

Table of Contents

1. Introduction	
1. ¹ Overview.....	2
1. ² Business Context.....	2
1. ³ Glossary.....	3
2. General Description	
2. ¹ Product/System Functions.....	4
2. ² User Characteristics and Objectives.....	4
2. ³ Operational Scenarios.....	5
2. ⁴ Constraints.....	6
3. Functional Requirements	
3. ¹ Sign Up/Register.....	7
3. ² Create Profile.....	7
3. ³ Upload Images.....	8
3. ⁴ Caption Images.....	8
3. ⁵ Log In.....	8
3. ⁶ Edit Profile.....	9
3. ⁷ Add/Remove Friends.....	9
3. ⁸ Manage/Delete Images.....	9
3. ⁹ Delete Profile.....	10
3. ¹⁰ Rate Profile.....	10
3. ¹¹ Message User.....	10
3. ¹² Search Database.....	11
3. ¹³ Tour the Site.....	11
4. System Architecture	
4. ¹ System Architecture Diagram (Fig 4.1).....	12
4. ² Web Site.....	12
4. ³ PHP Scripts.....	12
4. ⁴ MYSQL Database.....	13
5. High-Level Design	
5. ¹ High Level Design Diagram (Fig 5.1).....	14
5. ² High Level Description.....	15
6. Preliminary Schedule	
6. ¹ Overview of Preliminary Schedule.....	16
6. ² Task List (Fig 6.2).....	16
6. ³ Gantt Chart (Fig 6.3).....	17
7. Appendices	18

1.1 Overview

The product being developed is best described as an online community. It will allow people to join and interact with each other in various ways. Members will be able to browse other members of the community, with reference to their similarities. They can then interact with people through the use of an internal messaging system.

The system was developed to serve the needs of people with similar interests from anywhere in the world to be able to interact with other people. On this site people will display some personal information in their personal profiles and based on this information, other members can choose to get in contact by messaging them using our internal messaging system. Various other functions are available and these will be listed and explained more in depth later on.

Other features that we are toying with include a statistics area of the site where we can monitor how many users we have signed up, as well as the amount of messages users have sent each other and the average rating of profiles on the system. However all of this is in a very early stage.

1.2 Business Context

There are two possible business contexts in relation to this product. These contexts are:

- **Selling the Product:**
The product could be potentially sold to an external organisation. This organisation could potentially use this product as it is or they could choose to integrate it into an existing product.
- **Advertising:**
The product could also have the additional feature of internal advertising within the product. This could be general advertising, or it could be more targeted advertising relating more specifically to different peoples interests as stated in their profile and could generate funds from advertisers.

These contexts are both viable independently of each other, but at the same time it is also conceivable that these two contexts could be used in conjunction with each other.

1.3 Glossary of Terms

- **PHP:**
Recursive acronym for **PHP Hypertext Processor**.
An open source, server-side, HTML embedded scripting language used to create dynamic Web pages.
- **MYSQL:**
MYSQL is an open source RDBMS that relies on SQL for processing the data in the database.
- **Apache Server:**
The Apache Web Server is a public domain, open source Web server.
- **Redbrick:**
D.C.U.'s networking society. Our project will be hosted on Redbrick's Apache web server (Deathray) that has both PHP and MYSQL installed.
- **HTML:**
Acronym for: **Hypertext Mark-up Language**.
This is the authoring language used to create documents on the World Wide Web.
- **Session:**
The period of time a user interfaces with an application. The user session begins when the user accesses the application and ends when the user quits the application. This also refers to the amount of time a user uses a website for. The session starts once the user logs in and finishes when the user logs out.
- **Cookies:**
A message given to a Web browser by a Web server. The browser stores the message in a text file. The message is then sent back to the server each time the browser requests a page from the server. Cookies can contain information about a user in order to personalise certain web sites.

2

General Description

Section

2.1 Product/System Functions

Below is a list of the main functions of our project. This is a preliminary list and is open for additions should we think of anything worth adding. Each function has with it; it's own parameters, which will be discussed in more detail in section 3.

- Sign Up
- Create Profile
- Upload Image(s)
- Caption Image(s)
- Log In
- Edit Profile*
- Add/Remove Friends*
- Manage/Delete Image(s)*
- Delete Profile*
- Rate Profiles*
- Message Users*
- Search Profiles by criteria
- Tour

*Requires login

2.2 User Characteristics and Objectives

As the system will be hosted online, the product is accessible to anybody with a computer and access to the Internet. The target audience will be males and females between the ages of 18-30.

The site will be easily accessible with a user-friendly web interface so as to accommodate users with little or no knowledge of computing. Ideally the site would attract single people (within DCU desirably) and encourage them to meet with others and possibly spark up a romance. (Anything is possible).

The site will be moderated also. This enables users of a younger age group to use the site without being offended. A method of accurate moderation has still to be decided.

2.³ Operational Scenarios

Due to the design of our proposed website, there will be different scenarios under which the website will have to perform. These include different authorisation levels for users and different functions available considering these levels. While it is possible to prompt for authorisation (user name & password) on visiting the site, we reckon that by not doing this we can give users a reason to sign up. As far as we can imagine there will be 3 different user levels. These are...

- **Unregistered User:**

The unregistered user will be a casual browser who just happened to come across the site. For this user, we will need certain functions available to try and tempt them to join. We reckon that the most feasible functions to use to achieve this goal are:

- **Search**

Let the user search the database by specified criteria, allowing them to find current members that take their fancy, whether this is by look or common interests, or even location.

- **Rate**

The unregistered user has the ability to rate profiles before actually signing up. This gives the unregistered user a sense of what it's like to participate on the website and may encourage them to join. If they give someone a good rating, we can show a message saying that if they sign up, they can message this user.

- **Tour**

Touring the site will simply consist of a few pages, each letting the visitor know what the benefits of joining are.

- **Registered User (not logged in):**

The registered user shall also have a limited amount of functionality before they login. This scenario is similar to the one above because the same options are available to the user, but only until they log in. As well as the above functions, this scenario makes one additional function available. This is...

- **Log In**

The registered user, unlike the unregistered user, has the ability of logging in to make the most of the full range of functions that the site has to offer.

- **Registered User (logged in):**

This scenario makes use of the full function list of the site. This is the main area that we feel will be use the most and it offers a lot of functionality to the registered user. Certainly a benefit to them for taking the short time it takes to sign up.

- **Message***

This is a function only available to registered users. It allows the user to send another user a message (within the site). This is the first point of contact, and the whole idea and success of the project relies on being able to implement this function.

- **Add/Remove Friends***

Each user profile will contain a "friend" list. This can consist of members that the person knows in real life, or that they have met through the site, or even members they like the look/profile of.

- **Edit Profile***

Allows a logged in/registered user to update their personal profile with newer or more accurate information regarding them. You must have a profile for this, and in order to get a profile you must register.

- **Upload Images***
Images add to the content of your personal profile and can mean more people getting in touch with you.
- **Caption Images***
This is a fun feature that we thought would be good to add. It offers you the ability to add fun captions to your images, in order to explain to visitors what's going on in that particular image.
- **Log Out***
This is a secure method of closing the session connection. This will modify the cookie that logging in created.

*Requires login

2.4 Constraints

The only real constraints we face are regarding storage space and the deadline. These are detailed below.

- **Database Memory**
Because of the limit on our server account and MYSQL account, we have a limit on the scale on which we can test the project. We will be unable to test the product on the scale that it would possibly grow to in time, and would face problems later on, as more and more people sign up.
- **Moderation**
The moderation of the site also promises to be tricky. We need to devise a quick and easy method of making sure all images uploaded are appropriate and inoffensive for any younger members. We also need to have a method for members to report any problems they are having regarding other members abuse of the system. Placing an email link on each user profile, which wouldn't be too tricky, could do this.
- **Time**
As this project is the kind of thing that would continuously grow and evolve, we think that the deadline will only be there to demonstrate the project the way it is at that moment, and not for what it could grow to be as more people sign up and more features are added.
- **Server Memory**
Another problem we face is regarding the amount of storage space available to us through our server accounts. As the project grows, more and more images will be uploaded and this will eat up all of our space. This will limit the scale that we test the project on. Ideally we wouldn't be limited in this respect.

3

Functional Requirements

Section

3.1 Sign Up/Register

- **Description:**
This function is the first step the user takes on the way to becoming a member. A hyperlink is provided on the main page of the site which links to a registration form. This initial form will prompt the user for a few mandatory pieces of information. After completing this mandatory form, the user will be redirected to another form, where they will continue on the registration process.
- **Criticality:**
Obviously this function is essential to the whole community idea of our project. We feel it is not enough to merely have a community of casual visitors that we have no information about, and that once we have certain pieces of personal information about them, members will find it easier to approach each other and the project will have the community feel to it.
- **Technical Issues:**
The registration form itself will be designed in HTML and will fit in with the overall site layout. We will handle the form inputs themselves using PHP functions, which will access our MYSQL database in order to store user information.

Once completed, this will create an entry for this particular user in the users table. The users table contains a foreign key which links to the user_profile table.

- **Dependencies:**
None.

3.2 Create Profile

- **Description:**
After completing the initial registration form, the user will be redirected to this second form. With this form, the users can enter more information about themselves, thus, creating a user profile. The profiles themselves will contain information such as: Name, Age, Location, and Interests etc. All of the fields in this form are optional so to allow the user to maintain a degree of anonymity.
- **Criticality:**
The site relies on people finding other people with similar interests; so creating a detailed profile within the site is critical for being able to find similar people that might interest you. After finding a desirable profile, members can then make use of the other functions of the site in order to make contact.
- **Technical Issues:**
Once again, the form will be designed in HTML and handled with PHP and MYSQL. Once completed, this form creates an entry for the particular user in the user_profiles table. The user_profile table contains optional information about a user, which is associated with the users mandatory information through the foreign keys, so all user information can be easily referenced.
- **Dependencies:**
This area is dependant on the user having completed step 1 of the registration process.

3.³ Upload Images

- **Description:**
This function allows the user add images to his/her profile. At the moment we are unsure of whether to place this inside of function 3.². This can easily be implemented as a part of the "Create Profile" function, or individually as part 3 of the registration process. Basically, this will allow the user to browse folders on their machine and upload the selected images.

Once uploaded, the image will be posted for a moderator to approve/reject based on how appropriate the image is.

- **Criticality:**
This function is not essential but we think that it adds personality to the user profiles and is a feature of many similar websites.
- **Technical Issues:**
PHP has native functions suitable for implementing the file upload. We have not yet decided a method of ensuring adequate moderation for uploaded images.
- **Dependencies:**
Once again, this depends on the previous parts of registration being complete prior to upload.

3.⁴ Caption Images

- **Description:**
This function allows the user to add captions to the images they are about to upload. Once again we have yet to decide whether to make this function part of function 3.² along with 3.³. But this can be implemented along with the previous function.
- **Criticality:**
This is not important at all; it simply gives users a chance to let visitors know what's going on in a particular image.

3.⁵ Log In

- **Description:**
This is a simple user login script that prompts a user for their username and password. The user has entered this info during the registration process so we simply reference our database of users looking for the user name that's trying to log in.
- **Criticality:**
This function is vital to the project. Without this we have the problem of casual browsers abusing all functions of the system.
- **Technical Issues:**
This will require some security in order to stop unauthorised access to the system.
- **Dependencies:**
Depends on registration of the user name.

3.⁶ Edit Profile

- **Description:**
After creating their original profile on registration, it is possible that the user will choose to change some of the information we have on them. This “Edit Profile” function will be available to all users through a user menu, which is available after login. From here, the users public profile values can be edited and updated.
- **Criticality:**
Not critical but a nice feature to have added for increased functionality.
- **Technical Issues:**
Simple PHP script that will call the users profile information from the MYSQL database and present it to them to edit.
- **Dependencies:**
This is dependant on register and log in functions.

3.⁷ Add/Remove Friends

- **Description:**
Simple function that allows users to keep a list of people they meet on the site. This saves the user from having to remember different user names all the time by placing a link to each “friends” profile in their own profile. These will be visible to everyone visiting the profile.
- **Criticality:**
Not critical but relatively easy to implement and a very handy function for each user.
- **Technical Issues:**
None that are apparent at this time.
- **Dependencies:**
This is dependant on register and log in functions.

3.⁸ Manage/Delete Images

- **Description:**
This function gives users the option of adding (uploading) more images to their profile, as well as deleting images they have already uploaded. We may include an instance of “Caption” function here.
- **Criticality:**
Not at all critical but it definitely adds to the functionality and interactivity of the site. This allows the user to do whatever they want with their images.
- **Technical Issues:**
Like the above “Upload Images” function, this will require some moderation and this has yet to be fully thought out.
- **Dependencies:**
Dependant on register, create profile and log in.

3.⁹ Delete Profile

- **Description:**
Don't like the site anymore? This is for you. This is a function to delete all knowledge of your visit. This will remove all the tables for that user within the database.
- **Criticality:**
This is critical for the site admins. Because of space limitations we will have to free up space from time to time and this can be done for us by giving users the option of deleting their profile and images. This saves us from having to delete profiles ourselves.
- **Technical Issues:**
This function will make use of the SQL "delete" command. An additional feature could be a warning that is sent to the email address of a user telling them that their account has been inactive for some time and asking them to delete the profile themselves or risk having it deleted by an admin.
- **Dependencies:**
Dependant on register, create profile and log in.

3.¹⁰ Rate Profile

- **Description:**
Like the look/sound of a user? Tell us how much. You can rate a user profile, whether you think they sound cool from their interests and descriptions, or you simply like their photos. The rating system will consist of values 1-10, 1 being the lowest "Didn't like them at all" value and 10 is the highest "Ooooh I like them" value. Ratings will be viewable on all public profiles before and after people rate and will be updated immediately using MYSQL.
- **Criticality:**
Not critical but we thought it appropriate. From experience we have realised that these kinds of functions are popular on the Internet, and even within DCU. It adds a sense of fun to the site. People have been known to spend a lot of time on sites that offer this function, simply rating other users.
- **Technical Issues:**
The rating value will be a column in the MYSQL user_profile table. When a rating is received, the column value will be updated with the average rating and the number of ratings received. Each user profile will display that users average rating to others.
- **Dependencies:**
There are no dependencies. This function is available to all visitors to the site. This, we think, will encourage visitors to become members.

3.¹¹ Message User

- **Description:**
This function is the most difficult of the lot to implement. This function allows users to send and receive messages between each other from the user profiles. We have yet to decide how to implement this but we reckon we could store all info about every message in a database table. This table can then be queried and displayed in different formats in order to show different messages, i.e.) Inbox, Outbox (sent messages), and maybe even Saved Messages.

- **Criticality:**
This function will be critical to the operation of the website as it is the only way in which we allow each user to make contact with somebody new. This will also be a method of attracting people to sign up, rather than remain a casual visitor.
- **Technical Issues:**
We have a lot of issues to deal with regarding this function. First of all we need to devise a method of reliably sending and storing each message. After this is done we will need some scripts to query the database. These scripts will be the Inbox, Outbox and Saved Messages.
- **Dependencies:**
Register, Log In and Create Profile.

3.¹² Search Database

- **Description:**
This is a simple search utility available to everybody. It lets users search the database of active profiles. Users will search by various criteria, such as: Age, Location, and Interests etc. The results will be shown as a list of users matching the search criteria. Each user on the list will be linked to their own user profile so you can check them out.
- **Criticality:**
This is a critical feature of the site, otherwise users would have no way of finding other users to interact with.
- **Technical Issues:**
Simple PHP form querying the MYSQL database for data matching a certain search criteria.

3.¹³ Tour the Site

- **Description:**
This section of the site will be the first part an unregistered user will see. This will take the visitor on a short tour of the site, contrasting the different functions available to unregistered and registered users. It will outline the various benefits of membership as well as advising them on how to register as a user.
- **Criticality:**
This is definitely very critical to the success of the site. This function, along with the rate and search functions are the things we will use in order to get people to sign up. This will determine the success and popularity of the site.
- **Technical Issues:**
It will be completely designed with HTML, with some PHP/MYSQL to query the database and show some profiles maybe.

4

System Architecture

Section

4.1 System Architecture Diagram

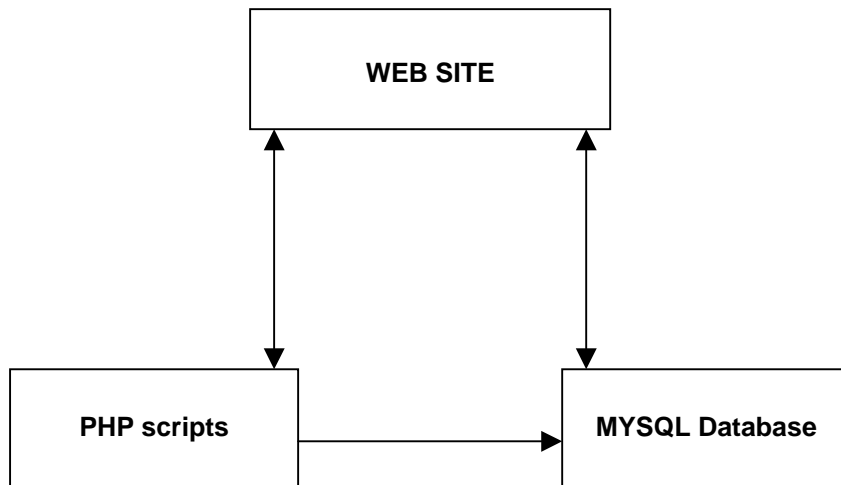


Fig 4.1

Fig 4.1 above illustrates the architecture of the product. As the above diagram shows there are three distinct aspects to the architecture. The first is the website (front end. What the users see), then the PHP scripts (which queries the database and returns the content for all pages) and finally the MYSQL database (which stores the site content).

4.2 Web Site

The website is the front end of this website. It is what the user will see and interact with to gain access to the other architectural elements of the product. This will be the least technical element of the product, as it is simply there so the customer can use the functions of the website in an aesthetically pleasing way.

4.3 PHP Scripts

This is the second part of the architecture and will be used in two ways. It will take information inputted on the website through HTML forms, and send this information to be stored in a database. The scripts will also be used to query the database and display the data returned for each page, formatted with HTML style. Therefore it is logical to say that this element of the architecture acts as a bridge between the other parts of the architecture, specifically for inserting and retrieving information to and from the database.

4.4 MYSQL Database

The database is the one part of the project that relates directly to each of the other two parts. It will store the information sent to it by the PHP scripts. This information will also serve as the content of the web pages. So what the user sees through the web site, is actually the information stored in the database, but being retrieved by the PHP scripts and formatted with HTML and CSS styles.

5.1 High-Level Design Diagram

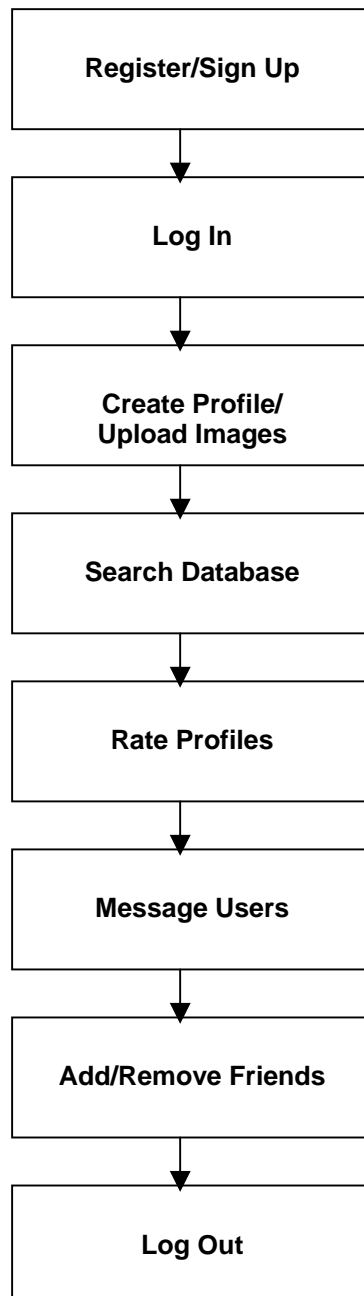


Fig 5.1

5.² High-Level Design Description

Fig 5.1 is explained below.

- **Step 1: Register/Sign Up**
Register a username and password to give you the ability to login to the site.
- **Step 2: Login**
Log In to the member's area using the username and password obtained above.
- **Step 3: Create Profile/Upload Images**
Input your personal details and perhaps upload an image. This information will be displayed on your public profile.
- **Step 4: Search Database**
Look through the profiles of various users using a search utility. After searching you can view the public profiles of the people returned from your search.
- **Step 5: Rate Profiles**
Give member profiles a rating, based on how much you like either their personal information or pictures.
- **Step 6: Message Users**
While browsing users profiles you can send them a textual message that will be stored, and viewable by them in their "Inbox". A copy of this message will also be kept in the senders "Outbox".
- **Step 7: Add/Remove Friends**
This is a collection of your friends with registered usernames. You can add and delete friends from this list as you wish.
- **Step 8: Log Out**
Once you have finished using the various features of the site, you can then log out.

6

Preliminary Schedule

Section

6.1 Overview of Preliminary Schedule

The schedule below was designed using Microsoft Project. **Fig 6.2** shows a full list of tasks and **Fig 6.3** shows the plan of how and when these tasks are to be completed.

So on the task list chart/table it shows the task name, duration, start and finish dates and with the team members who are going to complete the specific task.

On the Gantt chart is a more visual display of the information contained in the task list. It shows each task a bar or line, and its relation to previous tasks, i.e. task A may or may not have to be completed before task B can be completed.

6.2 Task List







		Task Name	Duration	Start	Finish	Predecessors	Resource Names
1		Submit proposal	0 days	Fri 19/11/04	Fri 19/11/04		
2		Functional Specification	4 days	Mon 10/01/05	Fri 14/01/05		Colin O'Driscoll, Patri
3		Submit Functional Spec	0 days	Tue 21/12/04	Tue 21/12/04		
4		Design of website	5 days	Mon 31/01/05	Mon 07/02/05	2	Colin O'Driscoll, Patri
5		user sign up	4 days	Tue 08/02/05	Mon 14/02/05	4	Patrick O'Toole
6			5 days	Tue 08/02/05	Tue 15/02/05	4	Colin O'Driscoll, Patri
7		image upload	2 days	Wed 16/02/05	Fri 18/02/05	6	Colin O'Driscoll
8		implement database	3 days	Fri 18/02/05	Wed 23/02/05	7	Colin O'Driscoll, Patri
9		query database	6 days	Thu 24/02/05	Mon 07/03/05	8	Colin O'Driscoll, Patri
10		messaging function	5 days	Mon 07/03/05	Tue 15/03/05	9	Colin O'Driscoll, Patri
11		full run of project	3 days	Tue 15/03/05	Mon 21/03/05	10	Colin O'Driscoll, Patri
12		write manual	2 days	Mon 21/03/05	Wed 23/03/05	11	Colin O'Driscoll, Patri
13		demonstration of project	0 days	Mon 28/03/05	Mon 28/03/05		Colin O'Driscoll, Patri

Fig 6.2

6.3 Gantt Chart

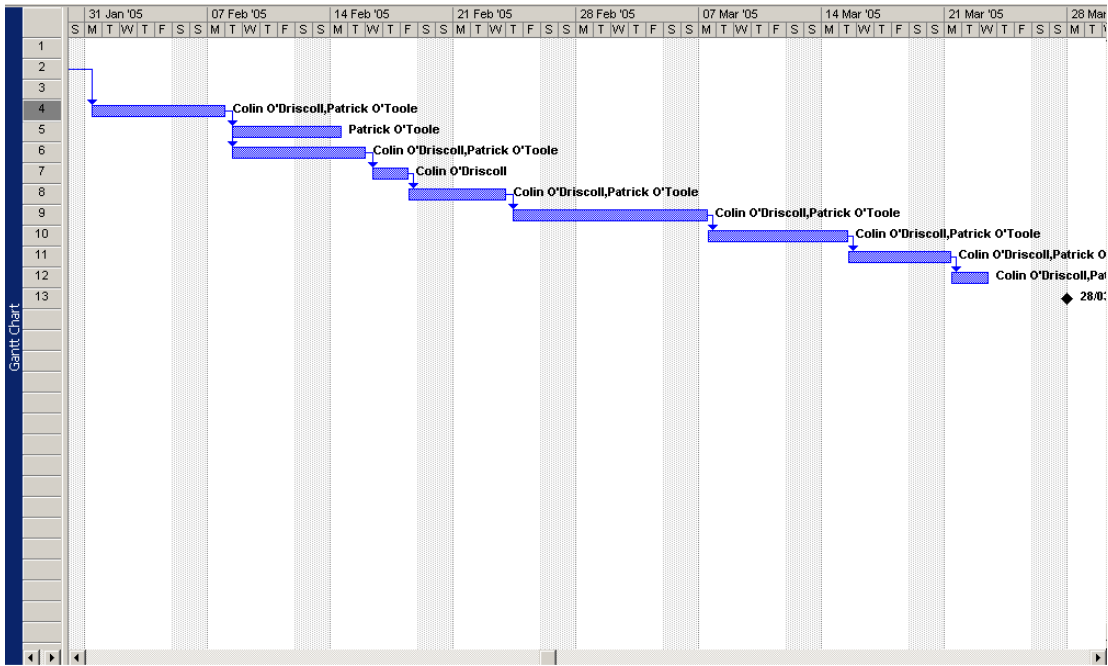


Fig 6.3

7.1 Appendix 1 – Resources

- **Similar Sites:**
 - <http://www.hotornot.com>
 - <http://www.okcupid.com>
 - <http://www.faceparty.com>

- **Research Tools:**
 - <http://www.php.net>
 - <http://www.w3cschools.com>
 - <http://www.devshed.com>
 - <http://www.sqlcourse.com>
 - <http://www.mysql.com>
 - <http://www.webmasterworld.com>
 - <http://www.redbrick.dcu.ie>
 - <http://www.computing.dcu.ie/~roconnor>

7.2 Appendix 2 – References

- <http://www.google.com>
- <http://www.webopaedia.com>