Interim Report: An implementation of an interactive annotation game on Facebook
Author: Louise Knight
Supervisor: Irena Spasic
Moderator: Paul Rosin
CM0343 (40 credits)

Table of Contents

Introduction

Having tags as metadata for images is important because of their use with screen readers for the visually impaired and image search engines, such as Google Images (1), but the task of collecting tags is not an easy one. If we wanted people to tag the images, for example, they would need to be paid (or offered some other reward), and methods that use computer vision to collect tags still have a way to go before they can be applied generally. (See the Background section for more information.) Luis von Ahn took the idea of obtaining tags for images and packaged it as a game (the ESP Game, see (2) for more information, and also his current version on the GWAP website (3)). Other versions of the game have also been created for collecting metadata for other media, such as music (4) and video clips (5).

This project shall involve making an ESP-style game for tagging images, but will implement it as a Facebook application to take advantage of the fact that Facebook is much more heavily accessed in comparison to the GWAP website, which may not be known about except through word-of-mouth. This report includes a background study of the problem, a definition of the functional requirements and a basic design of the system (including database design, interface design, and justification of technologies chosen), as well as an outline of the general method that shall be taken when implementing the system.

Background

Computer vision and image search are just two of a variety of methods used for obtaining tags for images, but neither of these can be considered "good enough" to be used generally. For example, with computer vision, although programs have been developed to locate objects in images, such programs have not been developed for many different objects (2). (6), for example, describes a detector that can find faces and cars, but "combining algorithms for detecting specific objects into a single general-purpose classifier is a non-trivial task" (2, p. 324). Also, image search tends to use metadata such as text found "around" an image on a webpage or filenames to tag images, which may not always be reliable (2) or may be scarce (2). (For example in Figure 1, you can see that the name given for the image ("P1580322") was probably the default name given when the image was taken, and does not accurately describe the image. Also, there is little information about the image itself besides when it was taken and the camera model - "This photo was taken 9 hours ago using a Panasonic DMC-GF1".) (The reader is referred to (2) for a more detailed review of these methods). Some "collaborative tagging" systems have tried to improve the ease with which relevant information can be found (such as Delicious (7), which allows users to add tags to URLs; as many users post the same URL, tags from many different users accumulate for that URL). The main problem with these systems is that there would end up being some pieces of data that are tagged by many people (the more popular content), and a lot of content relatively unknown, and therefore not having many tags. For example, Figure 2 shows the number of times a particular URL was tagged for a large number of URLs (144,574) in June 2008 on Delicious. The data used to help create this graph was taken from (8), and is an XML file of each of the 144,574 URLs with their corresponding tags. A Java parser, the code for which is in Appendix A, was used to collect the number of times each URL was tagged from the XML file, and this was then output into a CSV file of the format:

URL	Times tagged
URL	Times tagged
...	...

The rows were them sorted by decreasing value of "Times tagged", and this data was used to create the graph in Figure 2.

Note the shape of the graph; this gives rise to the name of the phenomenon, the "long tail". Also, it should be mentioned that other websites such as YouTube (9) and Flickr (10) use tagging systems, but these only allow someone to tag content that they have uploaded. From these various issues the concept of the ESP game was created, as a means of collecting tags for images (2).
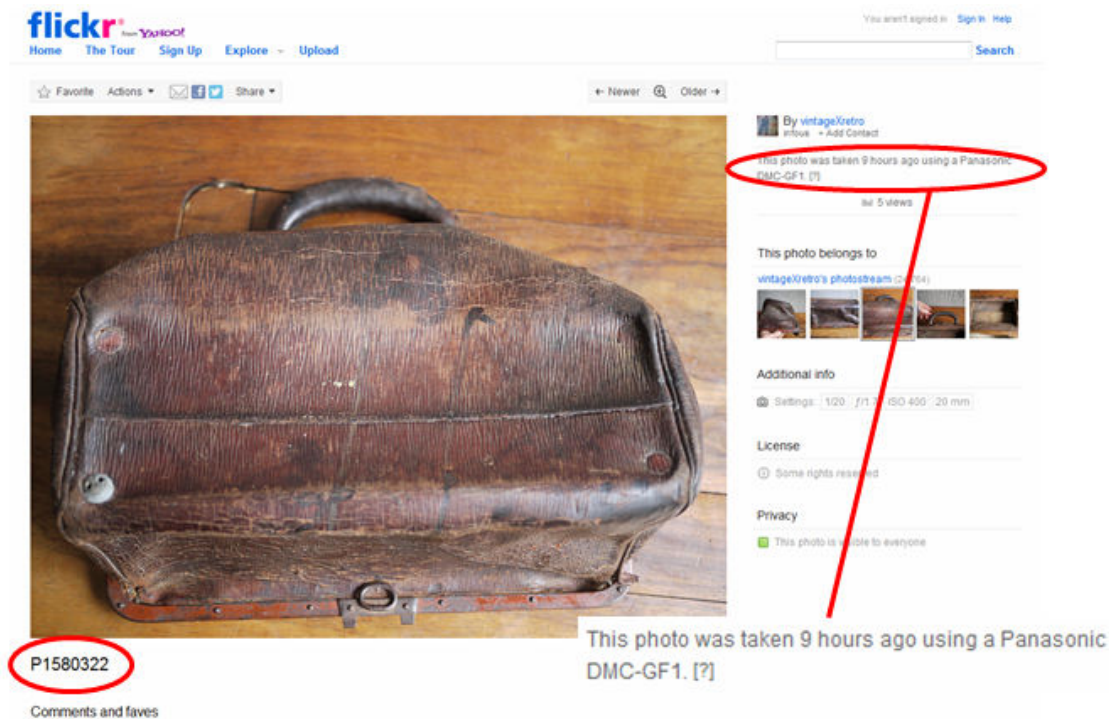
Figure 1: Screenshot from Flickr to demonstrate that metadata gathered about an image could be unreliable or scarce
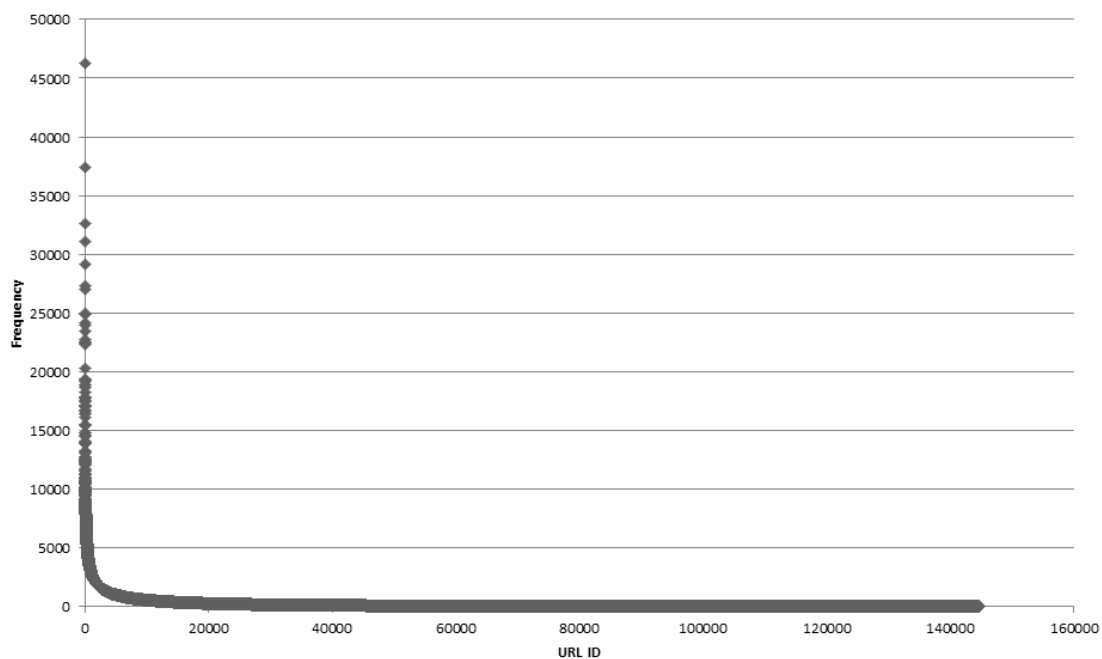

Figure 2: Graph showing how there are a lot of URLs that are tagged rarely and a few that have many more tags

Luis von Ahn's ESP Game, a version of which is hosted on the GWAP website (3) (see Figure 3 for a screenshot), is a well-known example of the concept of the ESP game. The instructions tell the player to "guess what words your partner is typing" (3), and each game is timed (3). Other aspects of play have been detailed in the Initial Plan for this project, so the reader is referred to that for more detailed information. The main insufficiency with this particular implementation is that the game is hosted on a website dedicated to GWAP games

3

(Games With A Purpose) (11), and one would presumably only be aware of this website through word-of-mouth. This problem would be resolved by our version by the fact that it shall be placed on Facebook as an application; this means it is more likely to be seen and also, users will be able to see whether and who of their friends play the game, motivating them to want to play it.
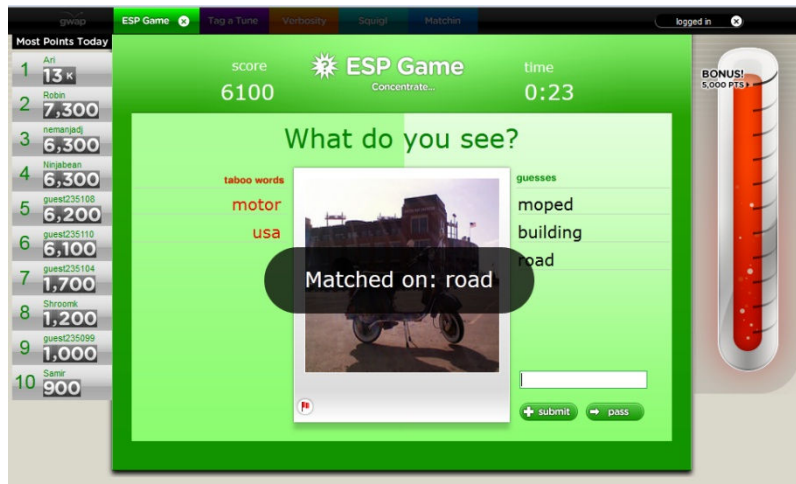

Figure 3: A screenshot of GWAP's ESP Game

We shall now describe some of the ideas addressed in GWAP's ESP Game, and which will need to be kept in mind during the implementation phase.

Taboo words: These are words which, when shown next to a particular image, are not allowed to be entered as a guess for that image (2). Taboo words are accumulated as the game is played (2). For example, if Figure 4 had the taboo words of "daffodil", "sky" and "yellow", players would not be allowed to use these tags for this image. This allows for a wider range of tags to be obtained for a particular image. For example, if you look at Figure 3, you can see that "motor" and "usa" are taboo words (listed on the left in red), forcing the user to enter terms other than those listed; we listed "moped", "building", and "road" (seen on the right), all different features of the image.
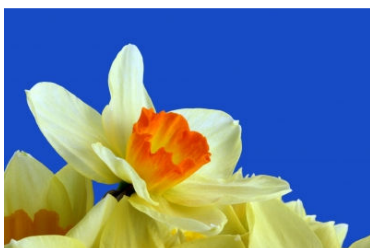

Figure 4: An example image (11)

Good tag ("label") threshold, X: This determines how many pairs of players must "agree on" a particular tag for an image before it is allowed to be used as a tag (and would then appear on the taboo words list for that image) (2). The smaller X is, the fewer the number of players that must agree on that tag for it to become a tag (2).

Pre-recorded game play: The ESP Game featured on GWAP's website is one to be played between pairs of users, so what happens if an odd number of players want to play the game? The game pairs one user with a bot which emulates a previous user's behaviour (2); this is

4

facilitated by recording a user's guesses for a particular image, to be used later. For example, see the behaviour for a player in Figure 5:

Player 1 (bot)

Player 2 (human)

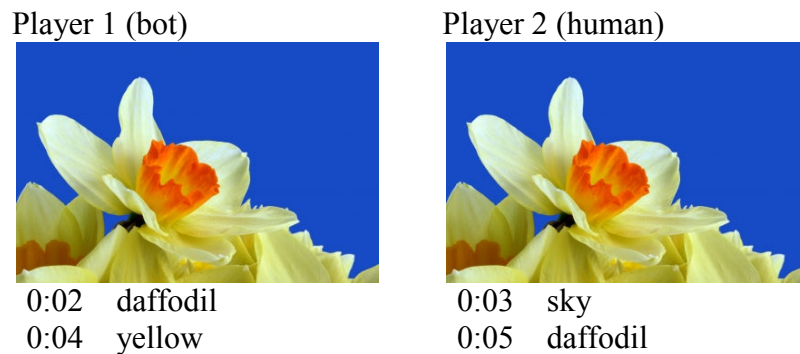| 0:02	daffodil		0:03	sky
0:04	yellow		0:05	daffodil

Figure 5: An example of how two players (one being a bot) enter tags for an image (11)

The game play on the left (Player 1) details how a previous human user played this round; at 2 seconds, they typed "daffodil", and at 4 seconds they entered "yellow". This series of moves is replayed while Player 2 is playing the game. At 5 seconds Player 2 (the real, "human" user) has matched with the bot on "daffodil", thus ending the round. Putting this into perspective, below we can see how this round of the game plays out.

0:02   Player 1 enters "daffodil"
0:03   Player 2 enters "sky"
0:04   Player 1 enters "yellow"
0:05   Player 2 enters "daffodil". This is a match with Player 1, and ends the round.

Aim: The aim of this project is to develop a multiplayer game that allows for the tagging of images in a way similar to the ESP Game developed by Luis von Ahn, but will be implemented as a Facebook application.

Research question(s): In order to demonstrate the achievement of creating such a game, this project will identify games and other applications currently in use for the collection of metadata, and will detail how these current approaches are lacking. The project will then involve creating a basic design of the system, including interface and database designs, and a description of functionalities to be developed. The application will then be developed by iteratively developing each feature and integrating it into the system, using a methodology described later in this report.

Requirements

User requirements are prefixed with "U", system requirements are prefixed with "S".

Core requirements

U1 The user shall be able to "play" the ESP Game online; in doing so:
> U1.1 They shall be able to play against another player, or against the computer if there are an odd number of users in the game/their partner exits (although theoretically, the user should not know whether they are playing against another player or the computer)
> S1 The system shall be able to pair two players that are in the game
> S2 The system shall be able to record the game play of players including the tags entered and the times those tags were entered; this shall be "replayed" if an odd number of players are playing the game
>
> U1.2 They shall be able to "tag" an image, not being able to tag with a taboo word
> S3 The system shall hold a number of images; the system shall be able to provide a random image from these on request
> S4 The system shall be able to collect tags from each player in a pair and compare them, looking for a match
> S5 The system shall ignore tags input to the game which are listed on the taboo words list for an image
> S6 When there is a match between a pair of tags for a particular image, the system shall store these in a database
>
> U1.3 The game shall be timed; that is, there shall be a time limit, for example two minutes, after which the game will end
>
> U1.4 The user shall be given points for "agreeing on" a tag for an image with their partner, and bonus points after tagging a specified number of images
> S7 The system shall keep a record of a player's score through the course of a game, giving points for matching pairs and also for matching on a certain number of images

U2 The user shall be able to recommend the game to a Facebook friend through a post to that friend's timeline
S8 The system shall provide a dialog box which allows the user to post to a friend's timeline

U3 The user shall be able to view which of their Facebook friends have played the game

Optional requirements

U4 The user shall be able to view a leader board for the highest scores in the game over the past day, week, and over all time (i.e. since the game was launched)
S9 Each of the leader boards shall be generated on-the-fly when necessary; how it shall be generated will be decided upon during the implementation of the system, when we know more detailed information about the system's infrastructure

U5 The user shall be able to post to their timeline about high scores achieved

S10 The system shall provide a dialog box which allows the user to post their high score should they get one

U6 The user shall be awarded a level based on their total accumulated points over all time playing the game
S11 The system shall store a user's accumulated points, and after every 500 point milestone (for example), the user shall go up a level

U7 The user shall be able to review the tags assigned to each image at the end of a game
S12 The system shall display the tags associated with each image after a game

Design

Choice of architecture

The repository style of architecture has been chosen for this system mostly because the system just seems to fit into this style naturally; the paths for each of the images shall be stored in a database, alongside any other image data required. The images themselves shall be stored in a regular file system. The image data shall be accessed by "clients", which will correspond to the users' browsers. This corresponds to the problem solved by the repository architecture detailed in (13), that is, of "maintaining a richly structured body of information" (13, section "Repository Architectural Style").

Design decomposition

Because the repository style of architecture has been chosen, it makes sense to decompose the system into modules based on function. The functions of the system are defined below:
- Authorise application
- Play a game
- Recommend to a friend
- View leader board
- Post about high score
- Review tags

We shall now describe the system from the highest level (i.e. the user's perspective), and then consider each of the functions in turn and describe them, breaking them down further where necessary.

Top-level (user) view of the system

Figure 6 shows a use case diagram which describes the various functions available to the user. In Appendix B are given the full use case descriptions in tabular format.
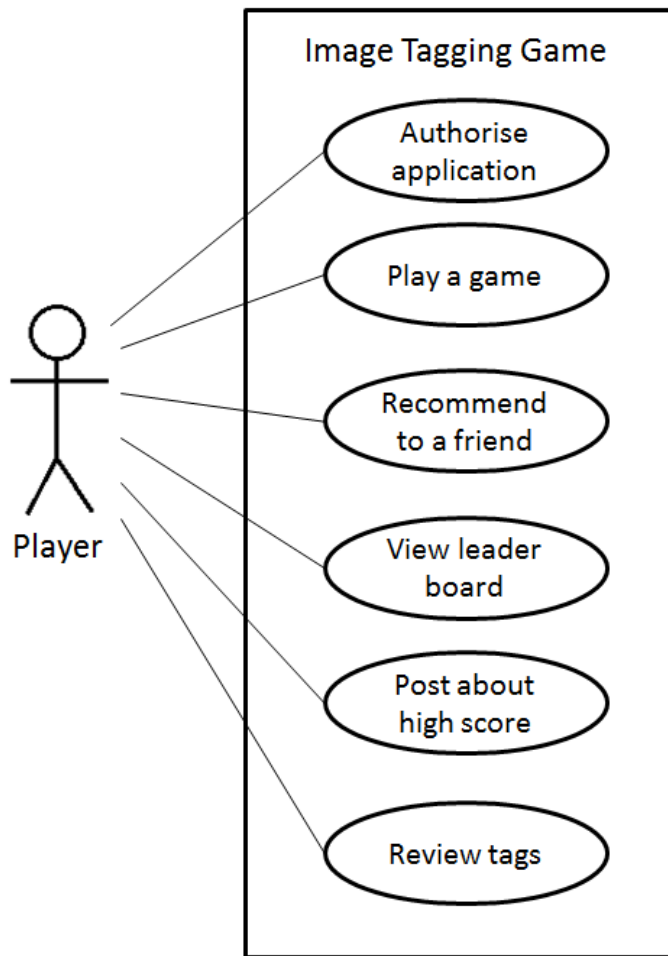
Figure 6: The use case diagram for the system

It could be argued that another actor that would be using the system would be an image search system, for example, and that it would use the tags collected using the image tagging game, but this has been omitted for simplicity, and because connecting the game to another system is outside the scope of this project.

Authorise application

For the player to be able to play the game, they shall have to agree that we can access some of their publicly-available Facebook data. For the purpose of this application, and taking into account the proposed functionality, at the most this should be their name and profile picture. This is a rather simple use case (in comparison to "Play a game" for example) that simply involves reading and accepting to the "small print" that the game has. (See Figure 7 for an example.)

Figure 7: Under the "Go to App" button, you can see that this application only needs the user's basic information, and that by default, the user's friends can see posts the application makes for them (but of course this can be changed).

Play a game

This is the main function of the game, in which the user plays the image tagging game. It is also the most complex function, and consists of pairing up two players, letting them tag images, and collecting the tags as we go along. Figure 8 shows an activity diagram of this function.
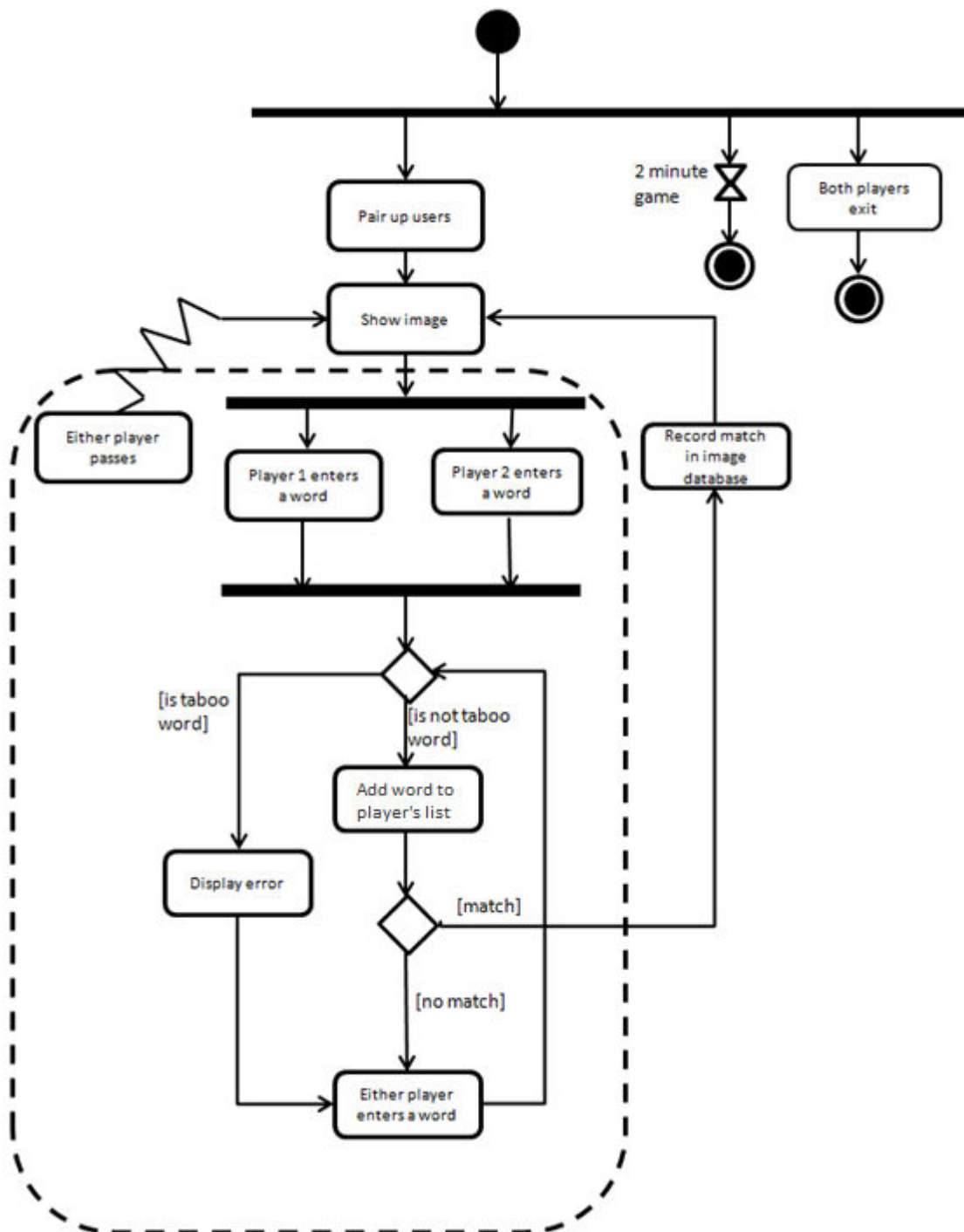
Figure 8: An activity diagram showing the "Play a game" functionality

This diagram shows that firstly two players ("users") are paired up to play, and are shown a (random) image from the database. They both enter a guess, and if neither is a taboo word, those guesses are compared. If there is a match, then that tag is recorded in the database, and if not, we wait until one of the players enters another guess, and then we check for a match again. Once there is a match, we record this in the database alongside the image, and then the players are shown another image from the database. Whenever either player enters a tag, we record that tag (if it is not a taboo word) in a list for that player ("Add word to player's list"), so that we can compare tags entered by one player against all of the tags already entered by the other. (Each player's list is emptied when a new image is shown.) If one of the players

decides that they no longer wish to enter tags for this image, they can click "Pass", forcing the other player to also pass on this image. Then, the game shows a different image. This continues until either: the 2 minute time limit runs out, or both users exit the game. If only one user exits the game before the time limit is over then the player that is left will essentially start "playing" with the computer as the other player, through playing with pre-entered guesses recorded during a previous game.

It should be noted that this diagram does not take into account recording the guesses of players or what happens if the player exits a game and there are no recorded tag entries for that image; these are finer details that shall be established once developing the system has commenced.

Recommend to a friend

This allows the player to post to a friend's wall recommending the game to their friend. An example of doing something similar (sending requests to friends using the request dialog) is shown in Figure 9.
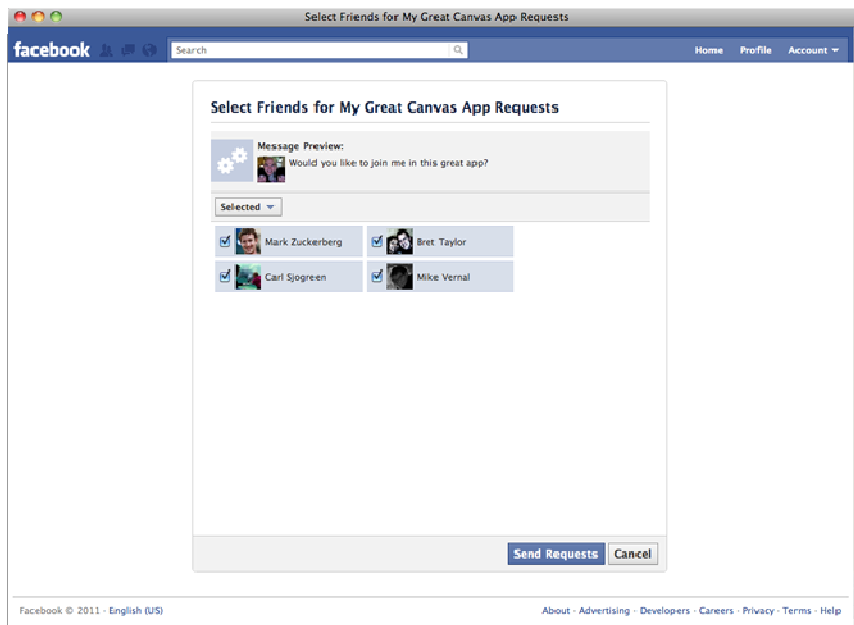

Figure 9: A way of recommending a Facebook app to a friend (15)

View leader board

This is a feature included in GWAP's ESP Game (see Figure 10), that allows a player to view a list of high scores for a specified period of time. With our version, the names given would be Facebook names, and so would the images.

Figure 10: GWAP's ESP Game's use of the leader board

Post about high score

If the user earns a place on a leader board, then they may wish to post a message about it on their timeline. An example of how someone can post to their timeline from within an app is given in Figure 11.
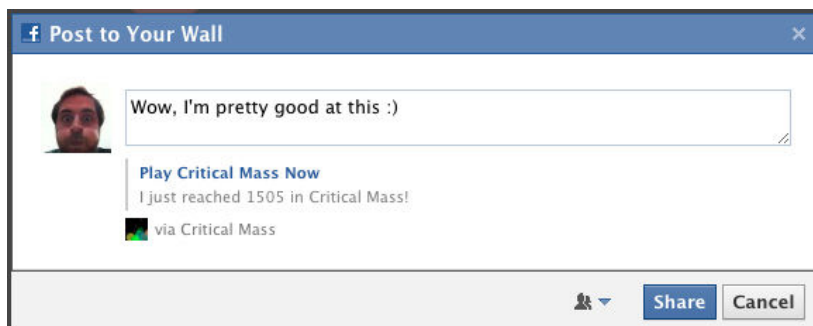

Figure 11: An example of how one may post to their timeline ("wall" when this image was taken) (16)

Review tags

This feature is also similar to GWAP's ESP Game's feature, but rather than showing the first entered term for each image (see Figure 12), our version of the game would simply show the "agreed upon" tag for each image for simplicity.
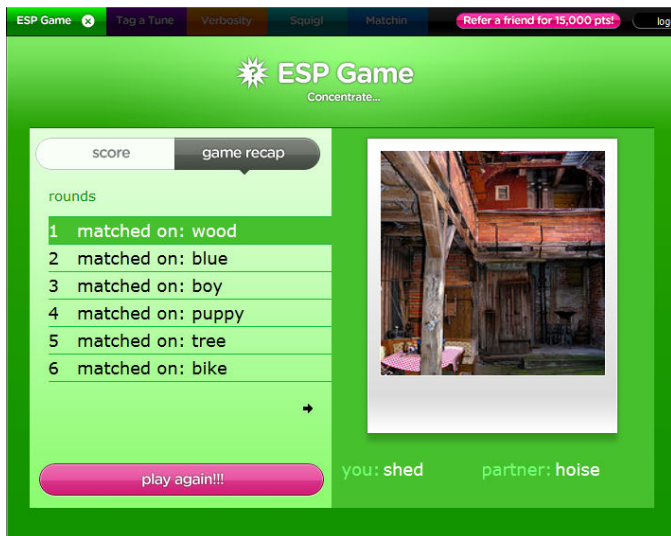
13

Figure 12: Using GWAP's ESP Game, we can see the first guess each player entered, as well as the term the players matched on

Choice of languages

We originally decided that the system should be implemented as a Java applet using sockets, but this was decided against once we had found out about Node.js; this is similar to JavaScript and is "perfect for data-intensive real-time applications" (17, paragraph 1). (The author is referred to the official Node.js website (17) for more information.) We shall be using Node.js with Socket.IO (18). Heroku (19) has been decided upon for hosting the app as this is one of the few free app hosts that has both Node.js and Socket.IO support (20), and also, Heroku can be chosen as the hosting provider when setting up a Facebook app, meaning that an app with some basic functionality can be set up quickly and easily (21). We chose to use the Heroku PostgreSQL (22) add-on to be our database simply due to its relative ease-of-use, and the availability of a program with a GUI (pgAdmin) (23) that allows for easy management.

Database design

To work out the database structure, we first listed the data that would need to be stored:
For each image:
- Unique identifier
- Path in the directory the images are stored in
- Creator of image (for referencing)
- The last set of tags a player playing with this image entered with times (by default, record the "moves" of the first player who entered a guess, and the number of seconds into that round that tag was entered)
- Tags agreed on for an image (taboo words)
- Tags which are below the threshold for a particular image, and how many more times this tag has to be agreed on for it to be used

For general scoring, we will also need to store information about each user:
- For core functionality:
  - Unique identifier linked to that Facebook account
  - Score for this game
- For optional functionality:

14

o   Accumulated score

From this, we came up with the following tables:

Tags (Tag_ID, Tag_Text) - stores all of the tags that have ever been entered, with a unique identifier for each

Image (Image_ID, Path, Creator) - stores all of the game's images, each with a unique identifier, and the path to that image and creator

Taboo (Image_ID, Tag_ID) - stores the taboo tags for an image

Non_Taboo (Image_ID, Tag_ID, Agrees_Left) - stores the non-taboo tags for an image; "Agrees_Left" is used to record how many more times that tag has to be "agreed on" for that image for it to become taboo

Previous_Guesses (Image_ID, Seconds, Tag_ID) - for each image, we store one player's guesses alongside the number of seconds into that round they entered the guess; used for replaying a user's actions if a player doesn't have a human partner

Users (User_ID, Score, Total_Score) - each user's score and total score so far
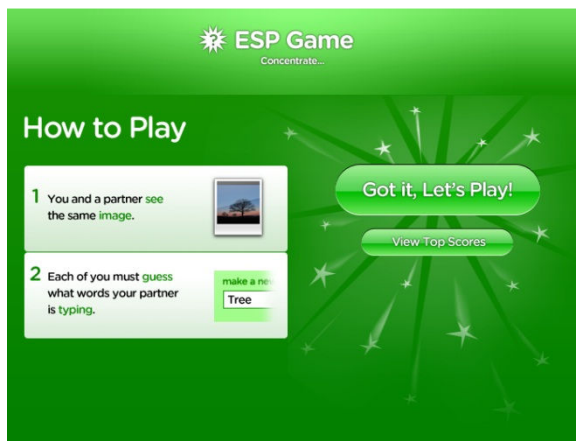
Interface

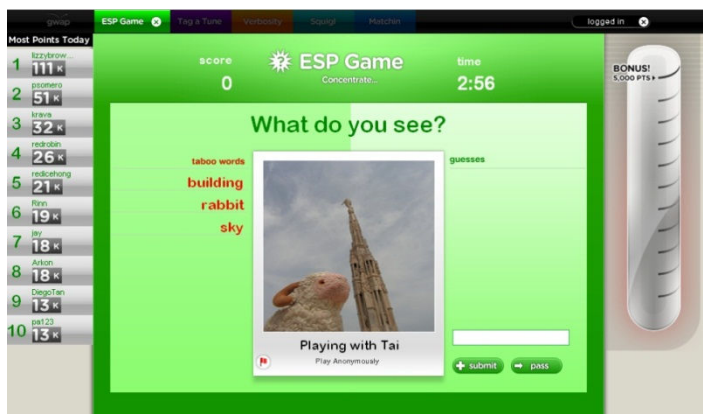*Figure 13: GWAP's ESP Game's splash screen*

Figure 14: GWAP's ESP Game's game screen

The first thing we did to move towards the final interface design was to establish that there would be two screens that would need to be designed (the splash screen and the game screen, see Figures 13 and 14 for GWAP's screens). We decided to base our designs on their layout,

15

refining it to better suit a Facebook game. For example, we decided against having a "playing with <name>" field considering that if a user knows that they are playing the game with a Facebook friend, then they could start up a Facebook chat with that friend and "cheat" on the game to win points, e.g. they could agree to just type "a" for every image, so that they always match. A couple of designs for the splash screen and the game screen were developed using PowerPoint, shown in Figures 15 and 16.

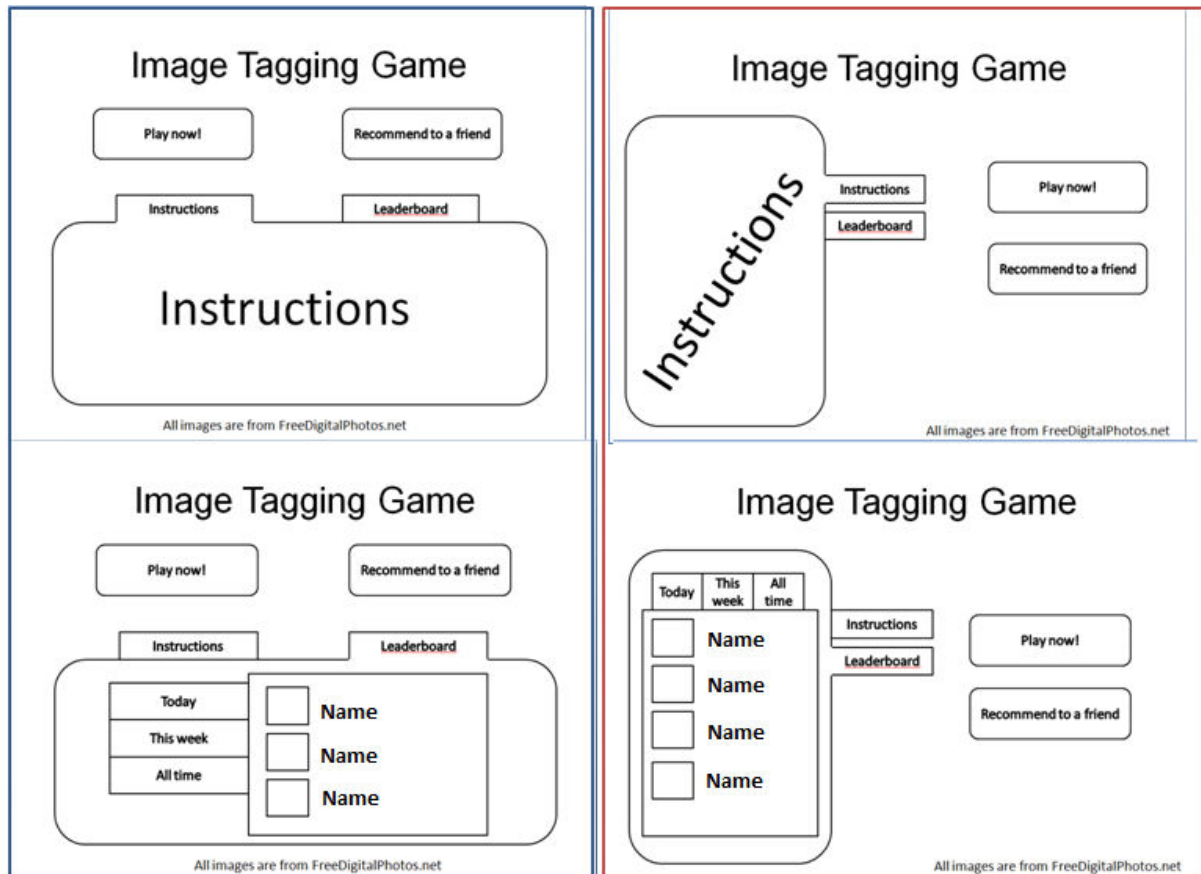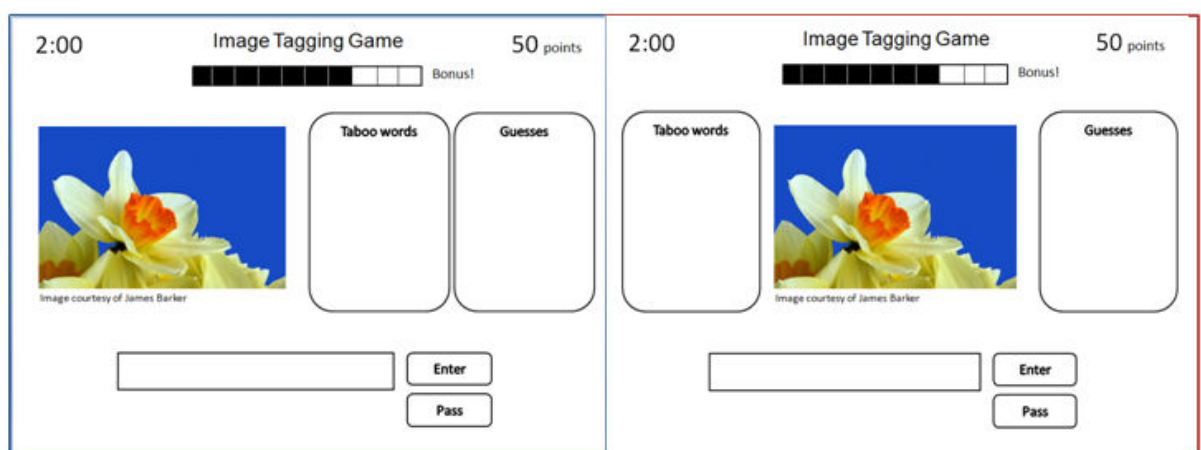*Figure 15: The two options for a splash screen given in the first survey*

Figure 16: The two options for a game screen given in the first survey

We took these options and put them into a survey, making sure to include a description of how the game will be played, as not all respondents may be aware of the concept of an "ESP Game". A link to the survey was posted on the COMSC Facebook group, and we closed the

survey after it had been live for a week, summarising the relevant responses in a series of tables. You can see this summary in Appendix C.
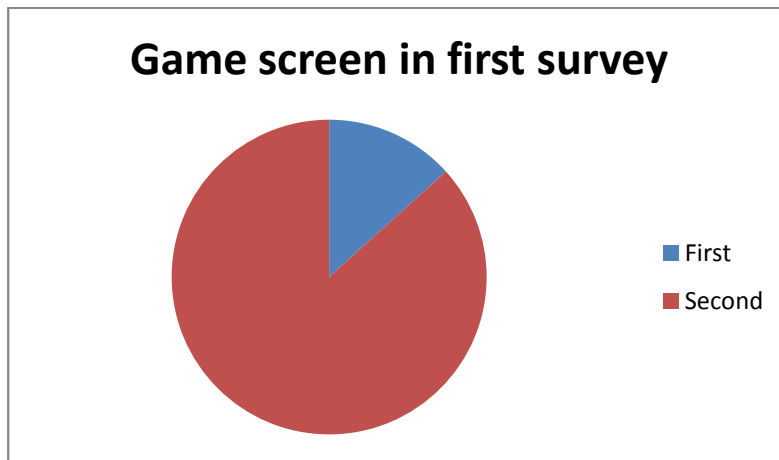

Figure 17: Pie chart showing the distribution of preference for the game screens in the first survey; there were 15 responses total, with 13% preferring the first screen and 87% preferring the second

For the game screen in the first survey, there was a clear split between the first and second options (see Figure 17), with the second coming out the winner. Taking into account the feedback collected about the game screen, we created the finished game screen, seen in Figure 18.
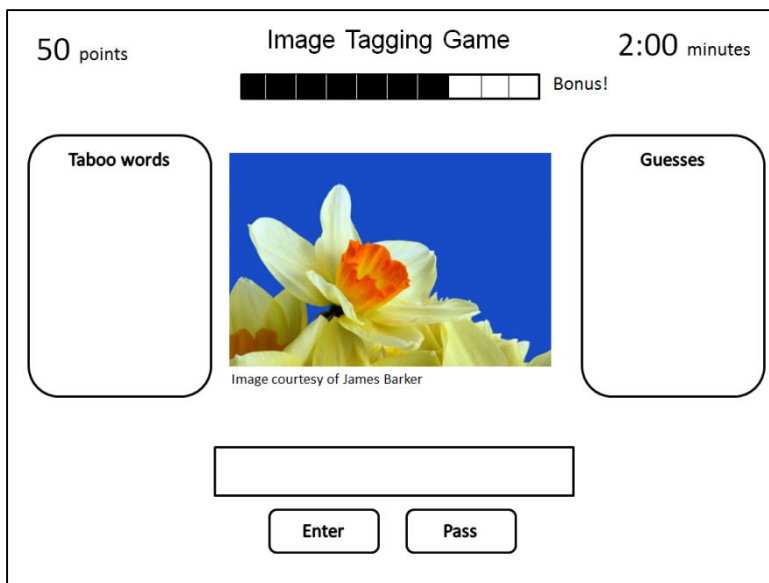

Figure 18: Finished game screen; improvements made:
- Placed "Enter" and "Pass" buttons under text box
- Added the word "minutes" after the time to show that it is the time remaining
- Swapped positions of time and points

However, there was less of an obvious divide between the first and second interface designs for the splash screen; see Figure 19.

Figure 19: Pie chart showing the distribution of preference for the splash screens in the first survey, with 53% preferring the first and 47% preferring the second

At this point we decided to take the feedback collected in the survey and use it to improve the splash screen designs, and then posted another survey on the COMSC Facebook group to get more feedback on these improved designs. Using the feedback from the first survey, the designs for the second were as in Figure 20.
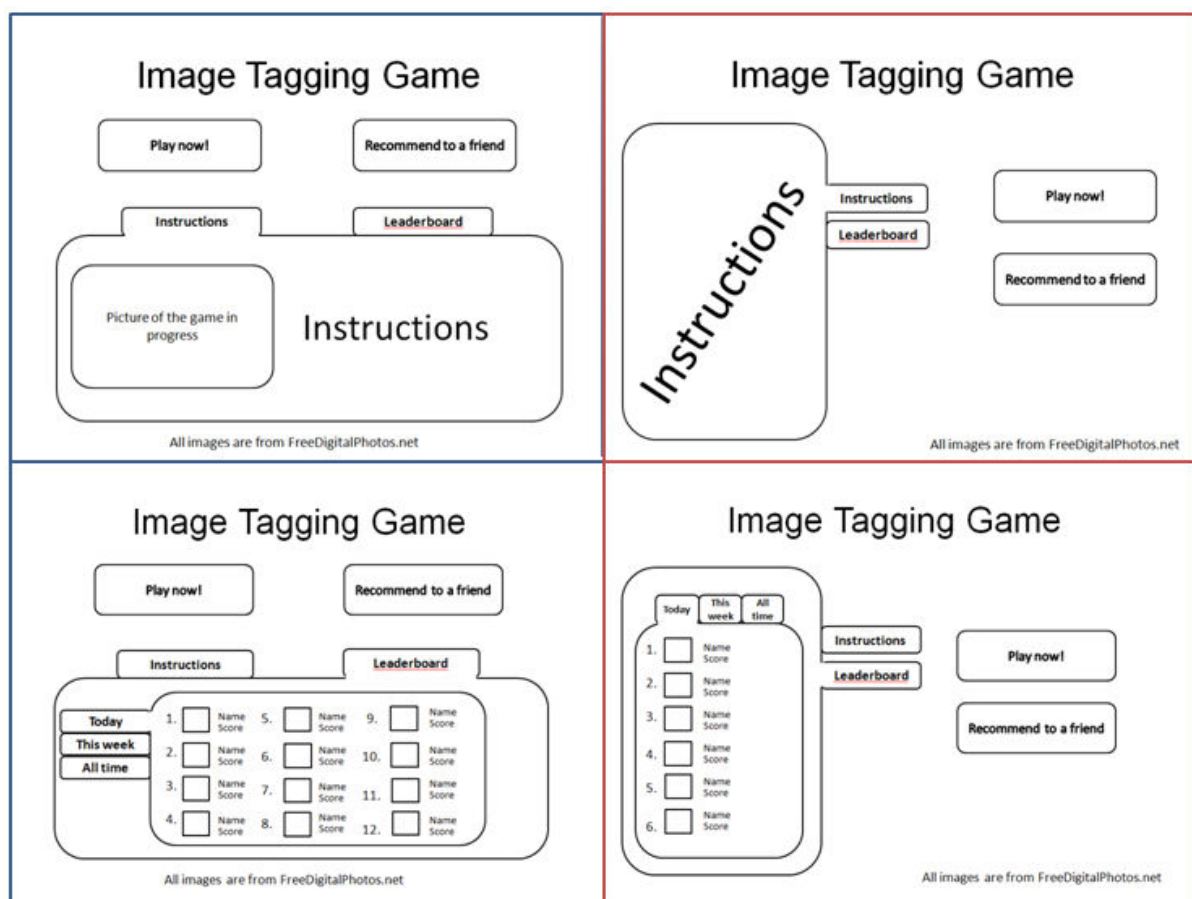
Figure 20: Updated splash screens taking into account feedback from first survey; these were used as options in the second survey. Improvements made:
- All curved shapes
- Scaled down profile pictures and names
- Numbering and scores on leader board (own improvements)

- Added an image of the game in progress (first)
- Scaled down "Today", "This week", and "All time" tabs (first)

From the second survey, which gained 12 responses in the week it was online, the first design was more popular again (see Figure 21), and also, it was found that all but one of the respondents who had not taken the first survey preferred the second interface. The summary of the responses from the second survey can be found in Appendix D. We took the responses/improvements suggested, filtered out the irrelevant responses, and used these to improve upon the more popular screen (this became the final design, shown in Figure 22). It could be argued that we should have gone through another iteration of the survey-improvement cycle, but we feel that the design is secondary to the functionality in this particular project, and perhaps the design of the system could be explored in further work, so we decided to leave it at that.
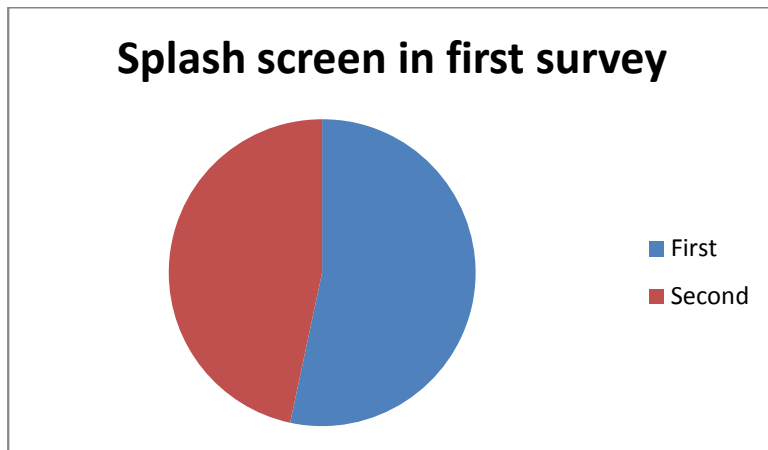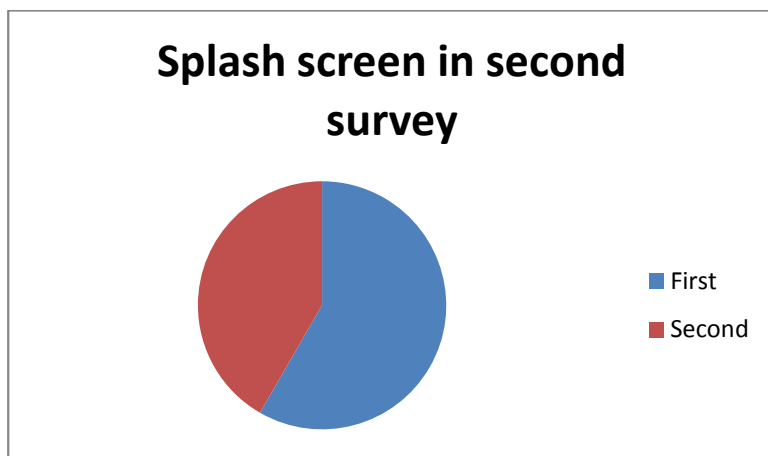

Figure 21: Pie chart showing the distribution of preference for the splash screens in the second survey; 58% preferred the first while 42% preferred the second
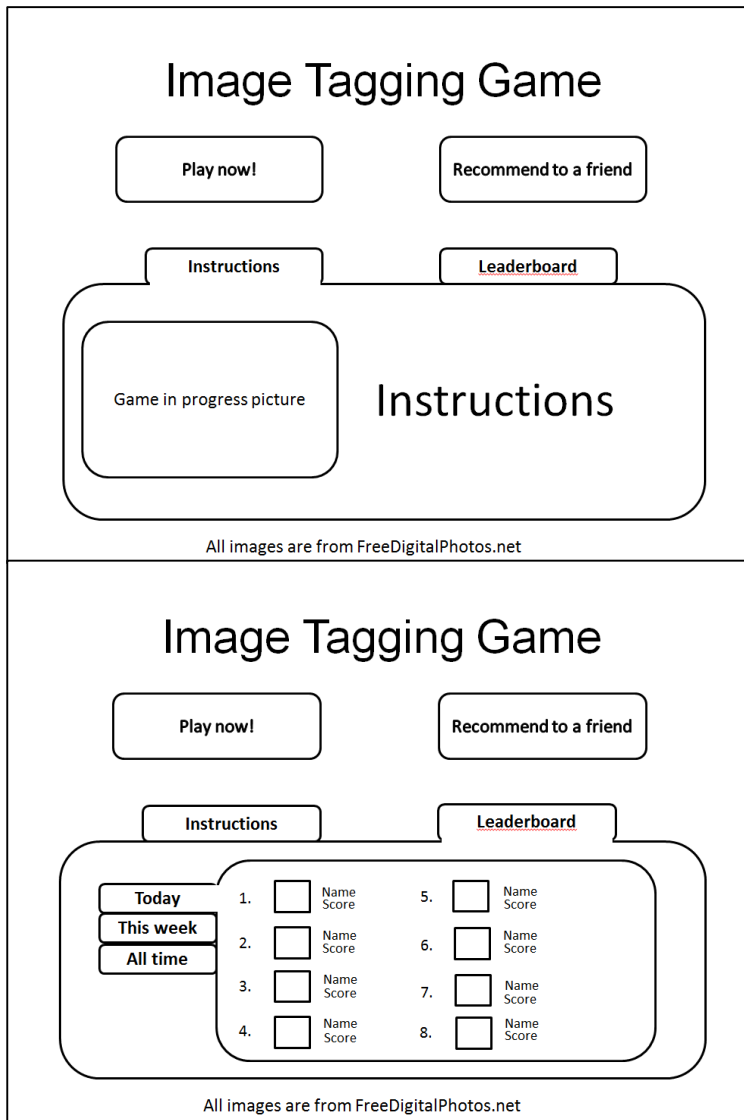
Figure 22: Finished splash screen; improvements made:
- Moved leader board so there is not a gap on the right-hand side
- Reduced number of people on the leader board

Method

Considering how we have already planned our work for the next semester by outlining the timescale for each function to be complete and integrated into the system in the Initial Plan, and taking into account that we accept that the design detailed in this report is subject to change, we feel that the most appropriate paradigm to use when developing the system is agile development. Rather than using a specific agile methodology, we feel that simply adhering to the twelve principles (24) as they fit the project will be most appropriate. Particular principles we would like to focus on during the development of the system are:

- "Our highest priority is to satisfy the customer through early and continuous delivery of valuable software" (24) – Focus is on core requirements.
- "Deliver working software frequently, from a couple of weeks to a couple of months, with a preference to the shorter timescale" (24) – We have just over a semester to work on this, so we plan on implementing and integrating a new feature every few weeks, according to our work plan.
- "Working software is the primary measure of progress" (24) – Relates to the above.
- "Simplicity--the art of maximizing the amount of work not done--is essential" (24) – Focus on the core requirements, again, due to the limited time that we have.

We shall now describe the testing that we will undertake during the system's development.

Acceptance testing
We will mostly focus on this type of testing because this is a relatively small project, and the focus of it is on the production of the software and its functionality. This will involve making sure that the requirements detailed in the Requirements section of this report have been fulfilled. This will be supplemented with lower-level testing as the development is taking place, to ensure that the outputs from the various parts of the system are as expected.

Evaluation of accuracy of tags collected
This type of evaluation was used with the original (von Ahn's) ESP Game, and involved using such methods as asking people to manually tag images to compare against the tags collected using the game (2). Although it was mentioned in the Initial Plan that the tags caught through the system would be evaluated in a similar way, this was decided against due to the time constraints the project has, so this will be left for future work. Instead, we shall calculate the inter-rater reliability, probably using some tool such as ReCal3 (25). We shall use such a tool to measure the degree of agreement between different users in tagging images with certain tags. To do so, we shall have to collect quite a few tags, so time shall be allowed for this towards the end of the project.

Conclusion

In this report, we have presented an analysis of the background of the project, including its context, as well as information on current versions of the ESP Game, in particular GWAP's ESP Game, and how they are lacking. We have established that GWAP's ESP Game is lacking mostly because of the fact that it is presented on a website that may not be known about unless through word-of-mouth. To remedy this, we shall implement our version of the game as a Facebook application, which could potentially reach a much wider audience, and therefore increase its value as it would have the potential to collect more tags quicker.

We have formalised the functional requirements of the system, proposed a design of the system, and broken it down into modules based on functions. From this, we have put forth a use case diagram, and an activity diagram. We have also discussed the database and user interface designs, as well as the choice of languages for implementing the system.

We have discussed how the system shall be tested, mostly through acceptance testing towards the end of the project's development phase, but also through smaller tests as development is going on. We shall also use inter-rater reliability to evaluate the accuracy of tags collected.

In the next semester, we shall be developing the system according to the core functional requirements, and if time permits, we shall add some of the optional functionalities as well. In the final report, our system shall be discussed according to its functionality, and code snippets will be provided where appropriate. Testing results will be presented as well as a general evaluation of the system and our work in producing it. Proposed future work will also be mentioned.

Glossary

Collaborative tagging system A system which allows large groups of people on the internet to assign tags to some content, such as a URL, image, etc.

Computer vision "The technology concerned with computational understanding and use of the information present in visual images" (26)

Game With A Purpose (GWAP) A game that uses the collective intelligence of humans to improve machine learning

Image search A feature that allows for the searching of a database of images, usually through inputting text strings which are compared against text associated with each image in the database. A well-known example of such a system is Google Images (1)

Inter-rater reliability "Used to assess the degree to which different raters/observers give consistent estimates of the same phenomenon" (27)

Metadata Data about data, or information about information. For example, metadata for a book could be its title, author, ISBN, etc.

Screen reader "Software that reads the content of a computer screen aloud. Screen readers can only interpret text content, so all [graphics] and multimedia must have alternative text descriptions using ALT text, captions, transcripts, or other methods" (28)

Tag A word, or phrase, assigned to some piece of data, such as an image, to describe that data.

Appendix A - Java Parser Code

```java
import java.io.*;

class Parser {
	public static void main(String[] args) {
		try {
			BufferedReader br =
				new BufferedReader(new FileReader("taginfo.xml"));
			String line;
			FileWriter writer = new FileWriter("tagfreq.csv");
			while ((line = br.readLine()) != null) {
			  if ((line = (line.trim())).startsWith("<url>")) {
					line = line.replaceAll("<url>", "");
					line = line.replaceAll("</url>", "");
					writer.append("");
					writer.append(line);
					writer.append("");
					writer.append(',');
					writer.flush();
				}
				else if (line.trim().startsWith("<users>")) {
					line = line.replaceAll("\\D+", "");
					writer.append(line);
					writer.append(System.getProperty("line.separator"));
					writer.flush();
				}
			}
			br.close();
			writer.close();
		} catch (Exception e) {}
	}
}
```

24

Appendix B – Use Case Descriptions

Use Case Number: 1	*Use Case Name:* Authorise application	*Rating:* Core
Description: The player is asked for permission to access particular pieces of their data, and they either accept or decline this. If accepted, they can then go on to access the game; if not, they cannot.		
Main Actor: Player	*Secondary Actors:* N/A	
Pre-conditions: The player has a Facebook account and is logged into this account.		
Basic Flow: 1. The user locates the application (this could be through seeing a friend's post about it, or through the search facility, for example) and clicks on its link. 2. The system presents the user with a login dialog, detailing the application's required permissions. 3. The user reviews these permissions, and also selects who they would like to be able to see posts generated by the application. 4. The user clicks a button to say they accept these permissions.		
Alternative Flows: 1. Declined permissions - if, once reviewing the permissions required for the application to run in Step 3, the user does not agree with them, the user shall click a button to indicate this. They cannot use the application until they authorise it. The use case ends.		
Post-conditions: If successful: the user authorises the application. If unsuccessful: nothing changes.		

Use Case Number: 2	*Use Case Name:* Play a game	*Rating:* Core
Description: The player plays the game, tagging images in the process.		
Main Actor: Player	*Secondary Actors:* N/A	
Pre-conditions: The player has a Facebook account and is logged into this account, and they have authorised the application (Use Case 1).		
Basic Flow: 1. The player clicks the "Play now" button on the game's splash screen. 2. The system presents an image to the player, with a list of words that cannot be entered ("taboo words") and a textbox to enter their guesses into. 3. The player enters a guess into the textbox and clicks the button to submit. 4. Step 3 continues until either: the players "match" on a guess, or one of them clicks "Pass", forcing the other player the pass on the image too to continue. 5. The flow beginning at Step 2 continues until time runs out.		
Alternative Flows: 1. The player exits: the player decides, at some point in the flow after Step 1, that they no longer wish to play the game. They exit the game, and the use case ends.		
Post-conditions: If successful: the user has successfully finished the game. If unsuccessful: nothing changes.		

Use Case Number: 3	*Use Case Name:* Recommend to a friend	*Rating:* Core

Description: The player recommends the game to a Facebook friend by posting a message on their friend's timeline from within the application.

Main Actor: Player	*Secondary Actors:* N/A

Pre-conditions: The player has a Facebook account and is logged into this account, and they have authorised the application (Use Case 1). They have at least one Facebook friend.

Basic Flow:
1. The player clicks the "Recommend" button on the game's splash screen.
2. A dialog box is displayed allowing the user to select the appropriate Facebook friend(s).
3. The player selects their friend(s), and clicks a button to confirm.

Alternative Flows:
1. The player exits: the player decides, at Step 2, that they no longer wish to recommend the game to their friend(s), and they cancel the dialog box. The use case ends.

Post-conditions: If successful: a post appears on the player's friends' Facebook timeline(s) recommending the game to them.
If unsuccessful: nothing changes.

Use Case Number: 4	*Use Case Name:* View leader board	*Rating:* Optional

Description: The player views the players of the game with the highest score over the past day, week, and since the game was posted.

Main Actor: Player	*Secondary Actors:* N/A

Pre-conditions: The player has a Facebook account and is logged into this account, and they have authorised the application (Use Case 1).

Basic Flow:
1. The player clicks on the "Leader board" button on the game's splash screen.
2. The system presents the list of the players with the top 5 scores over the past day.

Alternative Flows:
1. If the player wishes to view the top 5 scores over the past week, they may click a tab saying "Last week".
2. If the player wishes to view the top 5 scores of all time (i.e. since the game was posted), they may click the "All time" button.

Post-conditions: The player has viewed the leader board(s) of their choosing.

Use Case Number: 5	*Use Case Name:* Post about high score	*Rating:* Optional

Description: The player posts about their place on the leader board on their timeline.

Main Actor: Player	*Secondary Actors:* N/A

Pre-conditions: The player has a Facebook account and is logged into this account, and they have authorised the application (Use Case 1). They have played a game (Use Case 2), and have achieved a high score.

Basic Flow:
1. The system presents the player with a dialog to post to their timeline.
2. The player enters a message about their place on the leader board.
3. The player clicks a button to submit the dialog.

Alternative Flows:
1. The player exits: the player decides, at Step 2, that they do not wish to post about their place on the leader board, and they cancel the dialog box. The use case ends.

Post-conditions: If successful: The player has posted to their timeline about their place on the leader board.

If unsuccessful: Nothing changes.

Use Case Number: 6	Use Case Name: Review tags	Rating: Optional
Description: The player reviews the tags agreed on for each image throughout a game.		
Main Actor: Player	Secondary Actors: N/A	
Pre-conditions: The player has a Facebook account and is logged into this account, they have authorised the application (Use Case 1), and they have played a game (Use Case 2).		

Basic Flow:
  1. The system displays a list of the format Image ID : Tag agreed upon. "Image ID" is simply defined by the ordering of the images in the game, i.e. the first image shown is Image 1, the second is Image 2, etc.
  2. The player clicks on an item in the list.
  3. The system displays that particular image next to the list.
  4. The flow beginning at Step 2 continues until the user no longer wishes to review tags.		

Alternative Flows: -		
Post-conditions: The player has reviewed the tag(s) of their choosing.		

Appendix C – Survey 1 Response Summary

(We simply copied the responses as they were, so any spelling/grammar errors you may find are not ours. The column a response is in determines whether that player selected the first or the second option as their preferred display; for example, the comment "There is less white space…" was the reasoning for why someone chose the first option as their favourite in Q1.)

Q1. Given the above two options for a splash screen interface, which do you prefer, and why? (For each option, the top shows when the Instructions tab has been clicked, and the bottom shows when the Leaderboard tab has been clicked. Also, the square boxes next to each "Name" on the Leaderboard are for the person's Facebook profile picture.)

The first (with blue border) **8**	The second (with red border) **7**
- There is less white space, and overall it looks more complete. - The first interface is much more symmetrical, which is good but the second interface seems too weighted on the left side of the screen, which makes it look uneven. - For some reason, laying things out horizontally looks better and more concise. - From context I'm assuming that the most used button will be Play Now, with the second point of focus for for new users being the instructions and the leaderboard for experienced users and Recommend to a Friend being a more occasional/optional use case. Option one lays out the information much more logically for this flow, and also has less overall whitespace than option two. However, this is at the sacrifice of having a less neat layout for the instructions/leaderboard subsection. - The layout seems to flow better, and places more emphasis on the body of the content, rather than splitting it in two. - The play button is in a more immediately intuitive position. If I want to play a game, the play button should be the first thing that I am drawn to, given my tendency to read left to right and top down, makes the blue bordered one better. - Easier to read with the wider space. Also potential for a picture of the game in progress on instructions screen.	- It's more logical and user friendly. The blue border boxes appear UI appears to just be smashed together, whereas the red considers the user much more. - Second as its a bit less clustered - I prefer the bottom part showing the leaderboard with the names going down length ways its visually more appealing and doesn't look as cluttered. - I like that the leader board is longer, gives you space to see more people on the board. I dont know if the board stays or disappears when the game starts, but if it stays its cool to have it there on the side - I prefer a vertical box as it appears there is less text to read. A horizontal one looks like there is going to be a lot. I dislike reading lots of instructions for a short game with a friend. - Items seem more spaced apart and this makes it the interface seem more open and less cumbersome. Also, for people that have already played the game, they would just like to go straight to the 'Play now!' rather and if it this button is in a separate area from the instructions, again it is easier to find. - Easy on the eye

Q2. Is there anything you would change about the splash screen you chose in the previous question (e.g. put instructions box somewhere else)?

The first (with blue border)	The second (with red border)
- On the leader board the Name and Profile picture are noticeably disproportioned to the rest of the gui and needs to be scaled down to match. Youve got a combination of curved squares/rectangles and corned off one, to keep the design more in tune have either/or but not both, preferably curved shapes as they seem easier on the eyes. - Make the Today/This Week/All Time smaller and more offset on option 1, bringing more focus to the leaderboard content. - Centre the play button to put more emphasis on it. - Smaller space for leaderboard tabs	- One thing I do not like about the second red image (bottom left) is that the fb-profile-picture/name box is larger than the three above it "Today, This week and All time". I think it would look better if it was the same size. Just my thoughts thought. - I prefer the way the instructions run lengthways in the first (with blue border) than in the second one but the second one looks less cluttered.

Q3. Given the above two options for the main game screen, which do you prefer, and why? (Taboo words are words that the player cannot enter as a guess.)

The first (with blue border) **2**	The second (with red border) **13**
- It seems more organised and requires less eye movement to observe key areas of the screen, I found myself flicking left to right more frequently which done for a long time would probably give me a headache. - Tough one! The right one looks awsome but I think it would be annoying to have to keep looking back a to the taboo words and then to the guesses and then to the image. The blue one put the image all the way over to the other end but it makes it easier to look back at the taboo words.	- Again, it's more logical, but not perfect (for me at least) - When the both are together it could be quite difficult to distinguish between the two. seperating them on either sode makes this easier x - Again, the second interaface looks better because it looks more symmetrical. - By putting the image in the middle it does two things for me 1. Makes the image the focus of the Game, and of the screen, and it looks much more symmetrical 2. It splits up the two columns of words, I wouldn't really want to bombarded the information with all the info next to each other. - Option one puts less focus on the image than is needed, wheras option two gives the secondary information a peripheral look - it's likely to be something that's glanced at more than studied, so it works well being set close to the main point of focus without drawing too much attention. - There is better spacing between the two boxes making it easier to read and understand the screen. Plus it looks symmetrical which is more pleasing. - The image, I believe, should be centred to better focus the players. - I like the fact that the image separates the taboo words from the guesses that could get

	me confused whilst playing.
	- The focus of the game (i.e. the image) is centred in the screen and the two columns of words are equally distant from this so you can reference either of them quickly.
	- Makes it appear less crowded
	- Its easier to define between the taboo words and guesses when they aren't next to each other. Particularly, if they are going to have a lot of text in.

Q4. Is there anything you would change about the game screen you chose in the previous question (e.g. put image somewhere else)?

The first (with blue border)	The second (with red border)
- Switch round the Guesses and Taboo word boxes, to me it seems to make more sense to have the guesses closer to the picture as I would more frequently observe this. The game header could be shifted left so that on top there is the timer/title and the points shown. In the footer shift the entry box right and add in the time counter there, also make this bigger so it is of inequivalent size of both the enter and pass buttons combined. Will make it easier to see as it took me a while to even notice there was a time counter. - good designs here. I could be a bit picky about the buttons Enter and Pass being vertically positioned because the alignment looks off. Might be awkward to press then on a small touch screen device too. Would suggest putting them under the text bar in a horizontal position and making them bigger. maybe colouring them blue and red too.	- Yes, why not put Tabbo words and Guesses beside each other on the left hand side then the image on the right? Also, enter and pass would perhaps be better if they were beside each other? - I might put the enter and pass buttons below the text box in the middle so that the whole page now look symmetrical. - Maybe move the enter and pass buttons beneath the enter box so that they're parallel - Maybe move the score to a more prominent position, so as to encourage competitive play, rather than keeping it out of the way in the corner where some players may not take much notice of it. Also centre the text field to line up better with the image, shifting the 'Enter' and 'Pass' buttons to the side. - Enter and pass buttons in line rather than stacked - Yes. I would include the time (I'm assuming its 2.00 minutes, but it doesn't explicitly say) ... Also, include a description of taboo somewhere, it didn't come to me instantly!

Q5. Do you have any other comments about either of the sets of interfaces?

- Just make sure that you use the same styled shaped across the game for everything not using a combination, preferably rounded off corners :)
- I'd possibly swith the clock to the right hand side and the points to the left. I tend to look more towards the right hand side of a screen and so would prefer the clock to be there. (Red)
- keep it simple :)

Appendix D – Survey 2 Response Summary

(We simply copied the responses as they were, so any spelling/grammar errors you may find are not ours. The column a response is in determines whether that player selected the first or the second option as their preferred display, for example, the comment "There is more space to see everything…" was the reasoning for why someone chose the first option as their favourite in Q2.)

Q1. About 2 weeks ago I posted a link to a similar survey to this one on the COMSC Facebook group. Did you take that survey?

6 Yes, 6 No

Q2. Given the above two options for a splash screen interface, which do you prefer, and why? (For each option, the top shows when the Instructions tab has been clicked, and the bottom shows when the Leaderboard tab has been clicked. Also, the square boxes on the Leaderboard show the profile picture, with the user's name and score.)

The first (with blue border) **7 (6 Yes, 1 No)**	*The second (with red border)* **5 (5 No)**
- There is more space to see everything, where as a lot of the important information on the red border one is squished to one side. I also like the idea of a having a screenshot of the game in progress if you've stopped it to re-read the instructions. (Yes) - Like that it has a picture of the game in progress (Yes) - The items populate the screen more so it looks more whole/complete. Looks neater. Everything is centered (enough of it anyway) and looks more consistent. (Yes) - The first layout works better for maximising screen real estate as it uses just under half of the plane its in compared to more like a third in the second design, also takes advantage of human FOV being landscape rather than portrait. The leaderboard is now much more neatly laid out than what I recall, although I'm not sure about the gap to the right of it. I like the clear numbering, too. The second design hasn't changed a lot since the last iteration, as far as I can recall; the today/this week etc boxes still have the issues of appearing cramped in and the instructions/leaderboard appear to be jutting out. (Yes) - seemed neate rin its design and didnt confuse me as much as the other (No)	- I'm not sure just preferred the look of it! (No) - Its easy to follow and navigate in the game (No) - It looks more slick than the first one. I like things on the left. (No) - Elements on the page just seem structured in a better way (No) - It looks much more professional and is an interface I'm more used to seeing. (No)

Q3. Is there anything you would change about the splash screen you chose in the previous question (e.g. put instructions box somewhere else)?

The first (with blue border)	*The second (with red border)*
- Maybe not try to cram in so many people into the leaderboard? If people have large scores or long names it would be cut off or be really small text and hard to read, 1-8 only would give them a longer word space to populate. (Yes) - perhaps fade away the options the user did not click on. at first it can be slightly confusing in seeing that the leaderboard option is chosen, not instructions (No)	- I would make sure that the default for the screen is the leader board. people rarely read instructions for games (No) - I'd perhaps have the leader board be open by default and then having instructions underneath the play now button. If possible, I'd then have a separate window for instructions on click. (No) - Place the "Play now!" and "Recommend to a friend" buttons on the left and the instructions and leaderboard box on the right. Then put the "instructions" and "leaderboard" tabs on top. (No)

Q4. Do you have any other comments?

- I like that the boxes and buttons are round. It looks nicer! (Yes)
- FIRST ONE :D (Yes)
- Could there be a facebook link or something? (No) *This is a Facebook game*
- The game needs a proper name. "Image tagging game" doesn't sound fun! (No) *I will probably change the name, but right now its name is not really an issue*

Our comments are given in Q5 in italics. The (Yes) and (No) after each response denotes the respondent's answer to Q1.

References

1. Google. *Google Images*. Available at: https://www.google.co.uk/imghp?hl=en&tab=wi [Accessed: 12th November 2012].

2. von Ahn, L and Dabbish, L. 2004. Labeling Images with a Computer Game. *ACM Conference on Human Factors in Computing Systems, CHI 2004*. Vienna, Austria, 24-29 April, 2004. pp 319 – 326.

3. GWAP. *ESP Game*. Available at: http://www.gwap.com/gwap/gamesPreview/espgame/ [Accessed: 10th October 2012].

4. GWAP. *Tag a Tune*. Available at: http://www.gwap.com/gwap/gamesPreview/tagatune/ [Accessed: 27th November 2012].

5. GWAP. *Popvideo*. Available at: http://www.gwap.com/gwap/gamesPreview/popvideo/ [Accessed: 27th November 2012].

6. Schneiderman, H. and Kanade, T. 2004. Object Detection Using the Statistics of Parts. *International Journal of Computer Vision* 56(3), pp. 151 - 177.

7. Delicious. *Delicious*. Available at: http://delicious.com/ [Accessed: 19th October 2012].

8. Natural Language Processing and Information Retrieval Group at UNED. *DeliciousT140 Dataset*. Available at: http://nlp.uned.es/social-tagging/delicioust140/ [Accessed 1st November 2012].

9. YouTube. *YouTube – Broadcast Yourself*. Available at: http://www.youtube.com/ [Accessed: 19th October 2012].

10. Flickr. *Welcome to Flickr – Photo Sharing*. Available at: http://www.flickr.com/ [Accessed: 19th October 2012].

11. GWAP. *gwap.com – Home*. Available at: http://www.gwap.com/gwap/ [Accessed: 27th November 2012].

12. Barker, J. *Daffodils*. Available at: http://www.freedigitalphotos.net/images/Flowers_g74-Daffodils_p423.html [Accessed: 18th October 2012].

13. Indiana University South Bend. *Software Architecture Introduction*. Available at: http://centurion2.com/SEHomework/Chapter11/SoftwareArchitecture.php [Accessed: 19th November 2012].

14. Ivins, W. Software Design. *System Design and Group Project*. Cardiff University. Available at: http://cue.cf.ac.uk [Accessed: 19th November 2012].

15. Facebook. *Canvas Tutorial*. Available at: http://developers.facebook.com/docs/appsonfacebook/tutorial/ [Accessed: 30th November 2012].

16. Facebook. *Social Games Tutorial*. Available at:
http://developers.facebook.com/docs/guides/games/getting-started/ [Accessed: 20th November 2012].

17. Joyent. *node.js*. Available at: http://nodejs.org/ [Accessed: 5th December 2012].

18. Rauch, G. *Introducing Socket IO*. Available at: http://socket.io/ [Accessed: 7th December 2012].

19. Heroku. *Heroku*. Available at: http://www.heroku.com/ [Accessed: 7th December 2012].

20. Heroku. *Using Socket.IO with Node.js on Heroku*. Available at:
https://devcenter.heroku.com/articles/using-socket-io-with-node-js-on-heroku [Accessed: 7th December 2012].

21. Heroku. *Getting Started with Your Facebook App on Heroku*. Available at:
https://devcenter.heroku.com/articles/facebook [Accessed: 7th December 2012].

22. Heroku. *Heroku Postgres*. Available at: https://devcenter.heroku.com/articles/heroku-postgresql [Accessed: 7th December 2012].

23. pgAdmin. *pgAdmin*. Available at: http://www.pgadmin.org/ [Accessed: 7th December 2012].

24. Beck, K. et al. *Principles behind the Agile Manifesto*. Available at:
http://agilemanifesto.org/principles.html [Accessed: 3rd December 2012].

25. Freelon, D. *ReCal3: Reliability for 3+ Coders*. Available at:
http://dfreelon.org/utils/recalfront/recal3/ [Accessed: 13th December 2012].

26. The Free Dictionary. *Computer vision*. Available at:
http://encyclopedia2.thefreedictionary.com/computer+vision [Accessed: 12th November 2012].

27. Trochim, W.M.K. *Types of Reliability*. Available at:
http://www.socialresearchmethods.net/kb/reltypes.php [Accessed: 13th December 2012].

28. Keynote NetMechanic. *Accessibility Dictionary*. Available at:
http://www.netmechanic.com/accessibility/glossary.shtml [Accessed: 27th November 2012].