.

User stories:

Name: John Smith	Story: John is a student at Ireland University studying a business and computing course. He has strong computational skills from his course, and is interested in the latest fashion trends. John also works a part time job to finance his student life.
Demographic: Age: 21 Occupation: Student Technology skill: Strong	System expectation: <ul style="list-style-type: none">• John wants to find an article of clothing similar to that of a popular retailer.• Cannot afford to pay the high end retail price so needs to find cheaper alternatives for big brand items.

Name: Ann Apples	Story: Ann is a mother of two children who are both in college. She uses computers everyday but only for online browsing therefore has very basic technical skills.
Demographic: Age: 47 Occupation: Store manager	System expectation: <ul style="list-style-type: none">• Ann finds a dress she likes at a store but it's not the exact one she wants.• She wants to find a similar fit of the

Technology skill: Weak	dress but in a different design. <ul style="list-style-type: none"> • She wants to be able to take a picture and see similar dresses returned to her.
------------------------	--

2.3 Operational Scenarios

Scenario ID	1
User objective	Create account to access system.
User Action	A user enters a valid email, username and password and create an account.
Comments	User must have a valid email in case they forget their username or password.

Scenario ID	2
User objective	Log into system.
User Action	User types in his username and password and is authenticated, if successful, he is allowed access to the web application.
Comments	If a user is not registered he will be allowed to create an account and re attempt to log in.

Scenario ID	3
User objective	Upload image for recommendation.
User Action	User chooses image of clothing they want. User clicks 'upload image'. Chooses image from their pc.
Comments	The user will have to have the image file downloaded or uploaded to their pc before they can upload it to the web app.

Scenario ID	4
User objective	Take photo of article of clothing for recognition.
User Action	A user will see a piece of clothing that they like. They will take a photo with an android and send it to the web application to be recommend images.
Comments	The user will have to make sure the image of the clothing is clear, or it will not be able to return accurate results.

Scenario ID	5
User objective	User explores the similar clothes retrieved.
User Action	The user can explore all the retrieved images by the algorithm. A number of images will be returned.
Comments	The user may not like the first most similar image returned so it's important to return a few so the user can browse.

Scenario ID	6
User objective	Click image and get redirected to website to buy clothing.
User Action	User chooses image of clothing they like the most. They click on the image and get redirected to the website to buy the image.
Comments	We may use affiliate marketing to make money if a user buys a piece of clothing from the redirected website.

Scenario ID	7
User objective	Wants to look at previously uploaded images.
User Action	Navigates to previous uploads tab, if the user previously uploaded images they will be displayed here.
Comments	The user will have to be logged in to access the web application, so they should always be able to see their previous uploads.

2.4 Constraints

There are a few constraints to consider when implementing this application.

1. How accurate the similarity search is very important. If the results are not accurate the application becomes pointless.
2. The image clarity is also a constraint. If the image is very unclear and the algorithm will not be able to produce similar results.
3. Linking back to stores where users can buy the products will be a constraint because links tend to depreciate fairly quickly.
4. Speed of training is also a constraint as the dataset is very large and we will need a lot of computational power for training.

3. FUNCTIONAL REQUIREMENTS

Description	The web application must allow user to log into the web application.
Criticality	If a user is unable to log in or create an account, he will not be able to access the web application.
Technical issues	If a user forgets his password or username, there should be a system implemented to allow the user to retrieve their username and password.
Dependencies	The user will have to have created an account before attempting to log in to the system.

Description	The web application must allow users to upload any image from their computer.
Criticality	If a user is unable to upload an image to the web application, the application will have no input and therefore will not have any functionality.
Technical issues	This involves constructing a web application that allows user to upload a file, make sure the file is an image file and handle it correctly to pass it to then next stage of the web application.
Dependencies	N/A

Description	The web application must allow users to upload any image from their android device.
Criticality	A user should be easily able to take a picture of clothing or upload a piece of clothing from their phone.
Technical issues	Our matching algorithm will have to recognise the edges of the clothing so that we can make accurate similarity searches.
Dependencies	We will need the user to take a clear picture of the article of clothing so that we can detect

	similar clothes.
--	------------------

Description	The web application must be able to analyse the image and find similar clothes based on similarity using Euclidean distance.
Criticality	If the algorithm is unable to accurately find similar images, the application becomes useless as it returns irrelevant information.
Technical issues	I will need to make sure that the algorithm is accurate and returns relevant information. This will be measured with Euclidean distance and user testing.
Dependencies	For the algorithm to work, we need to make sure the image uploaded by the user is a valid image, and in an image extension.

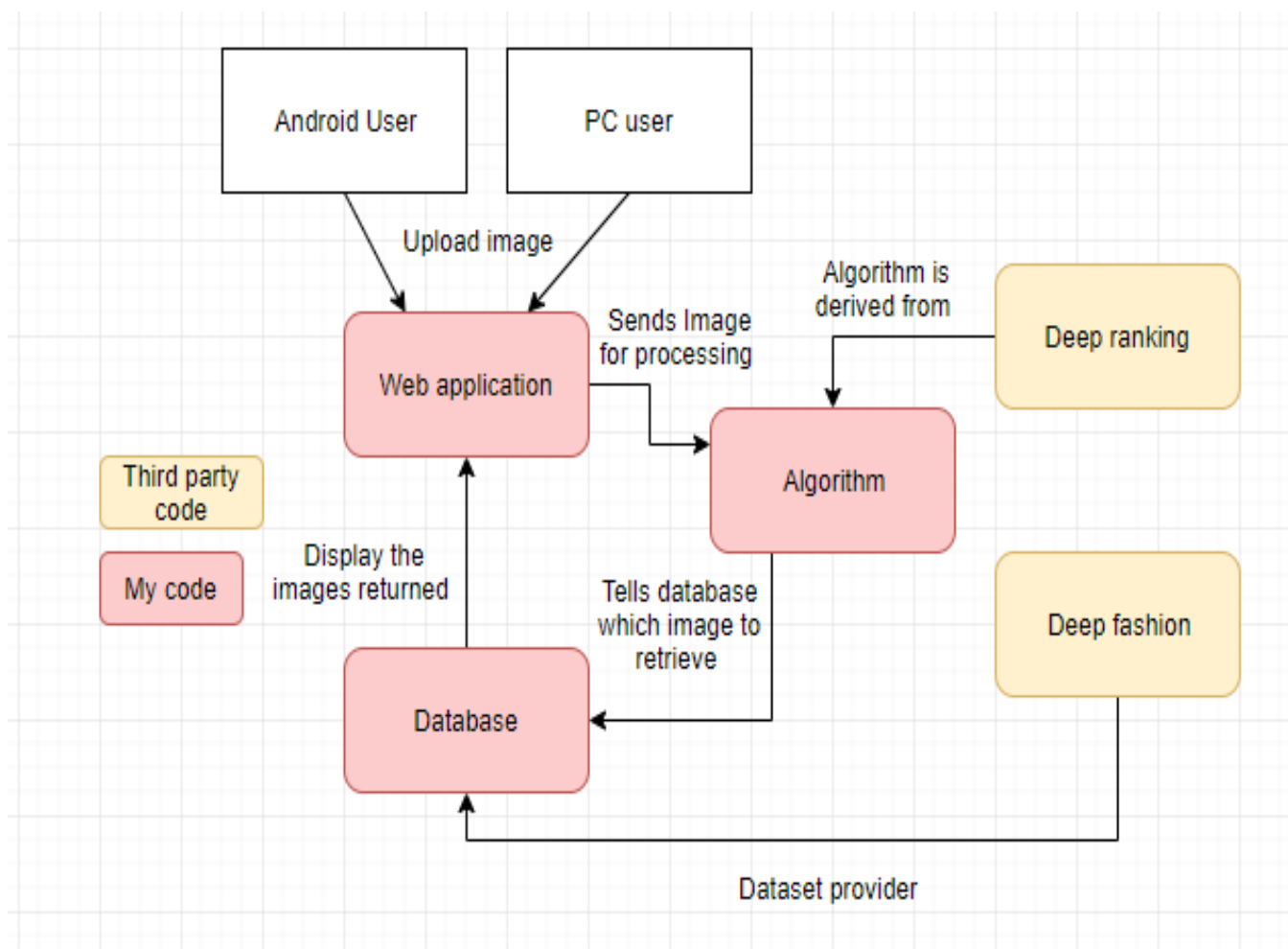
Description	The web application will be able to link you to a place to purchase the item of clothing.
Criticality	This is not that critical part of the project as users can reverse image search the images themselves, but it is important if we want to incorporate affiliate marketing.
Technical issues	It will be hard to keep the database up to date with the latest links as links usually expire.
Dependencies	For the link to work, the image link must be valid.

Description	System must be able to keep links in database up to date
Criticality	If the link becomes out of date the image will not be able to be searched.
Technical issues	I will need to be able to check if a link is out dated and remove it, or try and updated with a new link, it will be hard to replace a link once it

	becomes invalid.
Dependencies	N/A

4. SYSTEM ARCHITECTURE

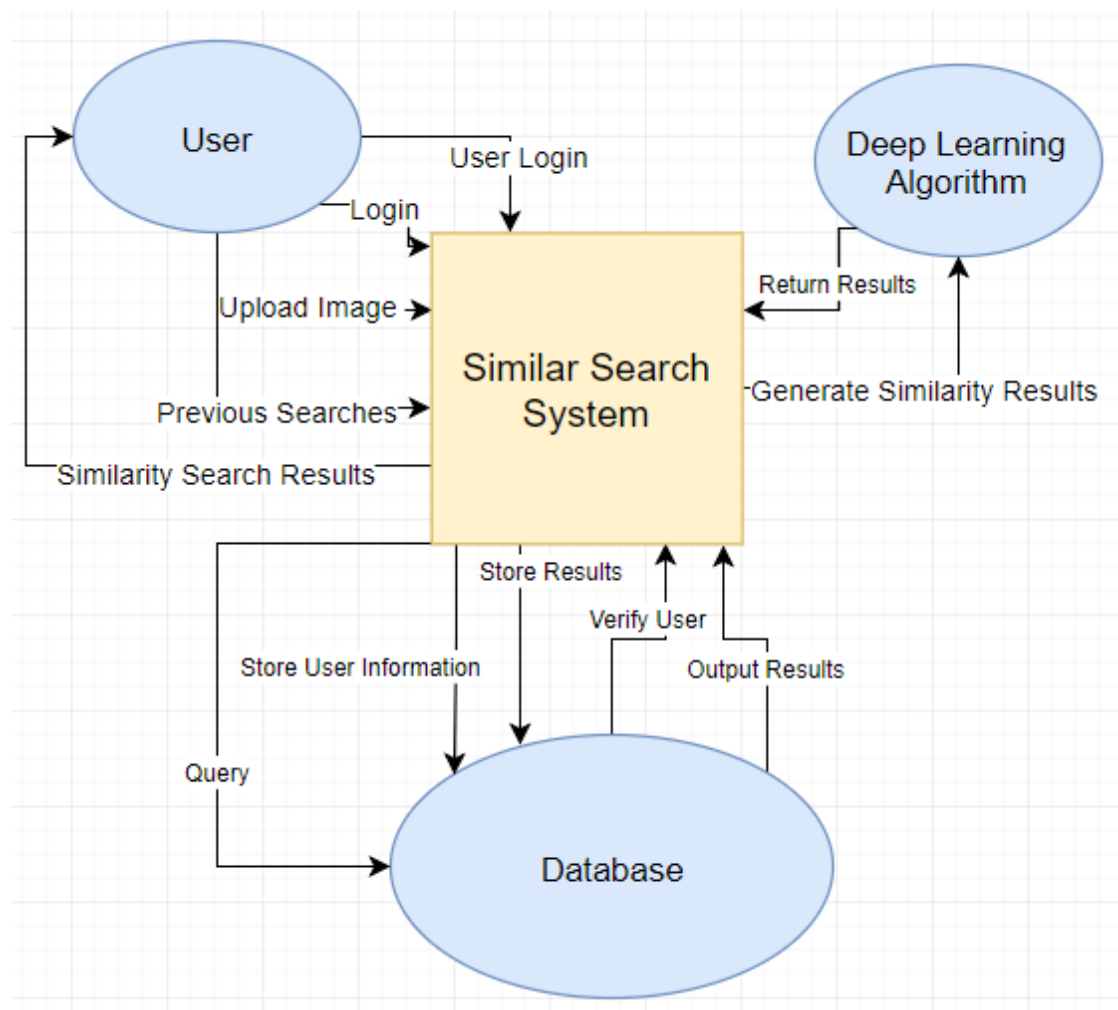
The user will interact with the system by using an android device, or by pc. The user will upload a picture to our system. The system will then take the photo and analyse it and compare it against our database of images. The algorithm will then send back the images that are most similar, and they will be displayed on the user's device. I will use the algorithm described in the deep ranking paper, and I will use the deep fashion data set to train the network. I will use the keras library for creating and training the neural network as it will save a lot of time by allowing me to use pre-trained networks for transfer learning as I have a very large dataset. I will be using the flask framework for making my web application. It will allow me to create the web application that will be accessible through android devices and will allow customers to upload images from their phones.



5. HIGH-LEVEL DESIGN

This section should set out the high-level design of the system. It should include one or more system models showing the relationship between system components and the systems and its environment. These might be object-models, DFD, etc.

Context Diagram:

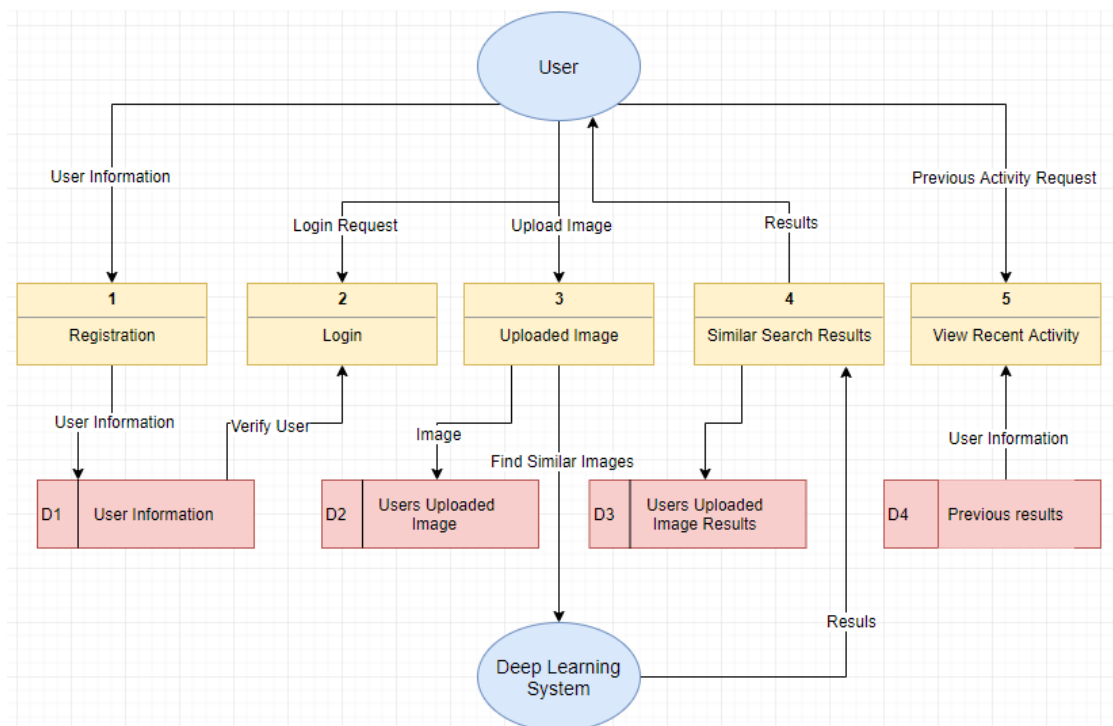


The Context diagram above describes how the system will interact with each of the major components. The user will interact with the system by registering for an account and logging in, they will upload an image, or look at their previous searches. The system will show the user the results for the query image.

The system will send the images to generate similar images by applying the similarity search algorithm. The algorithm will then return the images that are similar to the system.

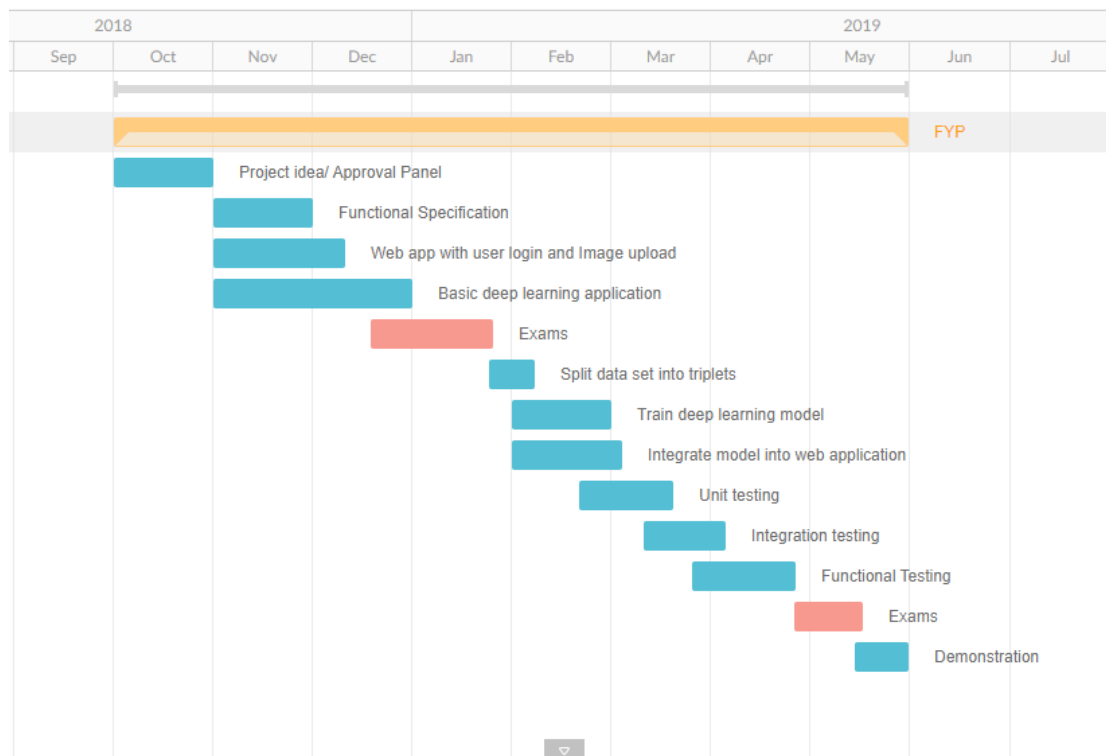
The system will send the queried image to be store in the database, the system will send registered user information to be stored in the database. We will also send the results of the queried image. The database will also be used to verify the user and to output result links of the queried images.

Data Flow Diagram:



The data flow diagram shows us in more detail than the context diagram, how the data will flow through our system. We see how the data enters and leaves the system and where the data will be stored. We see from the diagram that the user will enter their information, log in, upload and image, and request results. The system will store the user uploads, user details, and the results. A deep learning system will be used to generate the correct image similarity results.

6. PRELIMINARY SCHEDULE



The preliminary schedule is shown above. The web application development has commenced, and hope to have the basic user features finished in the next few weeks that allows users to log in and upload an image. Alongside this I am starting the machine learning pipeline. By January I hope to have a bare bones version of the system working. By April I hope to have most of the system working so that I can focus on testing and bug fixes in the last few weeks. Once the development is finished I can focus solely on the final documentation deliverables.

7. APPENDICES

[1] Deep fashion algorithm paper:

https://users.eecs.northwestern.edu/~jwa368/pdfs/deep_ranking.pdf

[2] Deep fashion algorithm paper:

https://www.cvfoundation.org/openaccess/content_cvpr_2016/papers/Li_u_DeepFashion_Powering_Robust_CVPR_2016_paper.pdf

[3] Flask documentation: <http://flask.pocoo.org>