**Contents**

**Abstract  
  
Acknowledgements  
  
Introduction**

Aim  
Motivation and Rationale  
Objectives  
Timeline  
Plan

**Background Research**

Increase in buying and selling online  
Similar Applications  
 Facebook Marketplace  
 Depop  
 Mallzee

Development Tools

**Design**

**Mobile Layout**Screen Designs

**Website Layout**Navigation system  
Newsfeed  
Profile  
Item Details  
Register and Login  
About Us  
Help  
Trades  
Logo

Mobile  
Adjusted

***Database***

Class diagram  
UML diagram

**Technical Specification**

Version Control  
 Software  
 Backups

**Implementation**

1.0  
 2.0  
 3.0  
 4.0  
 5.0  
 6.0  
 7.0  
 8.0

**Testing**

**Conclusion**

**References**

**Introduction**

**Aim**

For my dissertation, I aim to create a service that enables users to exchange clothing with each other for minimal cost. Everyone has clothes in the cupboard that just never get worn, or just don’t quite fit right and people would benefit from a quick easy way to exchange them.

The aims for the project are:

* To create an online platform for trading clothes
* Keep the trades easy for users to make
* Keep the cost and risk of a making a trade down

**Motivation and rationale**

Swapping/upcycling clothes allows you to keep your wardrobe fresh without the expense of purchasing new clothes regularly. It can also be difficult to find people your clothing size with similar same fashion interests and budget.

I came up with this idea as I regularly use mobile app ‘Depop’ (see section xxx) and receive a lot of messages requesting trades. However due to the risk involved in trading it doesn’t happen very often and I personally wouldn’t risk trading with someone I didn’t know or hadn’t dealt with before as I’ve seen many posts in online groups about users being scammed. The only times trades are successful are when:

1. You meet up in person to exchange
2. You use a middle man (trusted third party who both participants send their items to)
3. One user ships first (just risk and hope the other can be trusted to send their item)

None of these methods are perfect but one of the aims in my project is to make ensure minimal risk and cost to trades and these three ways are options to provide that.

**Objectives**

My objectives to achieve each aim are the following:

* **Research any pre-existing services and evaluate the pros and cons of the features in these**

This includes mobile applications and websites as I am aware of similar services that operate in both of these formats.

* **Build a diagram to show the navigation of the service**

Showing how the pages on the site link together and how to navigate to each page is an important part to how trades will go down.

* **Design and build a service that enables users to trade items of clothing with each other**

This objective best summarises the goal of the project. This is the overall aim, but that said, there are a lot of other features I must enable (for uploads) to make this a possibility.

* **Application should be fully tested to ensure there are no flaws or glitches with any of the features or pages.**

I have added time to make a testing plan on my work plan. When the service is completed I can test the application myself and then get users to test it too. Hopefully I will have time afterwards to improve the service based on the feedback from the testing results.

 **Conduct a full user evaluation that gives information on what imperfections the app has and how it can be more user-friendly**

This should be questions about how easy the service is to use, would you recommend the app, what you would change etc. As services like this can continuously be improved and developed I imagine I will have plans for other features that could be implemented anyway. I intend on asking questions about which of these features users would be most interested in using or what would make the biggest impact.

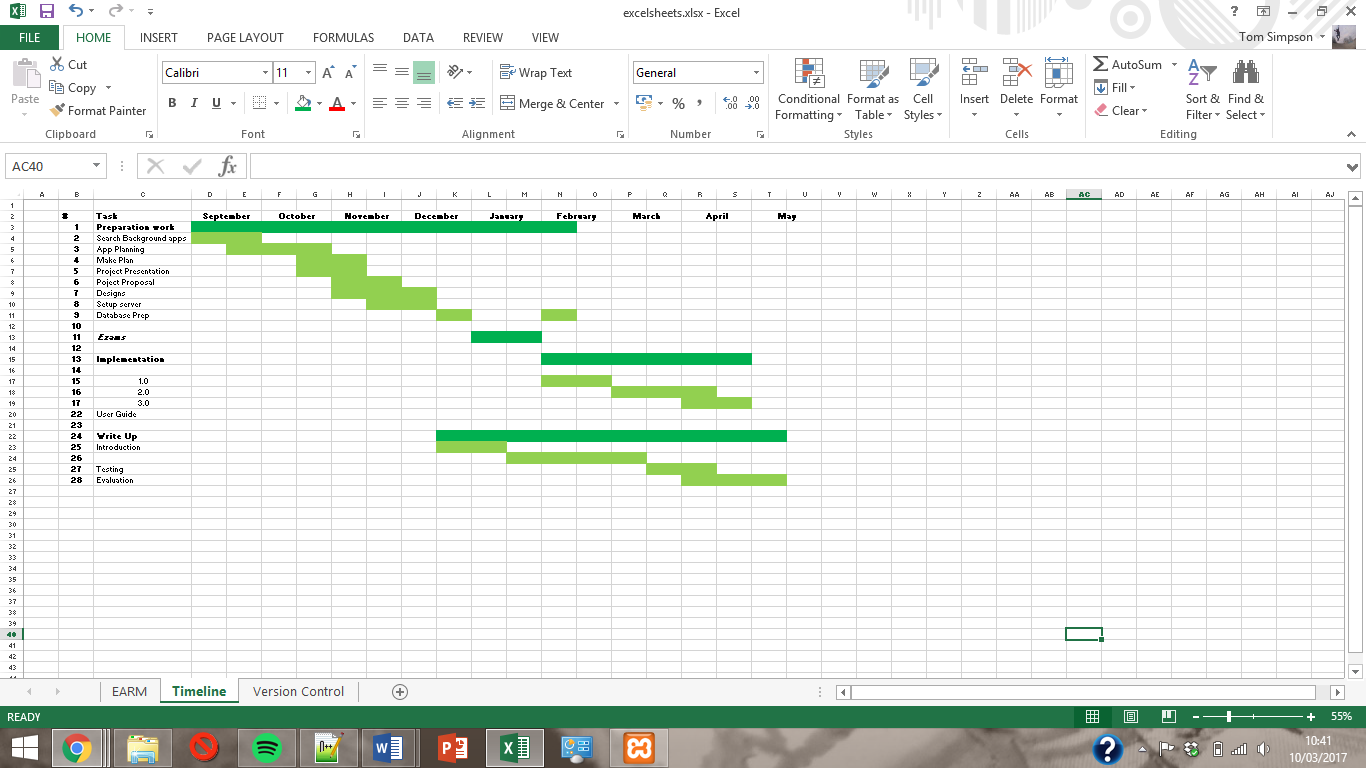
**Issues to resolve**

For this service to be achievable, there are still some complications that I need to find the most ideal way to resolve:

* Providing a safe method for two users to exchange items without the risk of being scammed or losing their item.
* Decide whether to give each item a representative value of its worth (could be monetary or just a rating)
* Best method to provide the service (mobile application or website)

When researching the similar applications, I will consider how these services deal with the same complications and whether any of these solutions will be the most ideal method for switchUP.

**Timeline**

The timeline below shows an approximation of when I will complete specific tasks for the project. I will update the timetable if I become significantly behind or ahead of schedule with a more accurate timescale complete each task. This way if I have issues completing a task I can adjust the timescale so everything is completed before the deadlines.   
  


The tasks under the header ‘preparation work’ includes researching and comparing similar services (mobile and web), planning how the app be navigated and analysing any potential issues that I will need to resolve. From here I will have enough information to start with the designing phase and I aim to have completed most of the research and designing before the Christmas holidays. I won’t have time to further the project during exam season as I have five exams but after this pause I will finish off any design work I have left and begin with the implementation.

The implementation phase should carry on till during the Easter holidays (more precise dates will be on a later plan) I will spend until then getting all the features working including hosting on a server (if website) or making downloadable (putting it on the appropriate app store). With making a website or application, there is always further improvement to make; so the completion of the implementation phase on my plan includes up to the point where the product has met all of the relevant objectives. I will be furthering the write up whilst still implementing the service too as this should make it easier to do and should also mean I have more time for the testing and evaluation phases or to fix any problems discovered in testing before the evaluation stage.

**Background Research**

As everyone knows, buying and selling clothes online has shot up in popularity in the last decade; now though there has been a vast increase in the use of social media to buy and sell clothes. There are many applications that I have come across that are relevant to the service I wish to provide although there weren’t any good services that offered trading capabilities. The services that I have looked into and have found to have be of use to my project and analysed below were:

* Facebook Groups
* Facebook Marketplace
* Depop
* Mallzee

**Similar Applications**

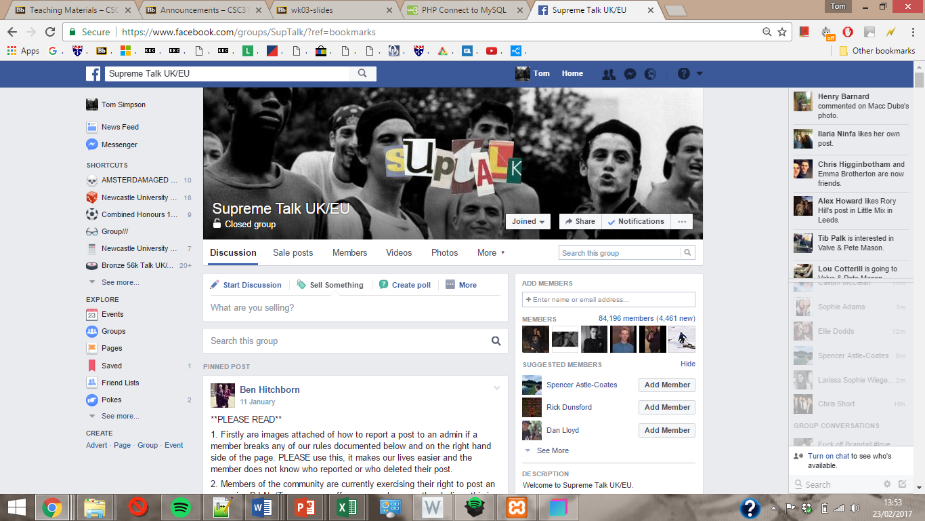
**Facebook Groups**

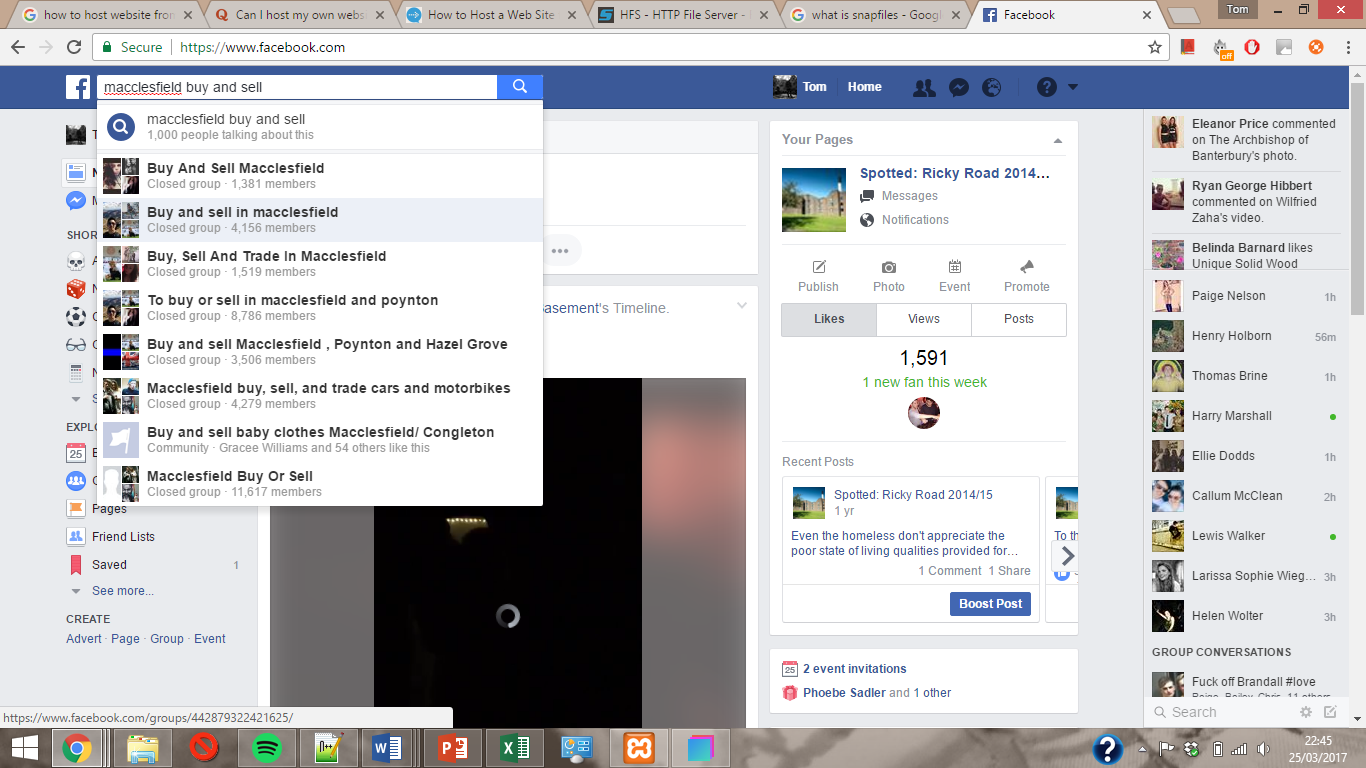
For the last few years there have been many Facebook groups that are focused around the idea of buying/selling or recycling old products or clothing. You may have come across either:

* Local buy and sell groups (pickup and collect in the same area)
* Larger spread groups specific to certain fashion styles or brands

[R] It is believed that approximately 450 MILLION people used Facebook groups online for buying and selling. This is a huge number of users and all of these people would also be interested in trading.

Brand based FB groups

There is a huge number of Facebook groups that are based around a sole brand. These groups tend to be either country based or worldwide and are not based on where the user lives. A lot of these groups exist to help members of the group **trade,** buy and sell rare and exclusive items that are sold out and no longer available in retail stores. Below are examples of these types of groups (some of which have nearly 100,000 members. ‘Supreme Talk UK/EU’ has 84,196 members (23/02/17) many of which are active in the group swapping items on a regular basis. People who follow the brands regular releases often like to buy items for a short amount of time and then sell them on not so long later to maintain a different wardrobe and to help fund their next purchase.

Area based FB groups

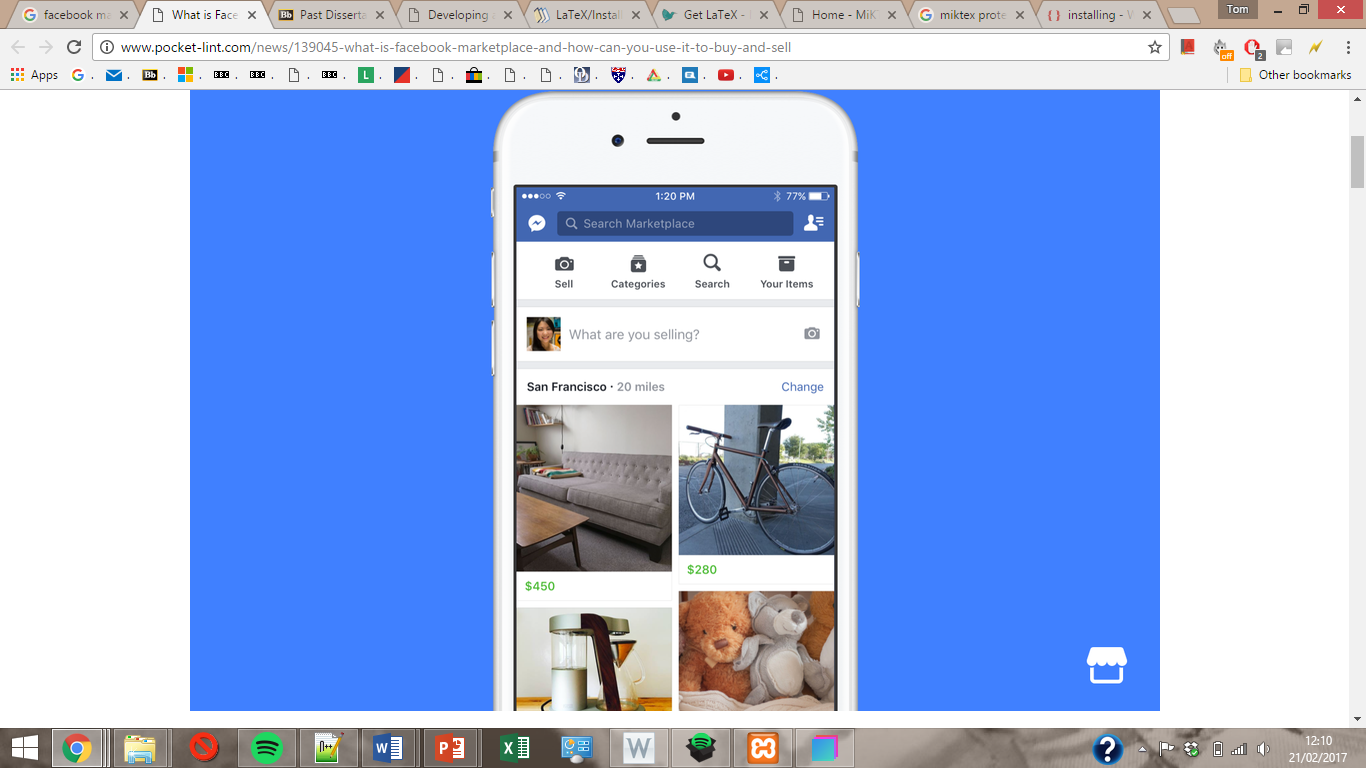
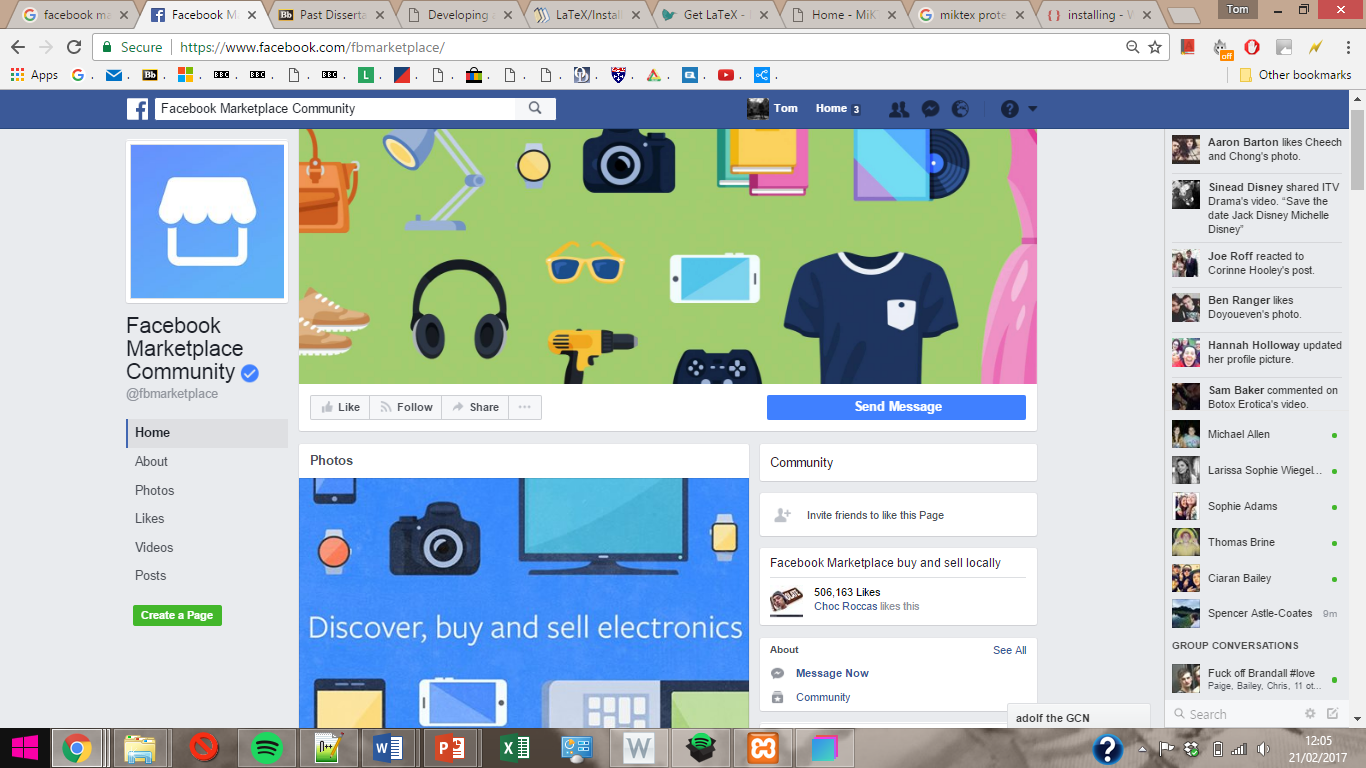
Area based groups are usually collection sales amongst people who live in the same town. These are a little bit less relevant to what I want to achieve as the sales in these groups are not always clothes. The image to the left shows these groups attract quite a lot of attention, especially considering Macclesfield has a population of 52,000 and 60% of people actually have Facebook in the UK[xxx].

***Facebook Marketplace***

Facebook then introduced ‘Facebook Marketplace’ early October 2016 rivalling other large selling markets like eBay due to the popularity of these Facebook groups. I tested the marketplace myself and found that it links with the Facebook groups and sends you notifications to your marketplace when new items are uploaded to the groups.

Facebook have considered the different types of Facebook groups when creating their service as the main marketplace feed displays items in your local area; but then also allows you to limit the type of items that appear based on categories. The range of categories available on the marketplace includes Home, Entertainment, Clothing, Family, Electronics, Hobbies, Vehicles and Classifieds. There are then subcategories for each of these that give more specific results.

It is easy enough to upload an item as users can either take photos of their item through the app or upload photos straight from the camera roll. All that is then required is to price your item and add a brief title and description. Before you finish the upload I noticed that Facebook offers you the chance to post the item into any of your groups too (some of the clothing items may sell faster in the branded groups as its then posted to a specific audience.



*Relevance of Facebook Marketplace*

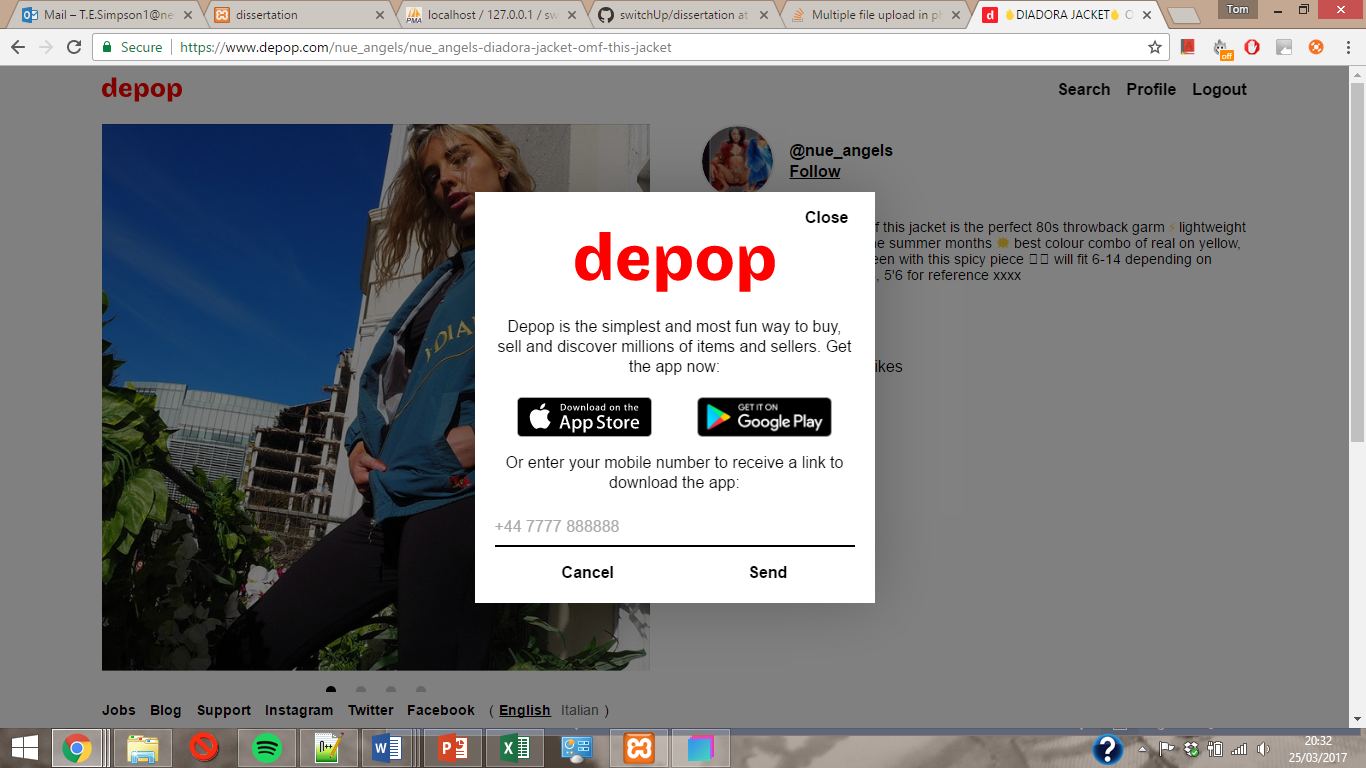
Most of the transactions through Facebook marketplace are generally cash transactions and through collection. This is mainly due to Facebook choosing to display the newsfeed based on the user’s location; encouraging local transactions with other people from the same area. I think this is one of the reasons why Facebook takes a 0% cut of all transactions. This also means all users are responsible for trusting each other and that Facebook are not liable for user’s losing items or being scammed.

**Depop**

*Background*

Depop is a successful mobile application that only launched in 2011 and was created for it’s users to be able to buy and sell goods to each other. Depop is mainly used for the sale of second hand clothes of all different values and is different to other online marketplaces as there is not a bidding system, instead users just outright buy the item from the users profile and are able to message the seller to negotiate the price beforehand. Sellers are expected to then post the item to the buyer within the next few days and once the item has been received both users leave a review for each other. The reviews should be about:

* Communication (response time/politeness)
* Speed of postage
* Item as described
* Would recommend/ deal with again

Depop is free to download, but makes money on the item sales as they take a fee of 10% on each sale.

Depop also has a website but it doesn’t allow users to buy, message users or upload items from the site and tells them to download the app as shown on the message above. The website allows users to view and search items that have been uploaded; but other than that serves no major purpose.

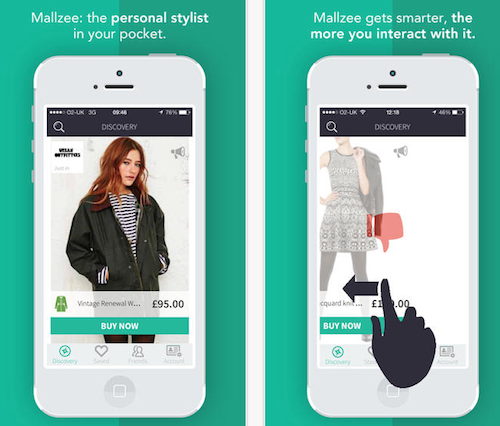
*What is useful to me?*

Depop is what gave me the inspiration for this service as I’m a regular user and often get message requests for trades through the app. I strongly believe many of the users of Depop would also be interested in using a trading service as it would save them money as they are constantly buying new clothes (aswell as get rid of old ones).

**Mallzee**

Mallzee is also a clothes selling app, but rather than using a newsfeed to present items to potential buyers, Mallzee offers users a few different ways to find items:

* Search page based on item type, brand, price and colour
* Through a carousel

The searching method is a common feature on a lot of apps and just for online marketing in general. Mallzee has chosen to include filtering by colour on the search which isn’t always an option.

The carousel feature is much less common for buying and selling apps, but is becoming increasingly popular for phone applications as a UI. More and more mobile apps have begun using a carousel since Tinder really kicked off and a lot of credit for that goes to the simplicity of the like or dislike system. The way the carousel works is the user swipes left for a dislike or right for a like on items and then the next item appears. If the user clicks like, this item is saved to the users ‘saved items’ page and the user can then think about whether they would like to buy the item.

*Relevance*

This carousel is an option I could consider if pursuing a mobile application. The way items are saved to a part of the users profile is clever and useful for the user and for Mallzee. It encourages purchases through the app, just because users can easily view a selection of items they like and could be tempted in to buying. It also means Mallzee can select items shown to the user on the carousel based on their like and dislike history.

**Technical Specification**

|  |  |
| --- | --- |
| Version Number | Requirement |
| 0.1.0 | Users can register |
| 0.1.1 | Form validations |
| 0.2.0 | Usera can login and logout |
| 0.2.1 | Login Validations |
| 1.0.0 | Pages from hierarchy added |
| 1.0.1 | pages all link to each other |
| 1.1.0 | Item listing accepted to database (upload) |
| 2.0.0 | Newsfeed able to loads items from database |
| 2.1.0 | Profile page loads users items from database |
| 3.0.0 | Item details page loaded on click of an item |
| 3.0.1 | Loads the specific item clicked on |
| 3.0.2 | Loads list of logged in user's available items |
| 3.1.0 | Users can make trade requests from this page |
| 4.0.0 | Trade requests can be accepted or declined |
| 4.0.1 | Status get updated upon decision & panel is reloaded |
| 4.1.0 | Accepted requests can be marked as completed |
| 4.1.1 | Status gets updated again & panel reloaded |
| 4.2.0 | **Users can trade** |
| 5.0.0 | Users can write reviews for completed trades |
| 5.1.0 | Reviews left for a user can be viewed on their profile |
| 6.0.0 | Help page |
| 7.0.0 | Users are able to send and receive messages to each other |
| 7.1.0 | Users can view their new and old messages |
| 8.0.0 | Login links with Facebook account (email) |
| 8.1.0 | Can find friends already using the app through Facebook |
| 8.2.0 | Link with other social media |
| 9.0.0 | Can search for an item by name |
| 9.1.0 | Can filter search by size, type condition or brand |
| 9.2.0 | Users can edit their uploaded items from profile page |
| 10.0.0 | Settings page from the profile where users can set preferences |
| 10.1.0 | Users can b lock each other from messaging |
| 11.0.0 | Users can like items |
| 11.0.1 | Users can comment on items |
| 11.1.0 | Users can follow each other |

**Development Tools**

**Design**

*Balsamiq MockUps*

Balsamiq is a software tool handy for creating screen designs as it provides phone/screen templates as well as shapes, button and text tools that can be added to create your layout. Balsamiq doesn’t offer a wide range of fonts or variation on the shapes/features provided in the app.

**Creating the HTML pages**

*Notepad++*

In the past Notepad++ has been the software I have used for the majority of website design work I’ve done just because of its simplicity. However, this will require a lot more work than my past projects and I think that it’s worth considering other software for this task as I could save myself a lot of time.

*Bootstrap Studio*

I found Bootstrap Studio had lots of positive feedback online, mainly because of its responsiveness and container/column/row system being extremely useful for organising layout. Bootstrap Studio is only around $25 for the license too so isn’t actually very costly for what you get.

*Conclusion*

I decided the most efficient software for me to use would be Bootstrap for creating any HTML features. As my website will use PHP, I will also need to use Notepad++ to create PHP files.

**Testing during Implementation**

*XAMPP*

While I’m creating the site, I plan on using XAMPP to test the PHP files on localhost. This should be simple enough to do as I only need to download XAMPP and have the latest version of PHP running on my laptop to get this working. I can also setup my database using phpmyadmin once this connection has been established.

**Other Software**

*Microsoft Excel*

Any spreadsheets will be made using Microsoft Excel. This will be mostly documentation tasks and will be used for creating things like:

* Timeline/Work plan for the project
* Version control document
* Testing

*Microsoft Word*

Any documentation or write up that I decide is necessary will all be done in Microsoft Word.

**Mobile Application or Website**

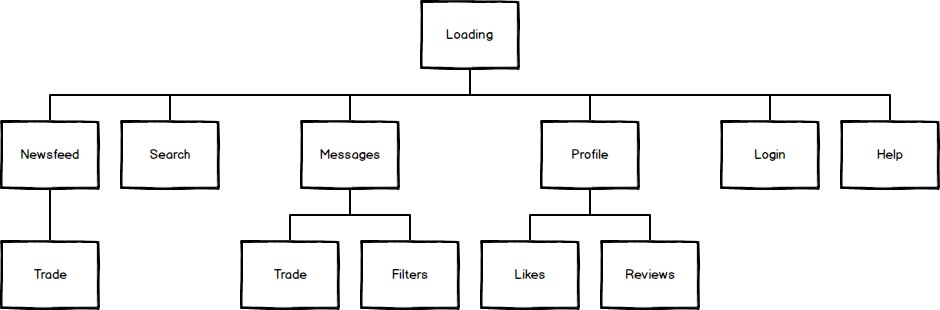
Originally when I wanted to pursue this idea, I was thinking of creating a mobile application as some of the similar applications (Depop, Mallzee) had taken this approach. I thought I could use Cordova to make an application using HTML, CSS and JavaScript; however after more research I think that a website would attract a wider audience as I was previously unaware of the popularity of the local buy and sell groups. People using the local buy and sell groups are from an older generation and are less likely to use the mobile application. Despite this, I still feel a mobile application would be a good progression for in the future after the website has been fully developed and has gained users.

Another factor that changed my mind is that I am using Bootstrap studio to create the site and this offers responsive services, therefore the website should function nicely on mobile devices.

|  |  |  |
| --- | --- | --- |
| Key  Source | Summary | Relevance |
| Vintage clothes reusage papaer | http://www.tandfonline.co m/doi/pdf/10.2752/175693 812X13403765252424 | Talks about the popularity of vintage clothes and the economic trends in buying/selling/reusing old clothes. |
| Cordova website  [5] | [https://cordova.apache.org](https://cordova.apache.org/docs/en/latest/guide/cli/index.html) /docs/en/latest/guide/cli/i[ndex.html](https://cordova.apache.org/docs/en/latest/guide/cli/index.html) | I used the guide on the Cordova website to install it and get an Android emulator working. |
| Guardian  article [4S] | [https://www.theguardian.c om/fashion/fashionblog/2012/mar/28/charityshops-thrive-duringrecession](https://www.theguardian.com/fashion/fashion-blog/2012/mar/28/charity-shops-thrive-during-recession) | Article talks about the increase in popularity of second hand clothes shops. With the increase of online shopping considered also, this pushes the idea that mobile buy and sell applications only increase in downloads and popularity. |

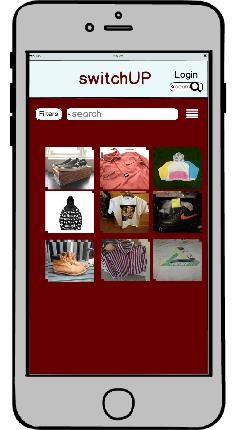
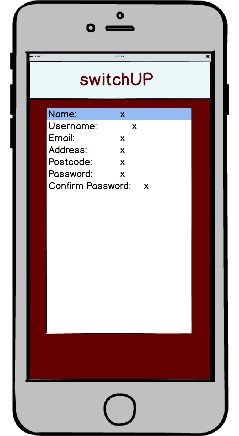
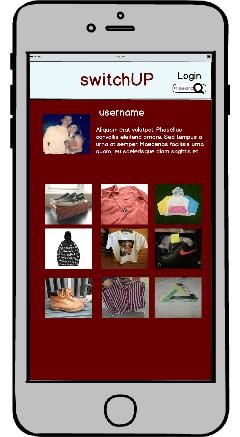
**Design**

**Mobile Layout**

During the design stage, I was leaning towards the idea of making a mobile application over a website. Therefore I started by creating a page mapping for how a mobile application to show the navigation and connections between pages (below). For the page mapping I tried to consider all possible features the service could eventually provide, this way I can design the system in a way that allows these features to be added easily rather than changing the whole system layout.

I decided to make some very basic mock up design screens (pictured below) for how a mobile application could look and what would be on each of the pages. I used Balsamiq to create these due to its simplicity and because I’ve previously used it for this sort of task. One issue however with Balsamiq is that it limits the user to a small selection of fonts and little detail variation on the existing shape tools. I still would use Balsamiq for this again though as my aim was to help visualise the layout of the app, not the styling and details.

**`**



**Website Layout**

**Navigation System**

The navigation bar should give access to all the pages mentioned below except for ‘item details’ as this can only be accessed by clicking on an item directly. ‘Home’, ‘Messages’, ‘About Us’, ‘Profile’, ‘Help’ and ‘Login(out)’ will all be navigation links and I will also put an upload button on there for users to add items. As with most navigation bars, it will be displayed at the top of the page and also contain a company logo ideally.

**Newsfeed**

When deciding the most practical way to display the newsfeed there were a few things I had to consider how the items would be laid out and how they would be loaded to the newsfeed. There were three different newsfeed systems that I was interested in from the background research and they are:

* Carousel
* Following system
* By Location
* Filters

As well as the newsfeed system to decide, I also need to consider the display method. Unless opting for the carousel which is its own display method, there are two display methods that I am considering pursuing:

* Gallery view
* List view

Newsfeed system

*Carousel*

The carousel could work really well on a mobile application, but I have been unable to find any websites that implement this feature. If I was to go with this option, then I think I’d have to modify the current system to enable items that are liked to be saved and make it possible for users to reflect on these. This could then perhaps be the page users can trade from as all the items that have caught their attention are stored there.

The only downside to carousel I have found is that it still needs a newsfeed or really good search system to complement it as I will not be capable of creating a carousel that can display items to users based on their previous likes and personal preference to the level that would be required. The carousel is more suited to swiping than clicking when it comes to user interaction, but it still provides the same service in the end.

*Following system*

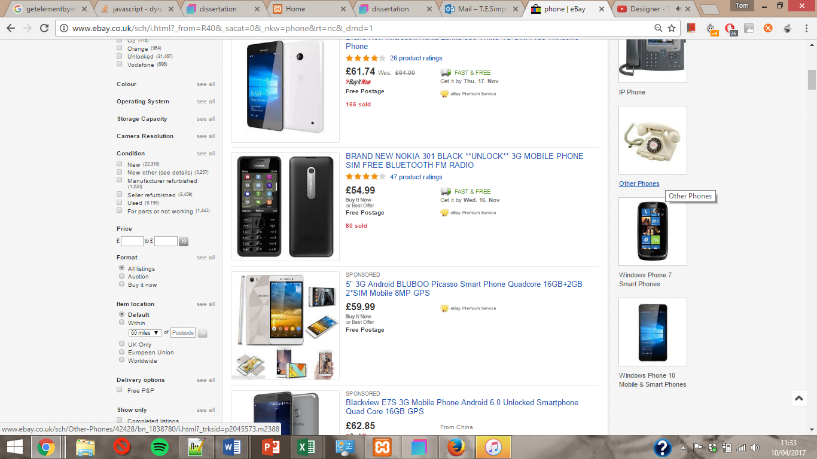
This is Depop’s way of implementing the newsfeed. The way this works is users follow other users that have uploaded items that they like and then new items uploaded by these same users then appear in their newsfeed. This is actually quite clever as it means the newsfeed is filtered based on size and price already, assuming a user uploads similar types of item each time.

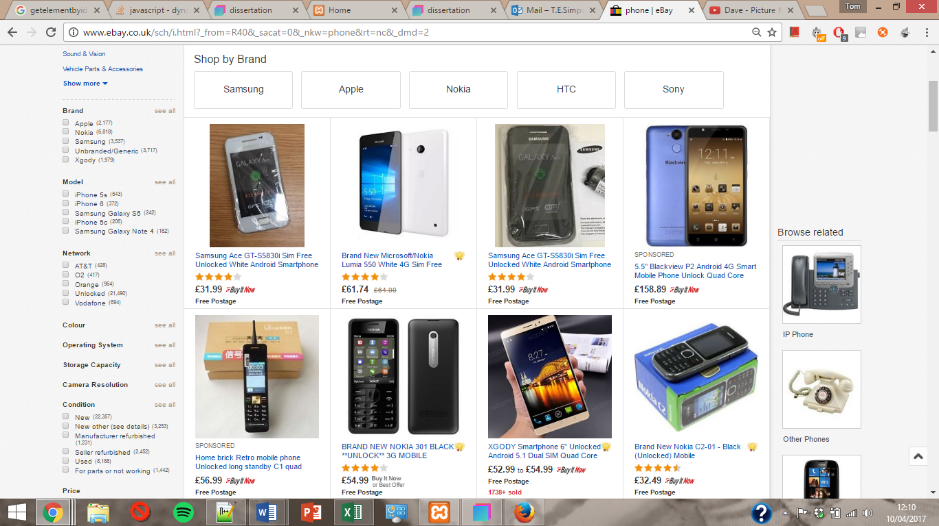
*Location*

After reading up on how Facebook Marketplace works I considered the option of a location based newsfeed, where users would set their location and a radius and then items uploaded from within this range would appear in the newsfeed. Users would then meet up and swap their items or could even still post if they wished to.

Display Method

*List View*

List view does the job very well and makes it easy to present the items, but my main issue with this method is that a list view will not maximise the number of items visible on the screen at the same time, without reducing the size of the images of the items. The images could be smaller; but then a high majority of the content on the page is text, when really it’s what the items the users want to see. An excellent example of this is shown on the left and this one of the options eBay gives you for viewing items.

*Gallery View*

Gallery view is the alternative option provided by eBay. I think it focuses more on the images than list view and that’s what catches the user’s attention. Depop’s profile page takes a similar approach as it has three columns of items and solely displays the image. When the user clicks on the image, this loads all the item images and the rest of the information about the item.

*Newsfeed Conclusion*

I will be opting to go with gallery view for the newsfeed. I would have liked to have gone with the carousel idea, but getting it to load items on preference would become a project within itself. List view would also do the job very well and is even eBay’s go-to display. That said my reason for thinking switchUP would benefit more from a gallery view newsfeed is that the items on there will all be clothing and don’t rely on a description as much as the other items you find on eBay (if the image and item name are good enough).

The following system also works well and the only reason I didn’t opt for this method is that it would be difficult to get the app going with this system. When there is only a small number of users and limited items on the site, there won’t be many people to follow and every user’s newsfeed will be rather empty. I will consider this option for further development because if the website was to gain a large enough user base then I believe this would be the best way to deliver the newsfeed.

**Profile**

I opting for grid layout on for the newsfeed and I intend on keeping this the same for the profile page as the layout should be the same; the only difference being the profile page will only display the items belonging to that specific user. The profile page will be responsible for displaying:

* The user’s items
* User details
* User reviews

I’d like to create a side panel for the user’s details to be displayed in so they are always on view when viewing the user’s profile. I’m unsure of whether to put the reviews in here as well or to just store a link to the reviews here and actually keep them on a separate page. Deciding this at a later date shouldn’t be an issue and I will have sorted it by the time implementing the profile page comes about.

*User Details*

Each user should have a little bit of information about them on their page, this can be whatever they want; ideally related to the style of clothes they will be uploading. Every user will also have the option of having a display picture; if not then a default one will be provided for them. I used <https://www.vecteezy.com/vector-art/> for the default pictures as their vectors are great and also copyright free, providing credit is given somewhere on the page (as I have done in the footer when a default picture is used).

*User Reviews*

The review section is an important part of a user’s profile. It should show who the review was left by, the rating they received, the date the review was left and a brief description of how the transaction went. User’s also need to be able to write reviews for each other but I plan on putting that with the trade section so reviews can be left after trades are completed.

**Item Details**

This page can only be reached when a user directly clicks on an item (either from the home or profile page). I am still sceptical about making this its own page as I could just create a pop up over the page with the same capabilities. Either way, the user description should then be loaded and there should also be a dropdown list of selection of some sort of all the logged in user’s items to compare against for a trade. The user should be able to select any of their items and propose a trade to the other user for their item. More details of this will be in the trades section.

**Messages**

When directed to the messages page, users will see their inbox as a big container with a panel down the left hand-side that displays the usernames of the people they have been in conversation with along with the time of the last message between them. These will be ordered from the top by the most recent conversation and upon click of one of these usernames the whole conversation will appear on the right hand-side of the container.

**Register and Login**

*Register*

Below is a table showing the fields I will require on my register form and any validations required:

|  |  |
| --- | --- |
| Field Name | Validation |
| Full name | Letters only |
| Username | Must be unique No special characters |
| Email | Input type email |
| Dob | Date format |
| Password | Same as repeat password |
| Profile picture | Optional, but must be image file |

Upon submission of the form, the password should be encrypted using MD5 and the validations should be checked before inserting anything to the database. I think JSParsley is the easiest way to do the validations but I will do further research when the time arrives.

*Login*

Users will be able to login to the site using their unique username and password. I plan on making the landing page the login screen, but still giving users the option of browsing without logging in. If a user isn’t logged in then they will not be able to see the message page, their own profile page of course or submit any trades from the item details page. Whilst on the actual login page, the only visible links in the navigation bar should be ‘Browse without logging in’, ‘About Us’ and ‘Help’.

Once the login is successful, they are then redirected to the home page where they should see a welcome message followed by their username. The use remains logged in until the browser is closed or the logout button is clicked which destroys all the session data.

**About Us**

Brief description of the purpose of the site, quite similar to the aim but put across in a friendlier manner. There will also be links to social media accounts for switchUP..

**Help**

Help will be the last page I create before the system is completed and will practically serve as a user guide explaining the functionality on the site. On this page there will also be contact information for any issues user are having and also any improvements for the site. The contact information will just be my email address, but it should be in a form format and give them the option to send an email to me directly from the site and select out of a few option what they are contacting me about.

**Trades**

*Trading on the app*

My plan for the trading system is to have users make trade requests to each other. There are four possible stages a trade can be in:

* Trade Request: A trade request is made from one user to another, specifying the item they want and the item they are proposing to trade for. This initiates a trade and the other user involved will see this trade request next time they log in and can then either accept or decline the offer.
* Trade Accepted: If the trade request is accepted then the trade status is ‘accepted’. Once the trade has been accepted then the trade can occur (if meet up then arrange a time and place, if post then users should ship). The items should then be marked as sold in the database so they cannot be traded again and users can no longer receive requests for them.
* Trade Declined: If the trade request gets declined then that’s the end of the trade and nothing is to be done. Declined trades should still be shown to the user so they are aware it’s been declined (perhaps on a temporary basis).
* Trade Completed: Once both users have received their items, they should both mark the trade as completed (even if unhappy with the item). Once both users have done this, they will be able to leave a review of how easy the other user was to deal with.

*Trading Safely*

As mentioned earlier, there are three feasible ways for trades to occur:

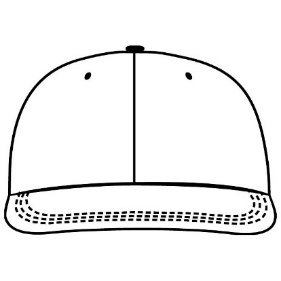
1. You meet up in person to exchange
2. You use a middle man (trusted third party who both participants send their items to)
3. One user ships first (just risk and hope the other can be trusted to send their item)

The first two options I would consider risk free solutions, if done correctly. With a meet up exchange, users should meet in a public place and check the item for any flaws and legitimacy before exchanging. I would consider this a plausible method; that said it only works when the users live near each other to make it worthwhile. Using a middle man works is a great solution too, if the third party can certainly be trusted. The only downside to this method is the additional cost that may be required as the item will be shipped twice and there is often a small fee for middle man services. I am considering allowing users to opt for the third method, at their own risk but it is not something I would recommend unless they know the other user or have dealt with them before through other services.

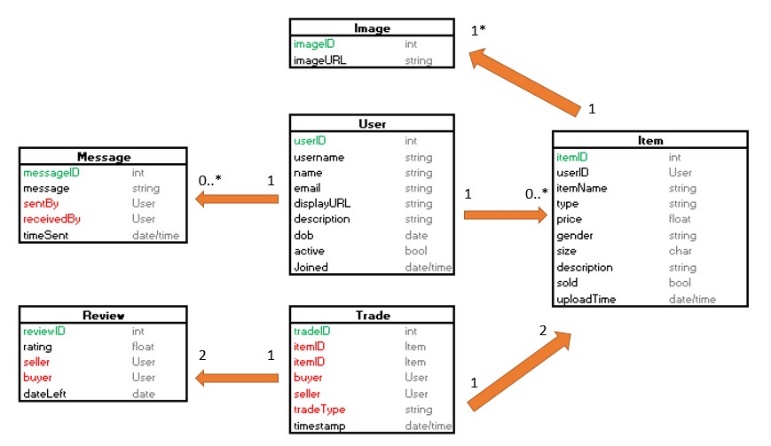
**Logo**

I created the logo for my site using Adobe Photoshop and focused on ensuring that:

* The colour scheme of the logo goes with the site
* The logo shows some relevance to what the service provides

Initially I experimented with up arrows (play on the name switchUP), but I thought the first design looked more like a real estate logo as the arrow was stretched to cover the UP bit of text and actually looked more like a house. I then came up with a better idea and altered it so the letter T in switchUp was replaced with a thinner UP arrow and also replaced the stretched arrow with a vector t shirt template and coloured it in appropriately as the site is for trading clothing. I was quite content with the logo as a whole, but felt the t-shirt vector image didn’t quite look right a free vector site [R] to get the outline of a t-shirt, but didn’t like the way it turned out and swapped it out for a cap. 

**Database Design**

Before creating the database in phpmyadmin, I made an ERM that shows all the tables I would need to store for the website to work.

I then made a class diagram to show what attributes each entity will have, what type they are and their relationship with the other entities. I’m glad I did this though as I originally didn’t plan on having a trade table, but then I needed a better way of linking the reviews with the user and it would certainly be beneficial; if not a necessity to keep a record of the trades.

Another decision I had to make was how to connect the images to the table. I was undecided as to whether to have an image table that had the ‘imageID’, ‘imageURL’ and ‘imageType’ (either userImage for the display picture or itemImage for an item) and then one more column that has either the itemID or userID for the image. After consideration I decided the best option would be to just store the display picture URL in the user table as it is always 1 to 1. Then I can just store the itemID in the image table with an imageID and imageURL.

***Item***

Swapped column there so that when a trade is accepted, the item will no longer appear in the newsfeed, otherwise users can still request trades for it and it shouldn’t be possible to accept two trade requests for the same item from different users

FORGOT:

1. Gender
2. Description
3. tradeID from review

**Implementation**

After switching from a mobile application to website, I used the screen designs to create all the webpages in Bootstrap Studio. These screens just showed each page would be laid out and lacked the styling/branding that is to be added further on.

The first step of implementation was to create HTML webpage screens based on the designs. I’d opted to use Bootstrap Studio to create mine and bought the software through their website.

Once I’d created the page screens, I started to use my version control document to begin implementation. I made three version control documents; all the same but with three levels of detail. The simplest version is shown in the table below and just shows the version number and the name of the task at hand.

|  |  |
| --- | --- |
| Version No. | Version Summary |
| 1 | Login |
| 2 | Navigation & Uploading |
| 3 | Load items to page |
| 4 | Setup a trade request |
| 5 | Trading Process |
| 6 | Reviews |
| 7 | Help |
| 8 | Messaging |
| 9 | Social Media |
| 10 | Searching |
| 11 | Settings |
| 12 | Liking and Commenting |

For my objectives to be met for this project I need to complete all the versions up to 5.

**1.0**

Version 1.0 requires the user to register to the site and log into their account. I encountered some slight difficulty when creating the login functionality as I was following a YouTube tutorial (https://www.youtube.com/watch?v=4oSCuEtxRK8&index=1&list=PL7C9AFC3942AC8E40) that explains how to create a user login in PHP as well as online resources, but this tutorial used MySQL rather than MySQLi and some of the code I was running is now deprecated. However, this was a good basis to start with as it got me using SQL syntax to retrieve objects/associative arrays from the database.

**2.0**

*Uploading Form*

The first step I took was to get the upload.php file to upload the data of the correct types to the item table and then got it to post from the form. From there, I changed the insert statement so it took PHP variables that stored the correct $\_POST values from the upload form.

For the item ID I created a method called getHighestItemID() which reads the highest current ID in the table by ordering the results in descending order and only retrieving the top row. The itemID column is then selected and the function returns that value + 1 and this is stored as the new item ID.

, had to get correct sizes to show depending on item gender and type. If unisex, show both, if accessory possibly no size

* I completely forgot to consider female clothing as an option during the design phase, added it after
* created php file for every page and added them to nav (inc messages)
  + New column phpmyadmin

**3.0 – Loading items to the page**

***Loading to Newsfeed***

Loading the items to display on the newsfeed wasn’t too difficult. I used the same approach as with the login, but had to adjust it to fetch an object rather than fetch assoc. My SQL query retrieved all values from item where the ‘swapped’ column was 0 and I then stored the results of the query into arrays for each column for future use. My Bootstrap studio html file contained the outline for how items should be displayed, so I made a ‘home.php’ file copied the HTML code over and embedded PHP into the HTML. This meant I could print out each item dynamically and then used the echo command to print the values that I had stored about the item (price, condition, description, the item’s user).

***Loading to Profile***

* Moved upload button from nav to user details (only shown when user visits their own page)
* Default display picture

**4.0 - Setting up a trade request**

I began constructing the trade request setup based on how I described the trading system in the designs.

**5.0 – Trading Process**

Rather annoyingly, after completing the implementation of the trading system, I spotted a crucial flaw. One of the important aspects of the project was to offer a few safe methods for users to exchange items. The review system works so that if a user has bad reviews then they aren’t exactly trustworthy and shouldn’t be dealt with. The way the system works in the design section means that once the trades have been accepted, the users are supposed to ship the item (if that’s the method chosen) after the items are received the users mark the trade as completed and leave a review based on how happy they were with the service. However, if a user was to never send the item and never mark the trade as completed then the user they were trading with would never actually be able to mark the trade as complete.

Therefore, I had to modify the way trading works so that after the trade has been accepted reviews can be left; then once both users have left reviews the trades are automatically set as completed.

Trade Status: -1 = Declined, 0 = Pending, 1 = Accepted, 2 = Completed

To make easier when fetching from db, user2 is always the one who made the trade request

**6.0**

Set form box maxlength to 50

Used textarea and set form attribute

Due to changes in pt4.0, had to add reviewID1 and reviewID2 to trade table, if both not 0; trade complete

Passing using hidden forms (username specifically) – had to set value without using $i, fixed some problems but still wouldn’t let me pass username with the form because its on the same page

<http://stackoverflow.com/questions/20738329/how-to-call-a-php-function-on-the-click-of-a-button>

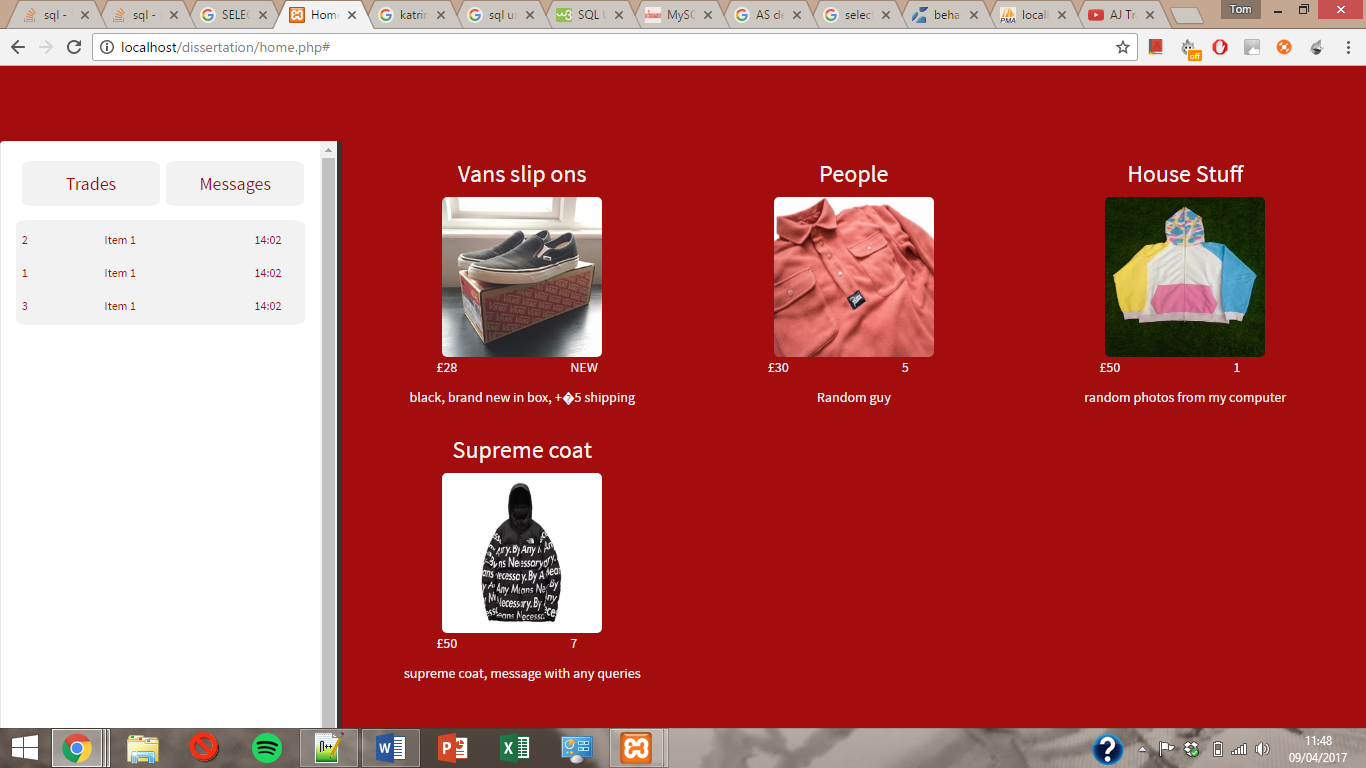
didn’t opt for ajax and didn’t want to use get due to it being easy to change url and leave a review for random user

used javascript dom element to pass

**8.0**

I decided to remove the messages page and instead add it into the panel on the home page where the user can access their trades. I did this because there was little need to associate a whole page solely for messaging and it also means that If a user wishes to message another about a current trade, they don’t need to change the page; just the tab on the panel.

**Features**

* Pop up message box
  + On message click, show box over the top (z-index)
    - Had to set position to relative/absolute
  + Did clever check on whether user is logged in, if not then give message button different class so pop up asks them to login first
* Cause a lot of issues loading the messaged users as no DISTINCT ACROSS MULTIPLE COLUMNS
  + Tried group by and opted for union of the two columns instead and use ‘value’ to separate
  + Finally printed out (screenshot) despite having error because I had an unintentional ‘$row = mysqli\_fetch\_assoc($result)’ outside my while loop meaning I was missing the first result everytime. 
  + Had to add if statement to prevent the users own userID from being added as they shouldn’t be in their own inbox
* Once I got the user ids of everyone the user messaged had issues retrieving the date sent and message based on that (for the users inbox
  + For every userid prints out the same message and date
  + Spent ages, but traced back through to find issue with sql statement but no error from it as it was ‘WHERE (senderID OR receiverID = ‘$MuserID’) when it should have been ‘WHERE (senderID = ‘$MuserID’ OR receiverID = ‘$MuserID’)

**Hosting**

[R] <https://www.apachefriends.org/docs/hosting-xampp-on-azure.html>

For now, I have chosen to host my site using Microsoft Azure to host my site as I found they offer quite good pricing for their services and a free trial for up to £150

* Created a bitnami and Azure account (Azure free trial)
* Linked them together on the Launchpad
* Downloaded certificate for subscription to for them to link together
* Launched a LAMP stack on a Linux VM

**Setting up the server**

When configuring my Azure Cloud Server, I set the region to UK West as this was the best UK option and the server size to the smallest/slowest setting for now. The ‘extra small’ setting means the server will share virtual cores and I can increase the size up to the point where the server has sole access to 8 virtual cores. Whilst I am still developing the site, there is minimal traffic as the only hits are when I check on it; once it’s fully complete I intend on increasing the size so it can handle higher loads.

**Software**

*Filezilla*

I opted to use Filezilla to move my local file directories onto the server because of its simplicity. To setup a connection to the server I am able to use the site manager and just have to enter the hostname and the SSH credentials to connect. From here I can move files between my laptop and server in the click of a button which is handy for updates.

*Putty*

I used Putty as my SSH client to check that my Apache and MySQL servers were running correctly and also that I was running the latest version of PHP.

**Evaluation**

**Future additions to improve the app**

* Offer two for 1 trades for items worth more
* Mobile app
* Multiple image upload

**References**

1. Author: Tracy Diane Cassidy, Hannah Rose Bennett

Used for: Vintage clothing trading

Date Accessed: 15/11/16

Reference: <http://www.tandfonline.com/doi/pdf/10.2752/175693812X13403765252424>

1. Reference: Depop
2. Reference: Mallzee
3. Author: Guardian

Used for: Research into vintage clothing still gaining popularity

Date Accessed: 15/11/16

Reference: [https://www.theguardian.com/fashion/fashion-blog/2012/mar/28/charityshops-thrive-during-recession](https://www.theguardian.com/fashion/fashion-blog/2012/mar/28/charity-shops-thrive-during-recession)

1. Author: Jinhee Han

Used for: Background research

Date Accessed: 17/11/16

Reference: <http://lib.dr.iastate.edu/cgi/viewcontent.cgi?article=4643&context=etd>

[6] Author:

Used for: Facebook Marketplace

Date Accessed: 02/12/16

Reference: http://www.pocket-lint.com/news/139045-what-is-facebook-marketplace-and-how-can-you-use-it-to-buy-and-sell

**Images**

Image Name: Mallzee Carousel  
Reference: <http://fashionretailfuture.com/wp-content/uploads/2015/07/mallzee.png>

Image Name: Cap vector for logo  
Reference: <http://www.vectorportal.com/free-vector/free-cap-vector-cdr-11/>

Image Name: Default profile picture vector  
Reference: <https://www.vecteezy.com/vector-art/73545-avatar-icons>