Peer-to-Peer Networks and their Application as Distributed Social Networks

by

Conor McEvoy, B.A (Mod)(Hons)

June 23, 2013

Submitted to Graduate Faculty of
Department of Computer Science
of the requirements for the degree of
Master in Computer Science (MCS)

Declaration

I declare that this dissertation has not been submitted as an exercise for a degree at this or any other university and it is entirely my own work. I agree to deposit this dissertation in the Universitys open access institutional repository or allow the library to do so on my behalf, subject to Irish Copyright Legislation and Trinity College Library conditions of use and acknowledgement.

Signature

Date

Abstract

In social networking, companies such as Facebook, Google and LinkedIn dominate the social networking space leaving little room for alternative models. Distributed social networks are slowly emerging following recent developments in distributed networking technologies, computing processing power, and developments in networking. Distributed social networks are developing into a new social model for users around the world to communicate without using the traditional client/server model.

Distributed social networks aim to remove data mining that companies perform on each user and guarantee privacy to each user of the network. This dissertation aims to examine distributed social networks, to examine their current and future use, and to propose a new model to networking. A secondary aim is to evaluate whether a distributed social network would cause a paradigm shift, from a client/server model to a peer-to-peer (P2P) approach to social networking. The dissertation will also detail a P2P model to research such social network interaction and to provide relevant research into the distributed social networking domain.

Finally the dissertation will research current user trends and opinions towards current social networks, and where a distributed social network may offer current users alternatives to today's centralized social networking models.

Attestation

I understand the nature of plagiarism, and I am aware of the Universitys policy on this. I certify that this dissertation reports original work by me during my University project.

I, the undersigned, declare that this work has not previously been submitted as an exercise for a degree at this, or any other University, and that unless otherwise stated, is my own work

Signature

Date

Acknowledgements

Firstly, I would like to thank my Supervisor **Dr. Hitesh Tewari** for his continued advice and guidance throughout this study.

A special thanks to the Department of Computer Science in Trinity College Dublin for providing me with the opportunity, facilities and support in continuing my studies and research over the years.

A final thanks to my parents, my sister and my grandfather Tom Cox, who have inspired me throughout my years to develop as an individual and pursue my career in Computer Science. This study would have not been possible without all of you.

Conor McEvoy

Contents

List of Figures

Chapter 1

Introduction

Today we live in a world of ever evolving technologies and electrical marvels, which drives industry, corporations and businesses alike. Computers and mobile devices have never been more functional, practical and mobile with new developments in technology and advancements in processors, memory, network bandwidth and accessibility year on year. Rapid strives in technology have allowed the world and many distant cultures to more connected then ever with newer, faster mobile nodes and smart phones emerging each day which each act as a digital source of information databases, knowledge and opinions. Without these three, the Internet would seem dull and boring and would certainly not provide as much depth and interest as it does worldwide today.

Social networking provides a solution to global communications, it provides a space for friends to be reunited, it allows a user to create a digital image of themselves and their personality, and it provides an online space that contains all of the social elements such as humour, interaction and event planning. With this however there are always pitfalls to such openly sharing data and social networks have recently become under scrutiny with regards to user privacy and data protection laws. A distributed social networking approach on the other hand would allow users to bypass the third party platform, keeps users information secure and allow the easy sharing of current models used by social networking platforms today.

We can define three models to social networking applications:

- **Closed Social Network** - This model describes the social network to one that is hosted and run by a third party. The application is designed, developed and deployed to which users use their services to establish social connections and share media. An example of this would be the Facebook [Zuckerberg, 2004] and Bebo [Birch, 2005] model.

- **Peer-to-Peer Approach** - A P2P model would exclude any third party communicator and would instead allow multiple devices to connect and exchange social media from their own devices. An example of this is the popular file sharing network BitTorrent [Cohen and Navin, 2004].

- **Abandoning of Technology in Social Communications** - This approach is unlikely given the circumstances of the current push of social media and online social traits emerging. One must consider a world whereby users revert back to the preconditions of the social media revolution. This would indeed spur its own unique social paradigm as it poses a scenario that even though we have these great interconnected networks, lower latencys, and much greater bandwidth, that users still tend to stay away from social networks and the use of technology for social media.

My research that I wish to provide during the course of my dissertation is to answer the question and propose solutions into the popularity of current social media distribution methods and how a distributed model potentially holds the answer to current issues with social media and the closed social networks.

1.1 Background and Context

In today's world, social networking is dominated by closed, single service, centralised servers for providing users with all of their social networking needs. These services are developed to be rich, interactive and seamless in connecting users, friends, colleagues and family together over networks.

Figure 1.1: Today's popular social networks

In the world of social networking, networking monopolies dominate users and leave little room for competition and advancements in the field. This is due to a number of factors such as hardware requirements and user adoption rate which will be discussed at a later stage in the dissertation report. Distributed approaches have proven to be resilient in the past with information and file sharing approaches such as BitTorrent and Napster [Parker, 1999]. With lower latency and greater bandwidth in connections, year on year it is becoming easier and faster to connect users over the network to each other around the world. This is where there is a growing requirement for research and development in technologies that were once impossible due to issues with Big Data and network barriers. We live in a world where network barriers are being broken and data operations that once required specialized hardware or complex software can now be done on our regular machines.

With all of the modern developments in networking and communication technologies, I wanted to research and develop into the emerging space of distributed big data problems with providing an alternative to one of the biggest uses in the Internet today, online social networking.

1.2 Scope and Objectives

The topic of this dissertation covers a board range of subjects and areas with regards to social networking, network design and implementation, data protection, and topics of user interaction and opinion. With such a broad category of study and research this dissertation take a brief analysis of the surrounding topics and will provide an in-depth analysis of how a distributed social network model can solve current technical and social problems in today's world.

Objectives that I set myself for the duration of my dissertation included the following:

- **An Analysis of Social Networking Problems** For a new social networking model to emerge it must hold potential to cause a paradigm shift or change from the current model to the new model. The research carried out throughout this paper aims to show that there is a demand for a secure P2P approach and that a P2P approach would be a viable replacement to current popular social network technologies.

- **How Peer to Peer Approaches Can Solve Big Data Social Networking Problems** Social networks create a lot of big data problems through their complex analysis of gathered information from social network activity. A lot of this data is analysed for marketing purposes and to evaluate user trends to pinpoint areas to improve the quality of service provided to each user. P2P networks will create big data across the network but this data will be shared amongst each user. There are many advantages and disadvantages to this approach that this paper aims to examine.

- **The Potential Application of a Distributed Social Networking Model** A valid P2P model designed specifically for distributing social media may have many other applications then just distributing content in the form of a P2P social network.

1.3 Dissertation Outline

This dissertation will aim to define and describe a new distributed social networking model. The dissertation report will first begin by describing the need for developments such as this paper and the problems in the world today that are leading the desire for advancements in the field of distributed networking. The paper will then move to evaluate current research topics and models to show how the world is slowly adapting to a distributed approach. The paper will define and describe current P2P technologies in the light of describing Microsoft's Peer Name Resolution Protocol (PNRP) [Microsoft, 2007] which is the driving technology for interconnection of users across the described network. The paper will continue by explaining the need for distributed social networks and the statistics and opinions gathered from users which provide direct feedback into the desire for the proposed model and solution of the dissertation. A description of the solution researched and provided will be given to the reader along with a breakdown of the interconnected components and features. Finally, the dissertation will conclude with a summary of the research completed, an evaluation of the technology and proposition and where work future work could be developed.

1.4 Overview of Dissertation

Throughout the course of this dissertation, we have looked at P2P networking models. We examined how distributed models play a vital role in networking today and the potential future in the development of P2P technologies. The paper examined the state and methodology of online social networks today, showing potential for a paradigm shift to P2P social networks.

To show a potential future paradigm shift to distributed social networks, a survey was conducted with engineers and computer science students enrolled under the Department of Engineering in Trinity College Dublin. The survey was designed to evaluate opinions about current social networking models and to examine future potential for distributed social networks. The survey results found that users currently have issues with regards to data security, data-mining, advertising, and data analysis that a distributed model could potentially solve.

A proposed model for a new P2P social networking model is proposed. The model implements networking, database storage, and user interface components which combined, create a distributed social networking application. The application known as "Peerple" is a working social networking application which provides each user an account, profile, social networking features, and network security.

The dissertation showed that there is viable demand for distributed social networking in the near future. Further research is needed to be carried about the scalability of P2P applications and the development of a final client for the social network.

Chapter 2

State of The Art

The field of social networking is one that every year is reaching more and more people and is leading to a massive boom in both big data and social connectivity over the Internet. Companies like Google, Facebook, and LinkedIn, focus and invest hundreds of millions of dollars maintaining and developing their networks each year, to strive to achieve the most user friendly network with the most active connections. The approach this dissertation aims to research is that of a distributed social network. A distributed social network differs from the likes from Facebook and Google+ due to its P2P and distributed approach to sending and receiving data. Many technologies have been invented and researched into data distribution technologies but these technologies have not entirely focused with the rise of social networking in mind. In this survey we will focus on the technologies that are used to developed P2P connections, create media rich user interfaces and provide seamless data transfer and maintenance throughout the network.

2.1 Peer-to-Peer Technologies

In this section we will cover current P2P technologies and the relevant research required to adapting these networks to an approach for a distributed social network. We will study improvements in this field in both popularity and as a means of successfully distributing media. We will also observe how bandwidth and improvements in networks have allowed for the emergence of reliable and

Figure 2.1: The battle for social network dominance

fast P2P network.

2.1.1 Developments in Networking & Computing Technologies

The development of computing can be defined in a number of ways such as the development of hardware, software, networking capabilities, popularity and many more. Computers have become much more then previously envisioned with them now being contained in nearly every electric device that we use today. Computers are present in our mobile phones, games consoles, smart devices, and even appliances, and they continue to provide each user a faster and more connected experience. Moore's law is known as one of the most famous technological findings to which the capabilities of many digital devices have stemmed from. Moore's law states that "That over the history of computing hardware, the number of transistors on integrated circuits doubles approximately every two years." [Moore, 1970]. Moore's observation has remained true up to this day, and computers are continuing to develop and double in speed and increased power efficiency every 18-24 months. This is one of the overall contributing factors to the development of computers, rich interactive media, networks, and have provided a computer in the hands and pockets of each user who owns a smart phone.

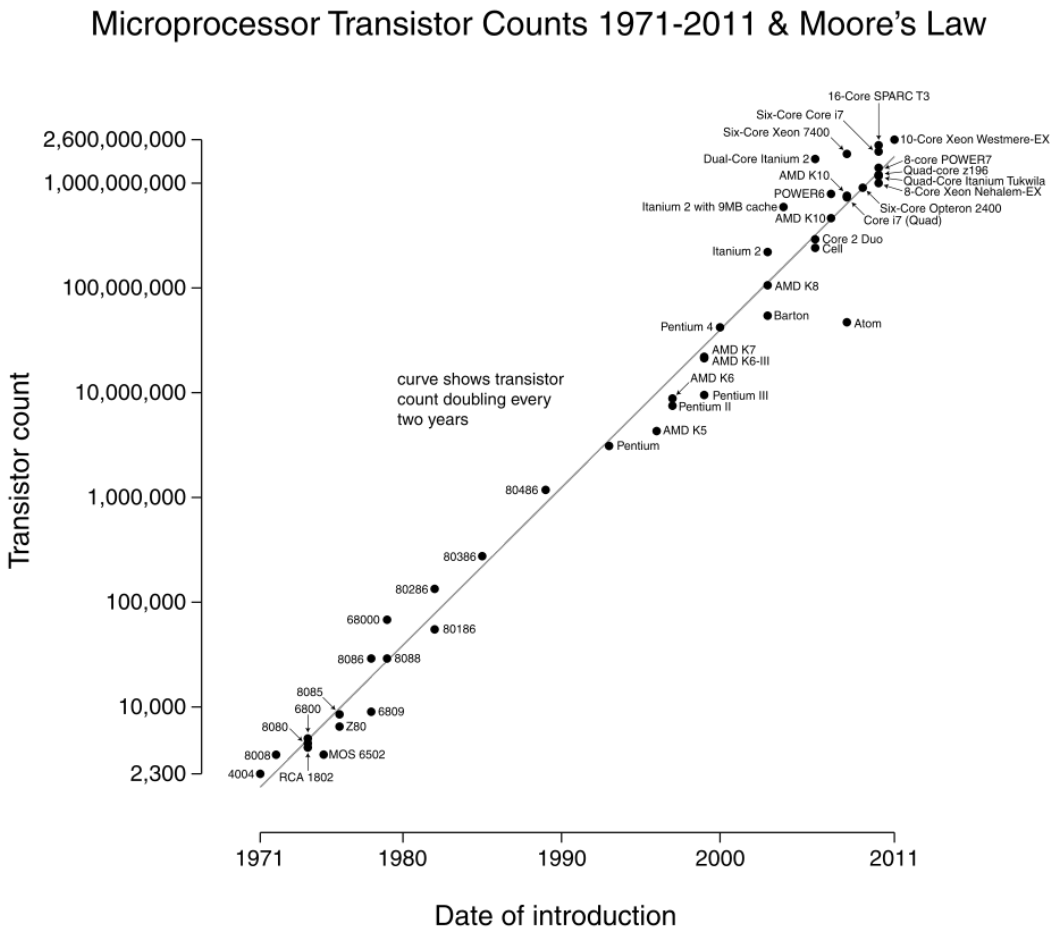

Figure 2.2: Moore's Law

Developments in computing is important to the development of social networking models today because we simply would not have enough processing power to provide and maintain big data storage and management as well as hosting services. Today the likes of Facebook and Google are constantly investing in the future of computing technologies to provide more processing power with less heat and power for their network services. With processing power on personal computing increasing approximately every 18 months, our computers in our phones, towers and laptops are becoming as powerful as heavily invested computing technology only a few years ago. As processing power and

capabilities continue to increase, our computers will be capable of much more then they are today. The increase in computing is leading to the viability of single computers being able to compute big data solutions on the P2P scale and conforming to providing a distributed service.

The development of networks will lead to a more interconnected and mobile world. Networks which were once confined solely to wires and low bandwidth connections have developed to the point where even our mobile phones are capable to megabit transfer speeds over mobile networks and all with greatly reduced latency as found in previous generations. The development in networks is allowing more and more data to be transferred quicker, more reliably and in greater bulk allowing rich media interconnections between nodes on the network.

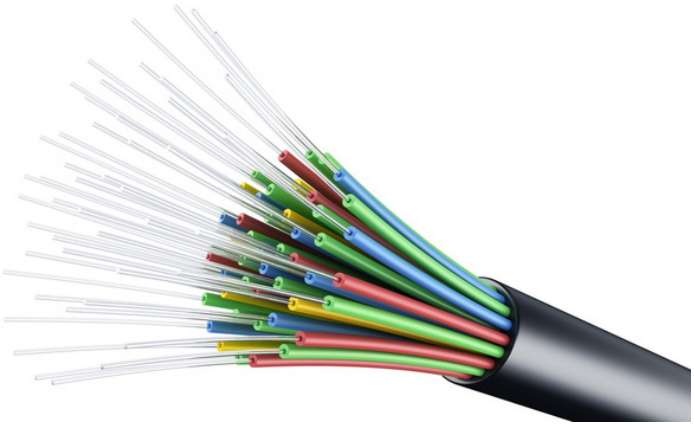

Figure 2.3: Fiber Networks are said to be the future of network connections

Developments in our networks and networking technologies are important to a distributed social network, as the network must be fast and reliable to provide the ease of use of the service and to guarantee the sending and receiving of messages. Fast interconnected networks will be able to sustain more P2P connections, handle more social media traffic and provide better overall connectivity.

Future developments - Future developments of computers and networks are going to continue to make our computers more powerful and our network

connections more reliable and seamless. As of writing this report, Intel is developing faster 22nm processors with future plans to develop 15nm, 10nm and 5nm processing technologies [Intel, 2013]. This will lead to much faster, more energy efficient computing solutions and will soon make our computers of today obsolete. It will open the door for our everyday computing devices to be able to host and process rich media, and in combination with advances in networking, will continue to make our everyday computers more viable as powerful P2P nodes.

2.1.2 Peer to Peer Network Technologies

P2P network technologies [Schollmeier, 2002] allow a computer to act as both as a client or a server to provide a service between computers such as file sharing, sensor data, network data, all without the need for a central server. This eliminates a single point of failure in the network and guarantees that every node has a copy of the data in the case of being disconnected from the network. Each computer is represented as a node in the network and each user of the node is called a peer. P2P networks can be set up at home, at a business, or over the whole internet and can be used to share any digital data content.

Peers are both suppliers and consumers of data on the network in contract to the typical client-server model where by multiple clients send data to a single server and the server replies to each client separately. A group of peers operating on a network is commonly referred to as a mesh [Rouse, 2006]. A mesh may be defined to providing a specific service, in the case of my research project, the mesh is the interconnection of social nodes on the network whereby data is transferred to each node in the network. Meshes may be configured to dedicate tasks to separate nodes, or to conform data aggregated from a group of nodes to a single source. Meshes can be complex, consisting of meshes within meshes, but this is all application dependant.

P2P systems often implement an abstract overlay network built at the application layer. This overlay network is used to implement peer discovery algorithms and data transmission between peers on the network. Content is mainly transferred through the use of Internet Protocol (IP) to identify devices on the

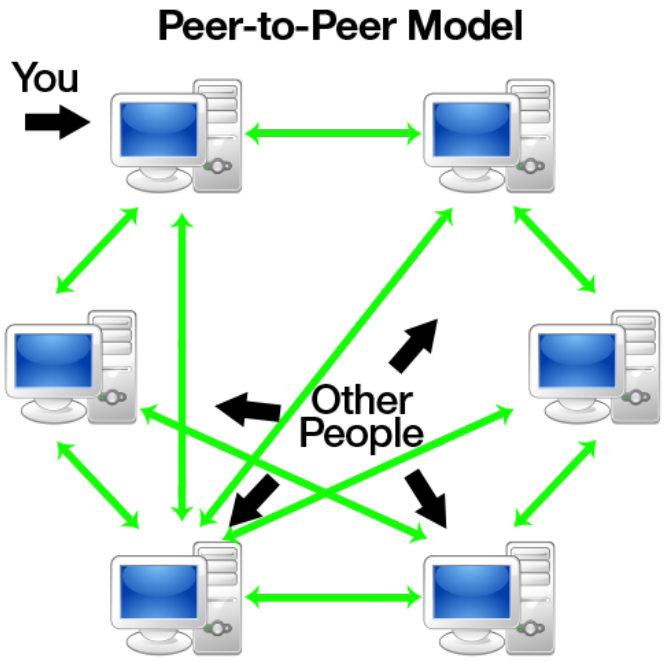

Figure 2.4: A peer-to-peer networking model

network and forward data packets to. A pure P2P network does not involve the notion of clients and servers but instead just peers. Pure P2P networks are resilient to connection outages by design as they simply communicate with all other nodes in the network if a single connection goes down.

Distributed approaches to social network are currently a hot topic in the field of computer science due to the emergence of a data retention problem in current social network services and the increase in both computer processing and network performance. One can observe a broad range of technologies when using P2P based systems. Several hybrid approaches(e.g Diaspora [Team, 2012]) use dedicated trusted nodes to address availability, security, reliability, and maintenance of network content and distribution. In the centralized case, such as Facebook or Google+, all communications and service interactions occur by the use of a central provided instead of a P2P network approach.

There are numerous approaches to implementation but all models attempt to satisfy the following:

- **Content Hosting** - A user hosts all the content they post. To ensure availability even when when they are offline, their content is replicated to a number of storage nodes.

- **Content Access** - The user designates restriction on the content. The content can only be viewed or in this case of security, decrypted, by a key shared among members of that particular network. Authorized nodes or other users can only access the content through decryption.

- **Connection Handling** - Two types of service, a central provider lookup, or a pure P2P approach, could be implemented to provide distributed functionality. A central provider lookup is a central service which accepts connections from distributed nodes and provides authorisation and the contact addresses of the other nodes on the network. A pure P2P approach removes the central lookup and instead relies on distributed storage and authentication nodes to provide consistency, content distribution and security throughout the network.

P2P networks can be categorised as structured or unstructured. Structured networks are once which follow strict networking topologies to conform to design and specification. Structured networks generally use a distributed hash table (DHT) approach to index the location of nodes such as in the Chord system [Druschel, 2010] developed by MIT. Unstructured networks do not impose any structure on the design of the network. Peers in these systems connect in an ad-hoc way and connections are based on a loose set of rules and configurations.

Each design to P2P networks has it's advantages and disadvantages. Deciding on which model and approach to use is application specific and will greatly effect how the network operates and preforms. From our research into P2P social networks and for the model of distributed social network that I wish to research, I choose to build my model on an unstructured distributed P2P network. For reasons that will be explained later in the paper, an unstructured P2P network provides allows the ease of connecting and joining peers in the network, it provides a lower complexity of network design and implementation

making networks simple to establish and maintain, and it provides the capabilities of each node sending and receiving social information through a connected mesh.

2.1.3 Resilience of Distributed Networking Approaches

P2P networks, by design, are fault tolerant networks. Each node in a network is connected to each other node and thus there is no single point of failure. Each node is not solely dependant on the nearest node, as it may simply route it's data through another connected node to reach its destination. Node routing and node availability are controlled by complex peer discovery algorithms which provide each node the ability to find any other node on the network, and once a connection is established, send any data across the network from node to node. P2P networks are resilient because a node can have access to every single node on the network through only one node which is the perfect design strategy in fragile or limited access networks.

In P2P networks there is no need for a system administrator or data coordinator. Each user is their own network administrator, and each administrator can choose what content to distribute to other peers in the mesh. This removes complexity of management of the network and allows each node on the network to be equal with regards to permissions, privileges and content.

There is no cost to creating and maintaining a P2P network. There is no central hardware to manage or maintain for the network to remain operational. Any user with a device that has networking capabilities can essentially be a peer in a mesh. There are no subscription fees, bandwidth limitations, greatly fluctuating latency issues in a P2P network, subject to the users actual connection to the neighbouring peers.

All of the content in a P2P is shared amongst each node in the network. This means that once content is consistent across nodes, that nodes can provide any content on request or be a reliable source of content in the case of a failure.

2.1.4 Peer-to-peer Social Networks

Distributed networks in themselves have been a popular research topic in recent years with increased processing power and network capabilities improving personal computers as potential to be efficient nodes in node meshes. Distributed networks and the idea and concepts of distributed social media are not new to the technology environment. Applications such as the infamous Napster brought about a distributed file and information sharing revolution in the early 2000s and introduced new thought patterns and functionality for sharing media across the Internet. Napster of course stored the references to its content sharing media centrally which led to its demise in the end due to copyright infringement but new approaches to distributed models such as the more modern BitTorrent network allow the direct connection of multiple peers worldwide for the purpose of information and data sharing.

There are plenty of examples of distributed social networks that can be divided into two categories by design: Web Hosting Approach, peer-to-peer/Distributed Hashtable(DHT) approach. A distributed social network page on Gitorious [Gitorious, 2013], lists the total number of projects and their active state. This dissertation will not list and detail each of the projects, but instead list the fact that all of the active projects on distributed social networking are currently in research or development stages and have not published technical documents or complete specifications of there projects.This dissertation would have liked to cover each aspect of current distributed social networking models, but the lack of information and research papers describing the models in development would have led to inconclusive results in comparison with the model that this paper describes. That said, Diaspora is one of the successful implementations of a distributed social network based on the Web Hosting Approach.

Diaspora - Diaspora is a free personal web server that implements a distributed social networking service. The project was founded by four students at New York University Courant Institute of Mathematical Sciences. The project was founded via a Kickstarter project to which they raised $200,000 which led to the release of an alpha version on November 23, 2010.

Diaspora allows users to set up their own server to host social networking content, servers can then interact to share status updates, photographs and other social data. The framework is being built on Ruby on Rails. The creators intention was to allow users to communicate directly, securely, and without running exchanges past social networking companies.

The highest user count in one pod is currently 213,070 users and increasing by the day, which shows that there are plenty of users who prefer a distributed social network.

2.2 Towards Distributed Social Networks

This section aims to examine the popular and successful social networking applications today. It looks at why social networks are successful, how they attract users and resources, and how current trends in popular social networks may attract users to distributed social networking solutions in the near future.

2.2.1 Social Network Monopolies

The goals of companies such as Facebook are purely about gaining market value and profit through revenue streams such as targeted advertising, data collection or data mining, and the ability to charge users for extra features and functionality of the service [O'Neill, 2010]. Social networking companies therefore strive to attract the largest user base and social interest by tailoring their service around social trends and achieving an almost online social monopoly. Having a social monopoly gives the network a strong lifespan and position in the market as the goal of online social connections is to connect as many friends and contacts as possible, and with services like Facebook hosting over one billion members, users are sure to know what service to use to find the largest possible reach and connections of friends in their online network. Users must be aware that even though such a service appears free to use and interact with almost transparently, there must be a generated revenue stream to power

and fund endless racks of services and data connections powering the social media platform. Without their comments, opinions, views, and wall-posts, the network simply could not fund itself and would evidently lead to an unpopular subscription model or the decoupling of the social connections themselves. Facebook spent over $50 million in 2010 on Data Centre leases alone and this does not include the costs of any of the management and development of the software behind these developments [Miller, 2010] .

2.2.2 Current Issues with Data Privacy

The Data Sharing Problem - Upon joining a social media platform the user must agree to the platforms terms and conditions. Hidden in these terms it is dictated that any information submitted to the social media site be it photos, videos, comments or links, is property of that network and the network remains the right to use this data in whatever means necessary. That said Facebook does state that if a user finds finds or discovers that their data has been in anyway manipulated, edited, or malformed then they are entitled to an investigation. This means that any information that digitally defines you and any description that you add to your digital profile is stored under the hood of the company and is open to manipulation and interpretation. This on its own poses digital identity risks for the user and leaves their lives exposed to the larger corporations examining their social habits and interactions to draw conclusions on the user and apply targeted media. In 2010 the StarTribune examines how human resource companies can potentially be using social media as another layer to the hiring process to examine a user before or after interviews based on their social presence and opinion [Phelps, 2010]. They argue that current guidelines for data protection and what is defined by companies such as Facebook allow the near transparent sharing of data, with its ease of use and user friendly interactions. However, social networking sites like Facebook gives users the impression that they are in control of their data by giving them the ability to delete posts and features that they upload onto the site. In fact, it is not possible to completely erase all personal information from the site as it may be stored in caches, backup disks or in the possession of advertising companies.

Developments in mobile technology have added functionality and ease of access to each member on the go and users can now stream their social networking information to them on the go or indeed share whenever they please. This additional model is directly related to both strives in mobile technology and the ties in with the companies model of being able to gather and collect data from each user wherever they are.

The problem with data mining is that each aspect of collecting data about a particular user and reselling that collected data is a hot topic in individual privacy and data protection rights. Once a user posts information on the Internet or through social media, it can be very hard to delete, as it is usually automatically cached or stored and users must be aware of the significance of the information that they are posting. This data could be indeed harmless, contain no relevant information and seem unimportant to the user but regardless the date posted, location posted, knowledge gained from the type of information shared and means of the post can all be valuable data mining knowledge to be analysed. So knowing this then why do people seamlessly upload embarrassing photos of themselves and their friends, leave their profiles open to direct interpretation from future employers or government officials or spend an unhealthy amount of time and effort in portraying themselves online?

Targeted Marketing and Advertisements - By implementing a distributed messaging construct model, there would be no means of a central server or any user snooping on the line to determine potential ads or pages to try and force the user to view. The P2P clients would not by choice support the display of ads and would thus free up space on the main pages for the ability to display more meaningful content or extra informative user interface elements. No one service has the sole access to data on the network and encryption methods can ensure that data transmitted over the network from a particular IP can be interpreted.

2.2.3 Why Are Distributed Social Networks Not Successful?

Distributed social networks must provide the old and existing users new functionality, features and the advantages must be greater then the use of the centralized social platform system for them to succeed and take interest in today's world. Unfortunately distributed systems face many hurdles in overcoming and overpowering the social networking monopolies today.

Hardware Requirements - A distributed computer or device must be discoverable on the network even with the change of IP address. Simply this means there must be a method of sharing an IP address or some form of unique identifier for each user that can cover a broad range of devices as to identify the user. Ideally a user should be able to move to any computer and have his user-name and password and be able to log into the network. This poses a challenge as IP can change and if a user logs in from a new computer he/she will have to download or stream all the current social network information from the nearest peer, process the information and redisplay that to the user. This may be a timely and costly event and would have to involve multiple security steps to confirm that the user is actually who they say they are.

User Adoption - User adoption prevents a huge problem to any potential social networking start-ups. Users will join sites and services that their friends and family are already members and understand that the more scope and users on a particular service, then the more that the service is going to benefit their experience and overall impression of using the service. Users are hardly motivated to defer from experiences that they currently enjoy and are fitting and fulfilling their current needs. The issue of privacy and data protection will have to reach crisis point before users will consider an alternative approach, which focuses on data protection and the privacy protection of their data.

Previous Attempts at Distributed Social Networks - The idea to distributed social media and create distributed social networks is one that has been around for many years. There have been several attempts to successfully

create new and distinct approaches to social networks and today these projects still go unheard of. During my research, we came across several different concepts and implementations of distributed social networking each more different from the last with one of the successes being the Diaspora project [Team, 2012] which has attracted over 375,000 users so far and has received investment from Mark Zuckerberg [Zuckerberg, 2012], founder of Facebook. Diaspora takes the distributed approach to social networking and is open source. Many other projects including Peerson [Buchegger, 2011], DiSo [Messina, 2010] and OneSocialWeb [Team, 2010] are actively studying and developing their project to produce a competitive project or solution to social networking monopolies. This highlights that the area surrounding distributed social networking is still one that is currently researched, prototyped and tested and I hope to see more interesting and adventurous projects in the near future.

Developments Towards Distributed Approach - Several projects mentioned before are slowly developing the tools and approaches for the complete distributed approach to a more open and more social internet. Open standards such as OAuth authorization [Cook, 2006], OpenID authentication [Fitzpatrick, 2005], OStatus federation [StatusNet, 2012], and OpenSocial widget APIs [Weitzel, 2012] are often cited as modern enabling technologies for distributed social networking. They each enable the open and distributed sharing of information in a secure and reliable manner. They are all modern and recent developments which show the distributed research field is still alive and ever developing. One would hope that it is only a matter of time before true innovation takes hold of the distributed approach and a working model or concept will be worthy of providing a paradigm shift to the new model. When it does the likes of Facebook and Bebo will have counter arguments against the networks and their competition. The reality is that if the data sharing analysis and sharing problem escalates, then social networking as we know it today may have a crisis on its hands and the opportunity for successful distributed social networks will have an opportunity. At the moment though, human nature dictates that users will stick what is known to be reliable, functional and provide the service they require and thus is the large mountain for any competition to

climb.

Chapter 3

Peer Name Resolution Protocol

3.1 Microsoft Peer Name Resolution Protocol

The following details and documentation have been provided by Microsoft under its Open Specifications documentation for protocols, file formats, languages and standards. The following section is adopted from technical details published by Microsoft regarding PNRP [Microsoft, 2004a].

The Peer Name Resolution Protocol (PNRP) Version 4 is a protocol that is used for resolving a name to a set of information, such as IP addresses. This protocol is used to maintain a network of nodes (referred to as a cloud) and to resolve names to their endpoint information when requested by a node within the cloud.

PNRP uses messages to maintain a cloud of peer nodes, to maintain a distributed cache of network endpoint information, and to transfer requests for Peer Name resolutions between nodes. Together these messages allow applications to use registered Peer Names to obtain corresponding endpoint information such as IP addresses and ports. When an IP and Port is obtained, users may bind these IP and endpoint addresses to open a network endpoint to establish communications between each peer in the cloud.

There are two primary roles in PNRP:

- **Resolver:** A node seeking to obtain endpoint information for a given

Peer Name by sending (and, when appropriate, resending) resolution requests to other nodes within a cloud

- **Publisher:** A node that provides endpoint information to a Resolver

3.2 Peer Names and PNRP IDs

A Peer Name represents an endpoint for communication which can be a computer, a user, a group, a service or anything associated with a Peer that can be resolved to an IPv6 address. The PNRP Protocol, takes the Peer Name for the creation of a PNRP ID, which is used to identify cloud members.
Peer names can be registered as unsecured or secured. Unsecured names are just text strings that are subject to spoofing, as anyone can register a duplicate unsecured name, Unsecured names are best used in private or protected networks. Secured names are protected with a password, or certificate.
The combination of cloud and scope provides a reasonably secure environment for peers that particpate in PNRP. However, using a secured peer name does not ensure the overall security of the networking application. Security of the application is implementation-dependent and may be implemented through the use of PeerChannel (discussed below).
PNRP IDs are created from the combination of the Peer Name and Service Location.

The high-order 128bits, known as the P2P ID, are a hash of a peer name assigned to the endpoint. The peer name has the following format: Authority.Classifer. For secured names, Authority is the Secure Hash Algorithm (SHA1) hash of the public key of the peer name in hexadecimal characters. For unsecured names, the Authority is the single character 'O'. Classifier is a string that identifies the application. The low-order 128 bits are used for the Service Location, which is a generated number that identifies different instances of the same P2P ID in the same cloud.

The combination of P2P ID and Service Location allows multiple PNRP IDs, to be registered from a single computer.

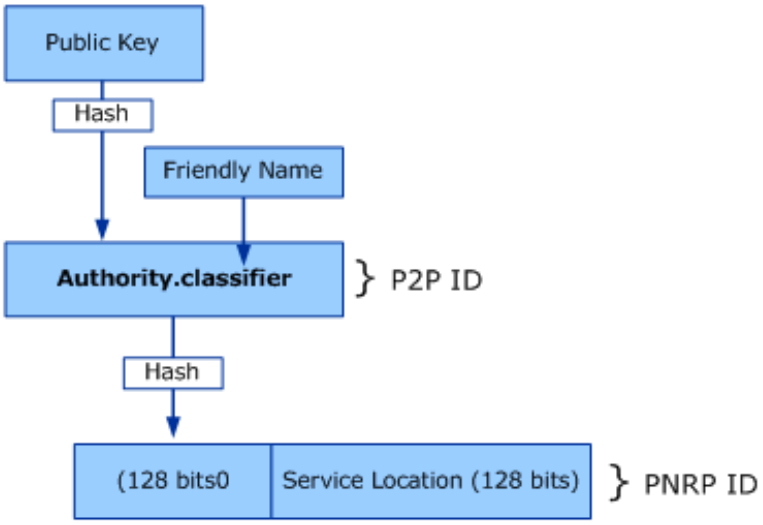

Figure 3.1: Construction of a PNRP ID

3.3 Resolving a Peer Name

Locating other peers in a PNRP network of cloud is a process comprised of two phases:

- **Endpoint Determination**
- **PNRP ID Resolution**

In the endpoint determination phase, a peer that is attempting to resolve the PNRP ID of a service on another computer determines the IPv6 address of that remote peer. The remote peer is the one that published, or is associated with, the PNRP ID of the computer or service.
After confirming that the remote endpoint has been registered into the PNRP cloud, the requesting peer in the PNRP ID resolution phase sends a request to that peer endpoint for the PNRP ID of the desired service. The endpoint sends a reply confirming the PNRP ID of the service, a comment, and up to 4 kilobytes of additional information that the requesting peer can use for future communication. For example, if the desired endpoint is a gaming server, the additional peer name record data can contain information about the game, the level of play, and the current number of players.

In the endpoint determination phase, PNRP uses an iterative process for locating the node that published the PNRP ID, in which the node performing the resolution is responsible for contacting nodes that are successively closer to the target PNRP ID.
To perform name resolution in PNRP, the peer examines the entries in its own cache for an entry that matches the target PNRP ID. If found, the peer sends a PNRP Request message to the peer and waits for a response. If an entry for the PNRP ID is not found, the peer sends a PNRP Request message to the peer that corresponds to the entry that has a PNRP ID that most closely matches the target PNRP ID. The node that receives the PNRP Request message examines its own cache and does the following:

- if the PNRP ID is found, the requested endpoint peer replies directly to the requesting peer.
- If the PNRP ID is not found and a PNRP ID in the cache is closer to the target PNRP ID, the requested peer sends a response to the requesting peer containing the IPv6 address of the peer that represents the entry with a PNRP ID that more closely matches the target PNRP ID. Using the IP address in the response, the requesting node sends another PNRP Request message to the IPv6 address to respond or examine its cache
- If the PNRP ID is not found and there is no PNRP ID in its cache that is closer to the target PNRP ID, the requested peer sends the requesting peer a response that indicates this condition. The requesting peer then chooses the next-closest PNRP ID.

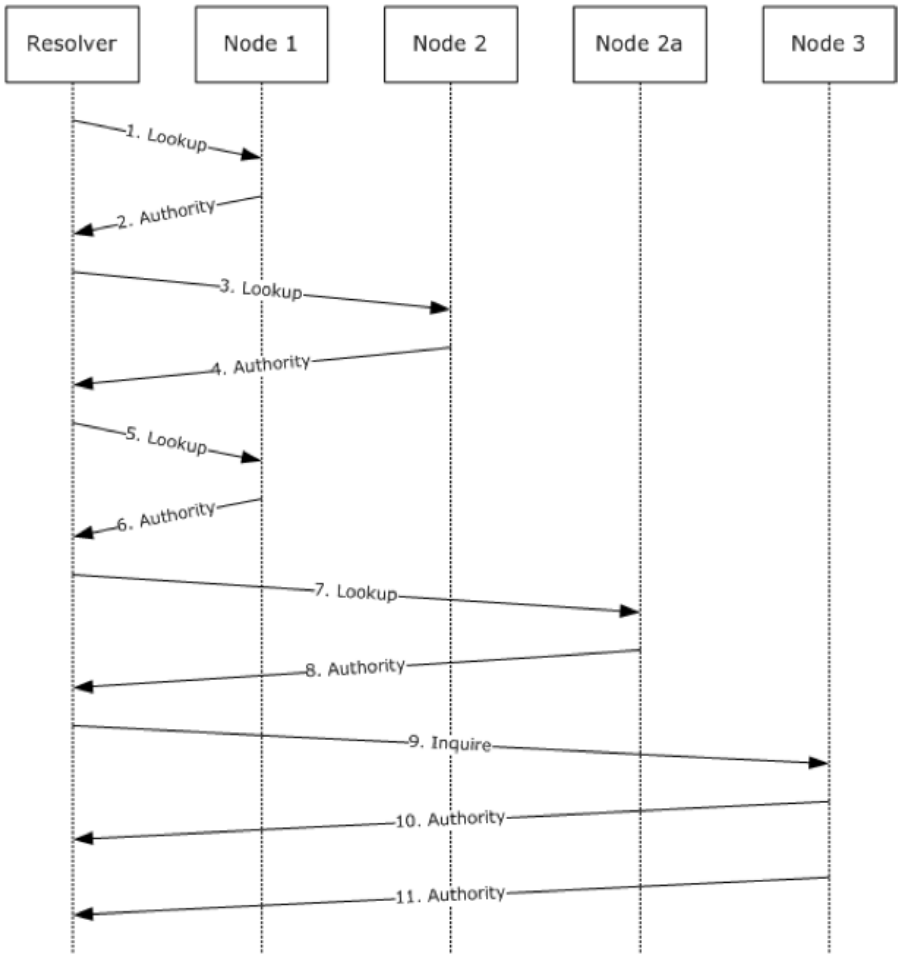

Figure 3.2: PNRP Resolving an Endpoint

The requesting peer continues this process with successive iteration, eventually locating the node that registered the PNRP ID.

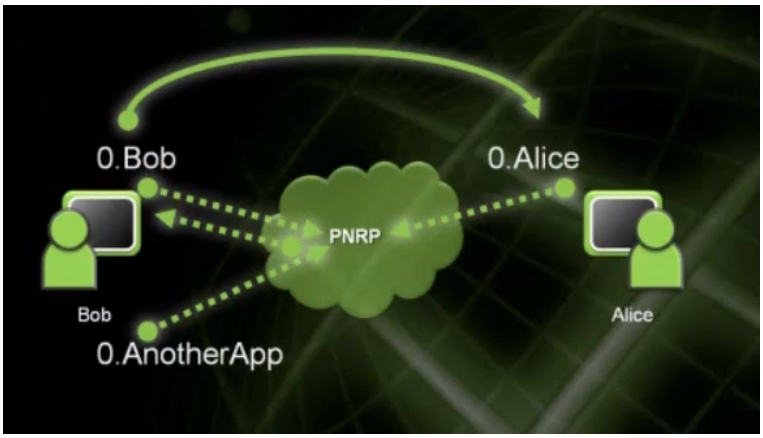

Figure 3.3: PNRP Bob & Alice Communicating

3.4 PNRP Clouds

A PNRP Cloud represents a set of peer nodes that can communicate with each other through the network. The term cloud is exactly the same as peer mesh used throughout the description of this dissertation.

A cloud instance is uniquely identified by its name which is case-sensitive. A single peer or node may be connected to one than one cloud.
PNRP uses three cloud scope, in which a scope is a grouping of computers that are able to find each other.

- **Global Cloud** - The global cloud corresponds to the global IPv6 address scope and global addresses and represents all the computers on the entire IPv6 network. There is only a single global cloud. If users wish to create a P2P connection over the internet, then the global cloud is used.

- **Link-Local Cloud** - The link-local cloud corresponds to the link-local IPv6 address scope and link-local addresses. A link-local cloud is for a specific link, which is typicially the same the locally attached subnet. There can be multiple link-local clouds based on network cards and their connections to the network.

- **Site-Cloud** - The site-specific cloud, corresponds to the site IPv6 address scope and site-local addresses. This cloud has been deprecated in

the latest version of PNRP but is still supported by the protocol.

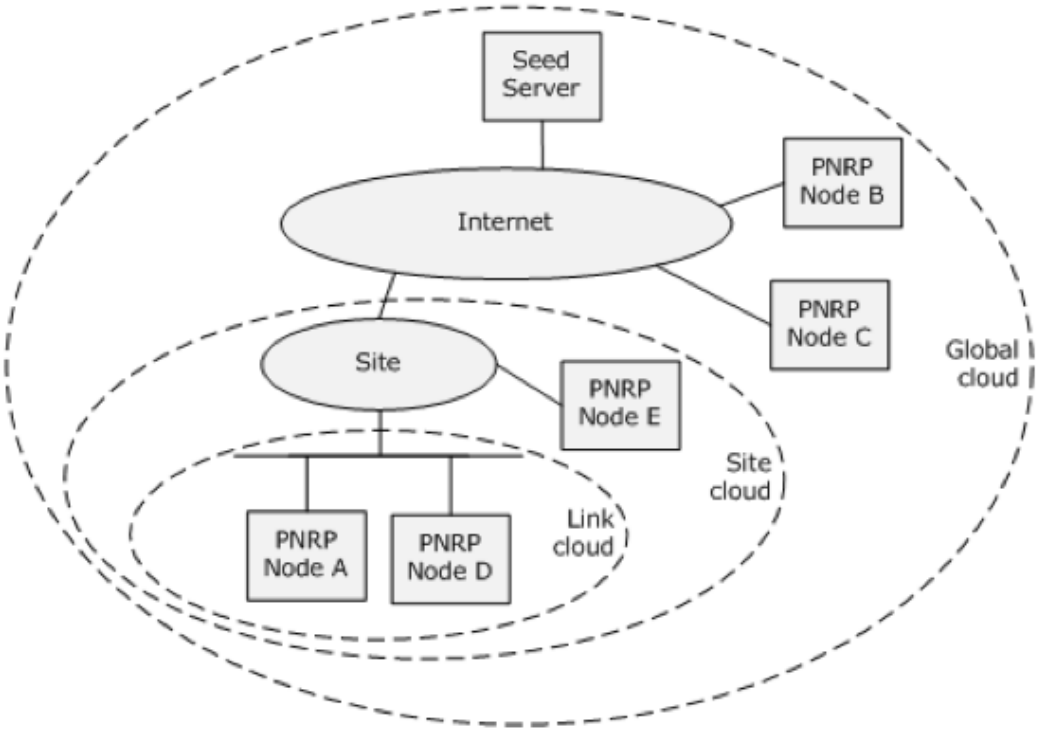

Figure 3.4: PNRP Clouds

3.4.1 Discovering and Joining a Cloud

Cloud discovery is the process by which a node outside the cloud finds an existing node within the cloud. To discover nodes on the same link, a node uses the Simple Service Discovery Protocol (SSDP) to discover other nearby nodes that are already in the cloud. If there are no other nodes in the cloud of interest that exist on the node's link, then the discovering node uses a seed server to find some. If the node still cannot discover any other nodes in the network, then network connection issues are reported.

When a new node is discovered, the joining node then engages in a "synchronization conversation" with the existing node to obtain an initial set of PNRP cache entries. The existing node provides the joining node with a selection of entries from its cache. On completing the synchronization, the joining

node may access the cloud; the joining node now has enough information to perform resolves of PNRP IDs.

Cloud Discovery Process:

1. The Resolver checks to see if the cloud already exists in its Cloud Table. When it does not, the Resolver creates a new entry. It does this by selecting a port and a set of four IPv6 Addresses, or as many as the local system has, not exceeding four, to use for PNRP communication, ensuring that none of the address/port combinations are to be found in the Local Endpoint List of any other Cloud Table entry.

2. The Resolver begins listening for incoming UDP messages on the chosen port, looking for messages sent to the chosen IPv6 Addresses.

3. The Resolver initiates cloud discovery by using the address supplied by the application. CloudDiscoveryMode is set to "LocalOOB", and it now initiates a synchronization conversation.

Having successfully opened a cloud, the PNRP node must first synchronize its cache before initiating Peer Name resolution.

The Resolver sends a SOLICIT message to another node within the cloud (the "Discovered Node"). The two nodes then use PNRP IDs to negotiate which route entries to exchange. The Discovered Node returns a Route Entry for each node that the Resolver is interested in.

A synchronization conversation includes SOLICIT, ADVERTISE, REQUEST, ACK, and FLOOD messages. The following figure shows the sequence of messages sent between the Resolver and the Discovered Node.

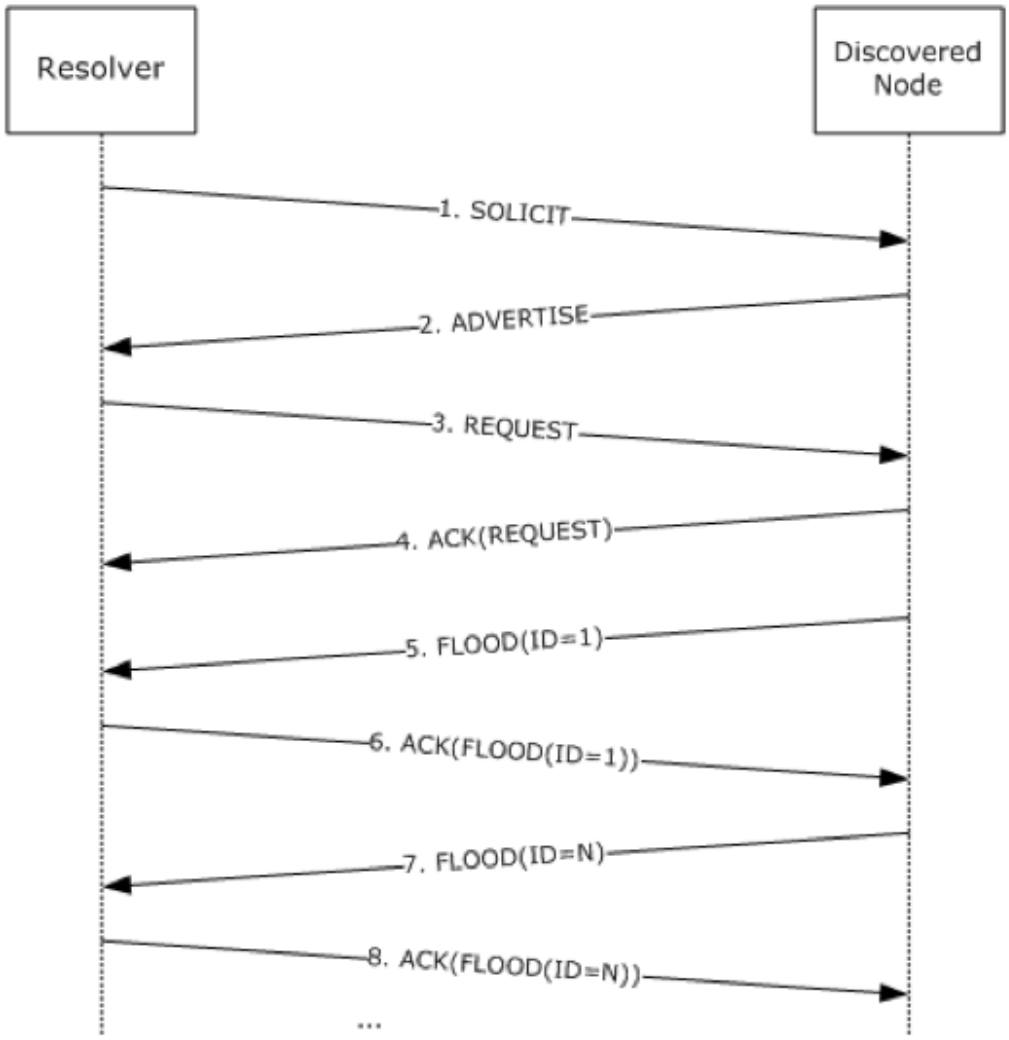

Figure 3.5: PNRP Node Communication

3.4.2 Leaving a Cloud

To leave the cloud, the PNRP node send de-registers all registered PNRP IDs and then terminates.

3.5 PeerChannel

Microsoft has a managed code implementation for a subset of P2P functionality that is exposed by the P2P networking stack called peer channel, peer channel is a part of Windows Communication Foundation (WCF). Since peer channel is a managed stack. you can use and .NET language, which makes implementing P2P applications easier and more productive.

A typical channel in WCF has two participants - a client and a server - but a peer channel can have any number of participants. A message that is sent by one participant will be received by all other participants on the channel. However, certain mechanisms in peer channel allow the user to send a message to an individual user or a certain part of the mesh, instead of the whole mesh, To resolve the address of a node in a peer channel mesh, WCF uses PNRP. When a node is resolved, that target node can either accept of decline the connection. If the connection is accepted by the target node, it sends a welcome message that among other things will contain a list of other nodes that are part of the mesh. If the connection is refused, then the existing node will send the prospective node a refusal message containing the reason and a list of the addresses of the other nodes in the mesh.

The number of neighbours of each node dictates the overall structure of a peer channel mesh that is actively maintained, resulting in an evenly distributed mesh. Once a node has retained its connection threshold, it will start refusing any new connections. If a node loses all its neighbours, it will enter a maintenance cycle, in which it tries to acquire new neighbours to get to its optimum state of three connections. The user does not have the ability to change or configure these thresholds, they are controlled and maintained by design.

The peer channel also tries to improve efficient by limiting communication within the mesh by keeping repetitive messages passed to a minimum. When a node sends a message to the mesh, it sends it to the neighbours to which it is connected. These neighbours in turn inspect the message and then forward

it to their neighbours, but they do not forward it to the neighbour from whom they got the message to start. In addition, a connection to a neighbour might be terminated if it keeps trying to resend a message that has been previously processed. Internally, each node keeps an idempotent local cache of the message ID and the ID of the neighbour that delivered that message. This allows for an optimized mesh network that does not waste resources by with repeating data.

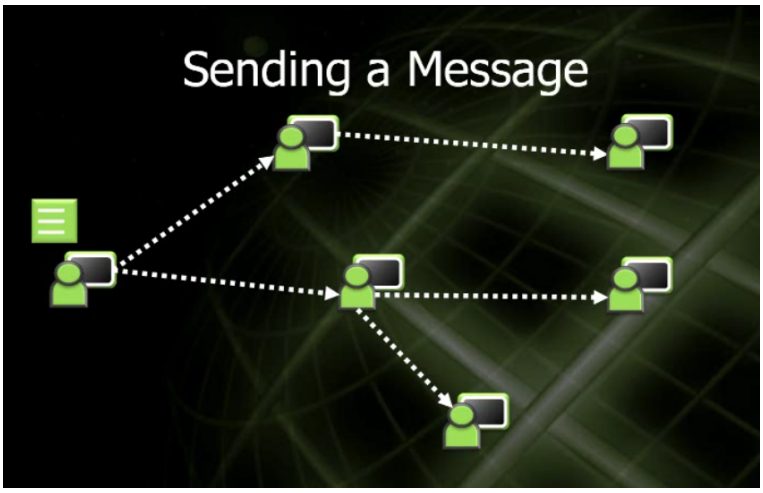

Figure 3.6: PeerChannel Sending a Message

3.5.1 PeerChannel Example

To help explain how Peer Channel networks work, let's compare and contrast a client/server website to a Peer Channel network. When you open a browser to a website, say, www.google.com - under the covers, your PC makes a request to a Domain Name Server (DNS) to resolve the IP address for the www.google.com website. Once you have the IP address, your browser then connects directly to that machine, and the site sends a response back to you in HTML. Multiple visitors to the site all connect directly to the server rather than each other.

Peer Channel networks use PNRP, which is conceptually similar to DNS in that they both allow you to register and resolve names into IP addresses over the Internet. PNRP has a number of advantages over DNS for name resolu-

tion, such as the ability to register and unregister a new name in the PNRP cloud in real time versus DNS name registration, which can take 24-48 hours to replicate worldwide.

PNRP can work in meshs which is essentially an ad-hoc network of nodes. The mesh itself is password protected and the PCs in the mesh can send messages (.NET objects, text, photos, video) between each other anywhere in the world without the need for a central server.

How exactly the machines communicates is not something developers have to worry about, because that is all taken care of b the Peer Channel protocol, the connections are optimized, redundant, and fault tolerant.

3.5.2 PeerChannel Security

You can specify the security settings for peer channel by using a Security property, which is available on the NetPeerTcpBinding, which is the binding used for creating P2P connections. You can apply four types of security at this level and they are exposed by an underlying PeerSecuritySettings class. The four options are as follows:

- **None** - No security is required.
- **Transport** - No message security is implemented, only neighbour-to-neighbour security is required (HTTPS).
- **Message** - Only message authentication is required when communicating over an open channel.
- **TransportWithMessageCredential** - A combination of Transport and Message.

Peer channel provides two ways to authenticate two peers , which are configured using the PeerTranspostSecurityElement.CredentialType: either via Password or X509 Certificate. When CredentialType is set to Password, then every peer needs a password to connect. The owner of the mesh is responsible for

setting the password initially and communicating the password to peers who you would want to join the mesh. An X509 certificate is used when the CredentialType is set to Certificate.
When an application initiates a peer channel instance, an instance of peer channel transport manager is started. The transport manager resolves the endpoint address of the requested peers and the mesh. Once the address is resolved, the transport manager initiates a connection request to each of the peers.

3.5.3 PNRP Scalability

A PNRP cloud has no scale limitation, and could consist of millions of registrations; because it would be prohibitively expensive, both in bandwidth and memory, for every node to cache every single registration in a cloud of this size, the selection of neighbours to cache is of critical importance to ensure a reasonable trade-off among search time, bandwidth, and memory consumption. Cache management is handled by each individual node through a combination of routing messages sent between the node and each neighbour in its leaf set.

A study into the Internet Scale Testing of PNRP in 2006, used a distributed network simulator, WiDS, to simulate a heavily populated PNRP environment [Yang et al., 2006]. The study used real-world Internet characteristics and PNRP node behaviours to simulate the Internet environment, assess PNRP performance, and support protocol debugging. The study preformed investigated two separate PNRP meshes and the merging of their nodes. The results of the study concluded that the PNRP node resolve time did increase with the number of nodes that it had to resolve, which is to be expected due to longer routes and more node hopping to find the shortest path. The research also noted flaws in the PNRP protocol with regards to the Repair-Lookup message which gets sent to detect whether a join message must be conducted between nodes between two meshes. The flaw only occurred at mesh sizes over 242k nodes, and appeared to work correctly and faster with a lower mesh node count. It is interesting to note that there were flaws in the protocol, but as this research was published in 2006, PNRP has developed since this date and

one would expect this flaw to be corrected by the writing of this dissertation, however, no further documentation was available to confirm this. What is important to note is that PNRP, by, design has no scale limitation, and the flaw found through the distributed network simulation will have no effect on smaller mesh sizes as per out application design (Chapter 5).

Chapter 4

Case Study and Research Methodology

4.1 Introduction

The purpose of this research project is to determine if a distributed social network is a viable alternative to centralised social networks such as the likes of Facebook, Bebo and Google+. Social networks are only as attractive and strong as the numbers of users they contain, as the numbers of users is proportional to the number of social connections that can be made. The opinion of current social network users is vital to try and determine the features that users currently enjoy on such networks and what they would change if they had the power to. This research hopes to show that users would be keen on trying a distributed social networking approach as opposed to the current popular model of a centralised approach.

4.2 Research Participants

Participants were recruited from members from the Department of Computer Science. The study aimed to gather data and results from one hundred members related to the field of Computer Science and to a subset of those members who actively participate in online social networking. The reason for this choice

of participant is that active social network users would be more aware of the system, its interactions and functionality, while computer science students would be able to relate to the technologies and methods of the proposed P2P social network during this study.

4.3 Location and Research Context

The research and proposed survey was distributed to participants during set fifteen minute intervals. My research supervisor Hitesh Tewari was happy to allocate fifteen minutes of one of his lectures to distribute the survey to students of his lecture. Before each test began, participants were given a brief description of the research and a list of it's goals that the research team hoped to achieve. Participants were asked to sign a consent form giving the researcher permission to record the data anonymously (in the case of minors, the participant was not allowed complete the survey as this survey was targeted at 18+ students). Each participant was handed out a the relevant forms to complete which consign with the Trinity College Dublin ethic's approval and a survey in the form of a questionnaire. At the start of each study, participants were asked to fill in all relevant details on the forms given and then asked to complete the attached questionnaire to the best of their ability. The questionnaire features thirty questions which is comprised of a combination of multiple choice and opinion based question spaces. The questionnaire is designed to gather as much information as possible from participants about their interactions and opinions of current social networks. The questionnaire is designed to isolate areas where participants would like to see improvements, additional future features, and their opinion of a distributed approach.

4.4 Rational for Case Study

The rational for the case study is to determine participant's views and opinions of the current methods of social networks. The questionnaire specifically asks would the participant prefer a distributed approach as opposed to a centralised approach and to give reasons why/why not. This research and study is highly

valuable for the purpose of the research as the overall purpose of the research is to try and determine are P2P models viable for distributed social networks. If in the case, students reject the distributed approach, the questionnaire will gather why the proposed model is rejected and areas where the model could be improved. If the general consensus of the research is that a distributed social network would be a more highly sought after, then this is a highly valuable point towards the purpose of the research and will greatly contribute to the overall research and motivation for the research.

4.5 Research Findings

A group of 53 participants (50 male, 3 female) took part in the survey for this dissertation. The average age of the participants was 21 with the youngest being 19 and the oldest being 23. Each participant was given a 29 question questionnaire to fill out in the allocated time. Each participant was active in at least one social network with every member having a Facebook profile. We divided the results into three categories, Social Network Participation, Social Behaviour, and Social Data for analysing our results.

4.5.1 Social Network Participation

Participants were asked a series of questions to determine their social networking habits and preferences.

Along with being asked what social networks participants are active in, we wanted to evaluate what social network users prefer and why. As per Figure 4.1, the majority of users choose Facebook as their most preferred social network,. The most common responses as to "Why?" was that Facebook provides each user with the greatest number of friends with ease of use over competing social networks. Users enjoy the simple UI, overall site functionality, and that most of their friends are located on the network and actively participate.

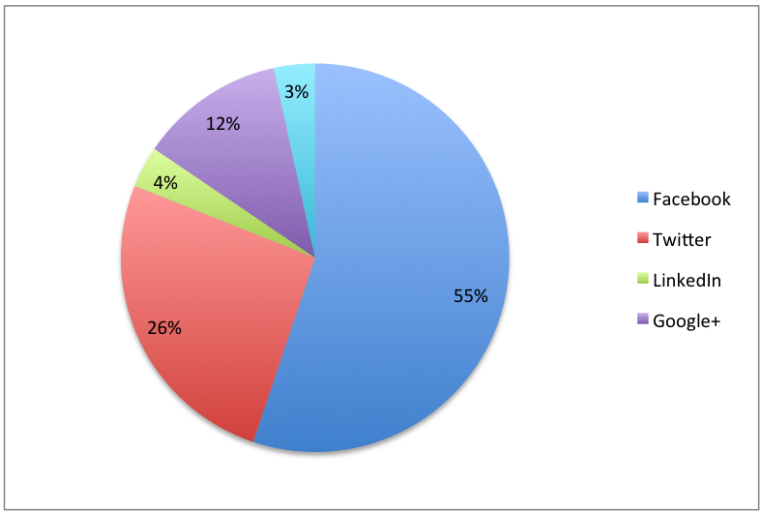

Figure 4.1: Which social network do you prefer to use?

Each participant was asked to specify roughly how many hours they actively spend on social networks. The responses (Figure 4.2) were quite varied and no conclusion can be drawn to an actual common pattern among users. One can note that 61% of users spend 2+ hours engaging in social networks each day. This is a clear indication of how important social networks are becoming in every day life and how involved people are becoming in social networks.

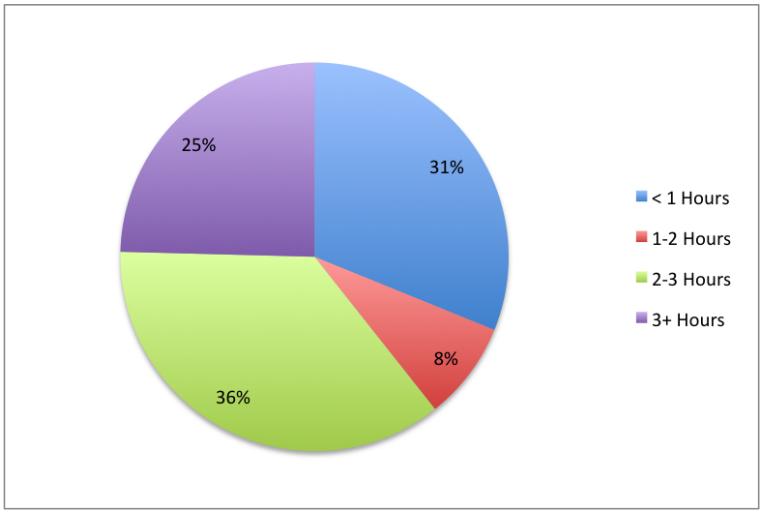

Figure 4.2: How long do you spend on social networks each day?

We wanted to ask participants which devices they use the most to participate in social network activity. Companies like Facebook and Twitter are continuously developing their mobile applications and one would expect a high number of mobile numbers due to the increase in smart-phone numbers in recent years. The results (Figure 4.3) were quite surprising showing that 66% of users still prefer to use their laptops, desktops, and web-browsers to view their social networks, even with the presence of mobile applications. This result of course varies with the location of the participant and level of interaction with mobile or stationary data connectivity.

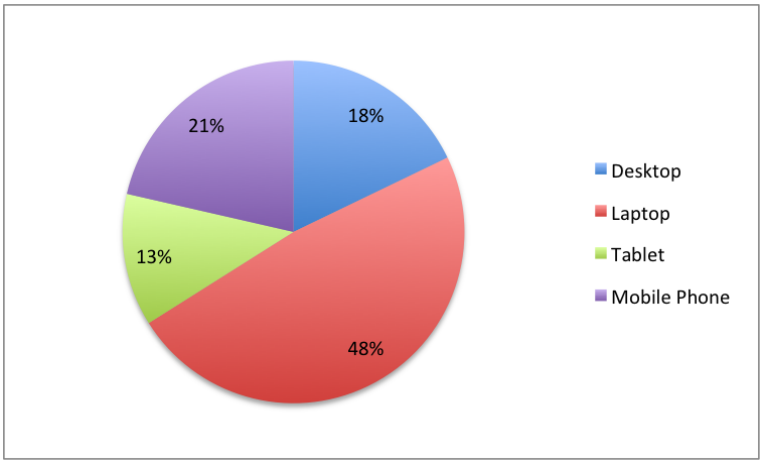

Figure 4.3: Which device do you use the most to access social media?

Social networks are become more and more involved in each users lives as they use social network to interact with their friends, make new friends, and organise events. We asked each participant how influential do they believe social networks are on their everyday lives. The results showed (Figure 4.4) that over 75% of participants felt that social networks were very influential or extremely influential which demonstrates how much users now rely on social networks for social interaction and how that reflects on their day-to-day lives.

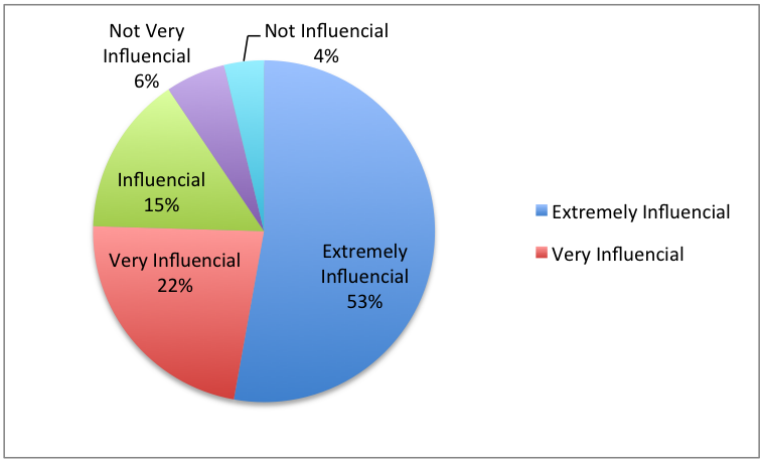

Figure 4.4: What device do you use most frequently to participate in social networks?

4.5.2 Social Behaviour

This section of the survey wanted to evaluate how users behave online.

We asked each participant how often they update their profile information to try and determine the rate of profile update messages that would be sent on a social network. The majority of users indicated(Figure 4.5) that they only change their profile information once a month or once a year, which indicates that the flooding of profile update messages on the network will be a rare occurrence.

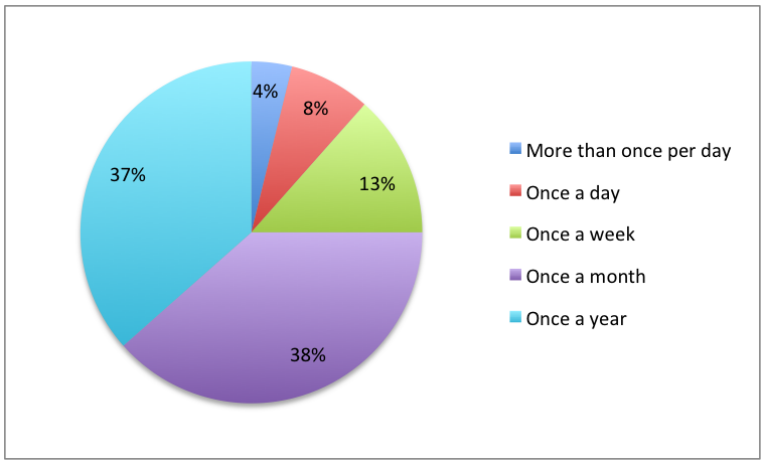

Figure 4.5: How often do you alter your profile?

We wanted to know how many participants actively engage in closed private networks or pages within a larger social scope. Over 90% of users (Figure 4.6) indicted that they participate in closed private groups in their social networks. This is a very important result for the purpose of the research as it shows that a high majority of users like to keep certain parts of their networks private, restricted and protected, which our P2P model aims to implement.

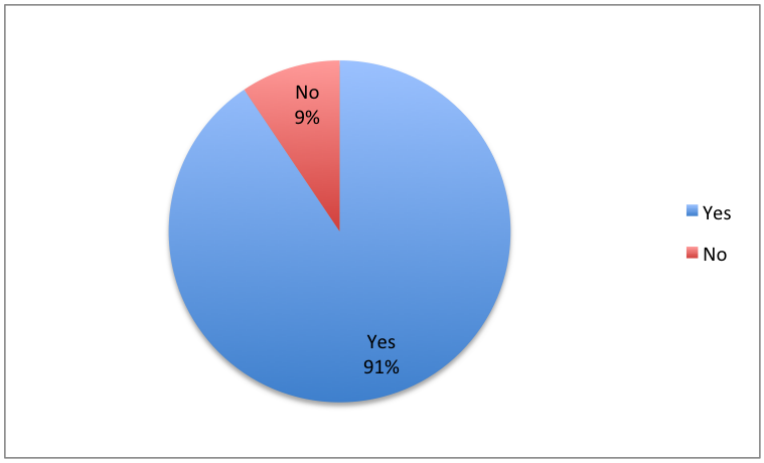

Figure 4.6: Are you active in private social groups or social pages?

4.5.3 Social Security

We asked each participant about security with regards to social networks and how highly they would rank the safety of their online credentials. When asked about the importance of a participants private data, 72% replied(Figure 4.7) saying that they rate the security of their information to be of high importance or extremely important. Social network users care about how security is implemented on their network if their social messages and personal data is secure.

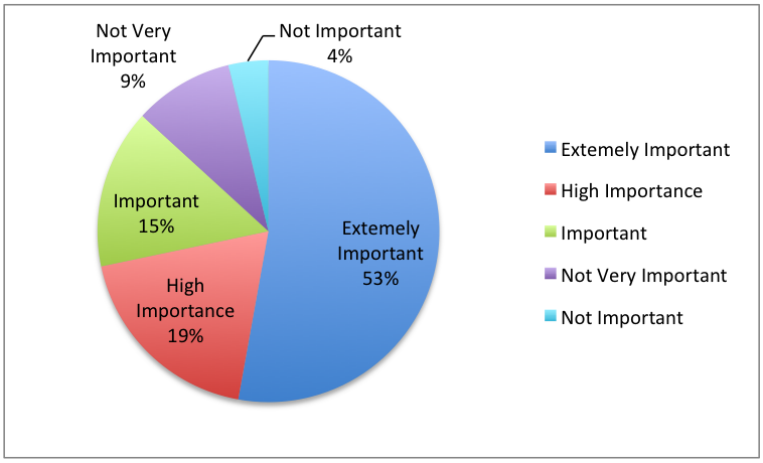

Figure 4.7: How would you rate the importance of security of your information?

The model discussed in this dissertation implements HTTPS transmission between each peer for enhanced security. We wanted to find out if users are aware when HTTPS communication is being utilized on their social networks. All major social networks implement HTTPS transmission, such as Facebook, Google, Twitter and LinkedIn. Our survey results showed (Figure 4.8) that, when asked, 56% of our participants check to see that HTTPS transmission is enabled when interacting with social networks.

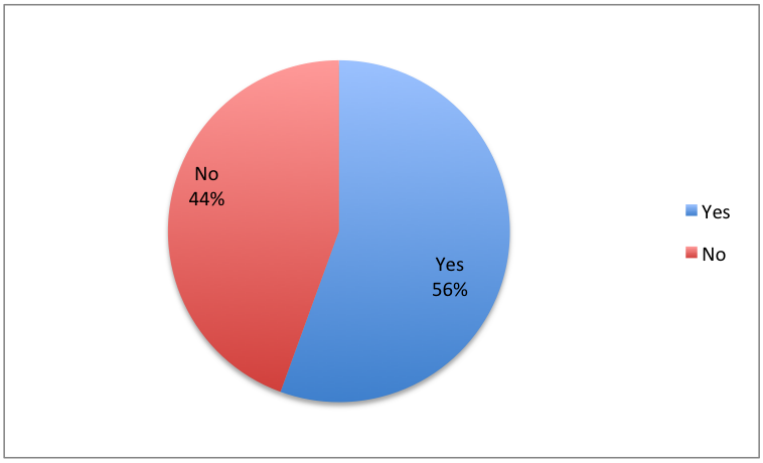

Figure 4.8: Do you check to see if HTTPS is being utilized?

4.5.4 Social Data

The final part of our survey, evaluated participant opinion towards the notion of social data and how this data is stored, used and protected.

Each time a social networking user adds a friend, likes a comment, or views a picture, data is collected and stored and an analysis is preformed on user data to determine their interests. Our survey showed that over 85% of our participants do not trust social networks with their data (Figure 4.9). The research in this dissertation shows alternate methods to the storing and safely of data through a distributed approach which would guarantee the safely of their data from analysis and long term storage. Influential

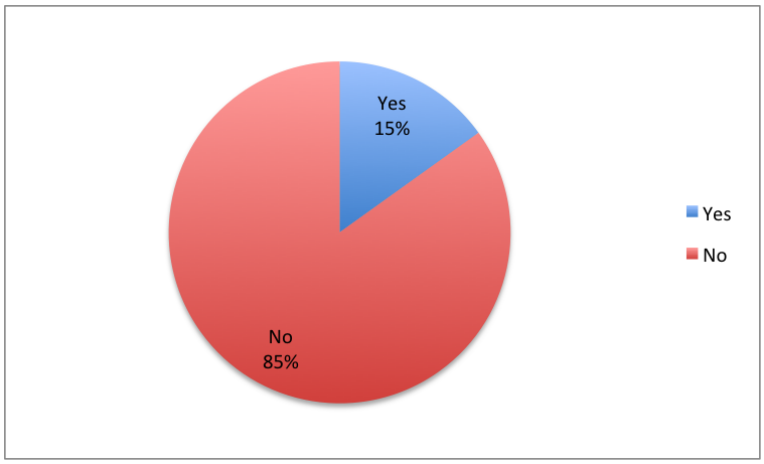

Figure 4.9: Do you trust social networks with your data?

As defined previously, data-mining in social networks involves the analysis of collected data from each user actively engaging with the platform. Users were asked whether they would prefer a network that guaranteed them that no data-mining would occur. Our participants voted (Figure 4.10) that if their had the option to switch to a data-mining free social network, that 62 % of them would. This shows a large amount of users are concerned about the analysis being preformed on their data and the level of detail that they submit to social networks.

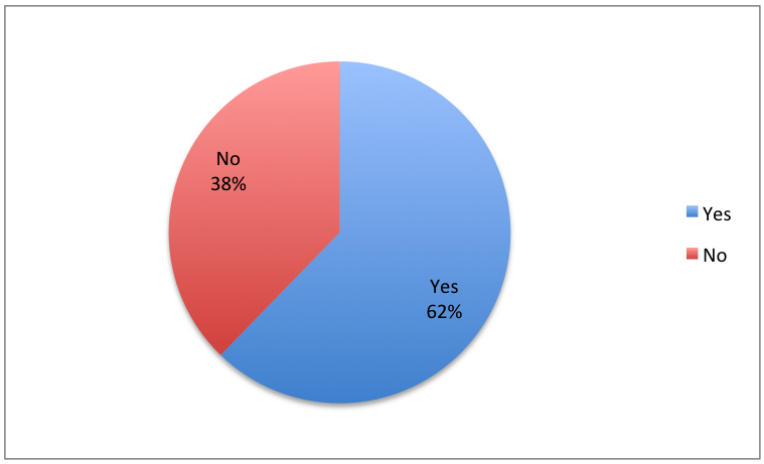

Figure 4.10: Would you switch to a data-mining free social network?

Our survey continued by asking users about advertising on social networks.

Users indicated that they were aware (Figure 4.11) that social networks use their data to target ads at them but there was not a distinct majority.

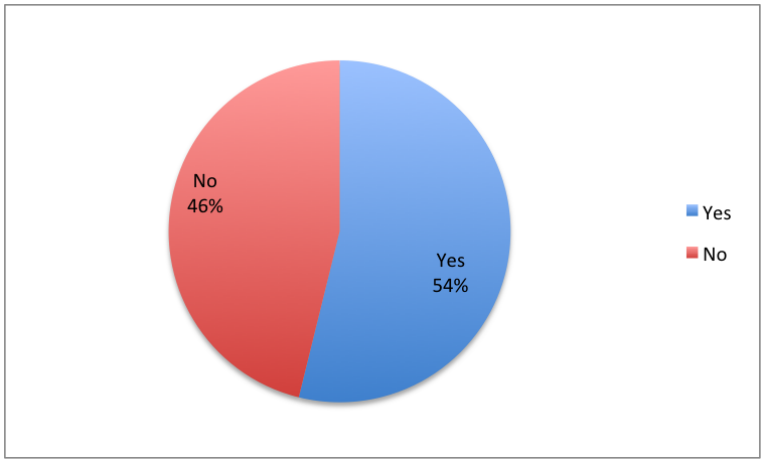

Figure 4.11: Targeted Advertising

Our survey concluded by asking users would they prefer to use a social network with no advertisements. The distributed model proposed in this dissertation proposes a model that is ad free and that there would be no ad content management system implemented. Users voted with 71% in favour of ad-free social networking(Figure 4.12), which indicates that the presence of advertisements takes away from their current social networking experience.

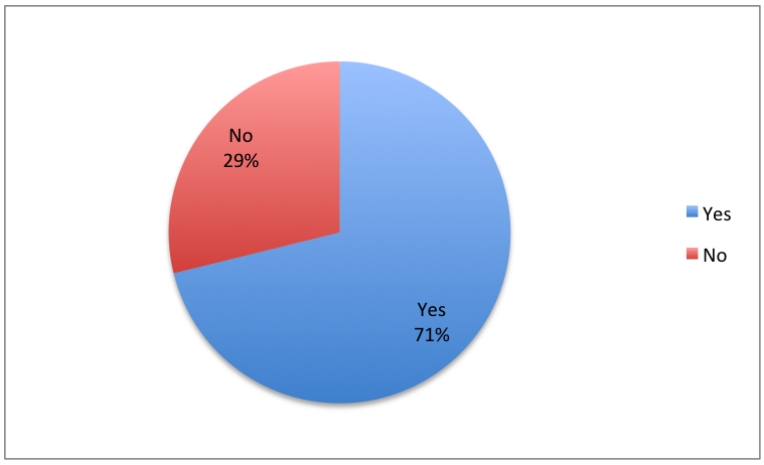

Figure 4.12: Would your prefer a social network with no advertisements?

4.6 Summary of Case Study

The case study preformed was very important in determining users current attitudes and opinions towards social networks today. The survey showed a number of important points that the dissertation aimed to uncover.

4.6.1 Social Networks are becoming a big factor in day-to-day lives

Our survey showed that every user that we asked to participate in the survey was active in a social network. Facebook was deemed the most popular network with Twitter following a close second. Several participants noted that they chose Facebook because"Everyone is on it", "It's clear, easy to use", and "It provides all the functionality you would need" showing that Facebook and popular social networks understand that the amount of users and participants are key and that they must implement the features that users will use the most. Users of social networks are becoming more diverse in the types of devices they are using for social networks, with our participants using desktops, laptops, tablets, and other mobile devices to access their networks. Our survey did not that most users use their laptop to connect to and use social media but one most note that mobile applications and smart-phones are becoming increasingly popular and these numbers will potentially shift in the near future. Our survey showed that users are highly active in social networks, on average spend 2+ hours a day actively participating, and that the majority find social networks extremely influential in their day-to-day lives and social interactions.

4.6.2 Trust Concerns

Throughout our survey, our participants expressed concerns about the level of trust they have in companies being in possession of their social information. With, 85% of the participants expressing that they do not trust companies such as Facebook, Google or LinkedIn with their data, it shows that a majority may seek an alternate solution then sending their data. This is where a distributed model as described in this dissertation would implement a model whereby data

is only stored and sent to trusted users on the network.

4.6.3 Security Concerns

When asked about security with regards to social networks, 72% of users rated security as of high importance or extremely important with regards to the safety of their information online and how their data is stored and accessed. Companies such as Facebook and Twitter implement HTTPS transmission and server-side encryption of users data but retain the right to decrypt and analyse the data for the purposes of marketing and for analysing the network. Facebook retain the right to owning a users data once submitted which raises ownership and privacy concerns that users have expressed concerns about. A distributed social network would provide security between the peers by using a HTTPS connection between peers and implementing data encryption on all social data between peers. Encrypted social messages would only be distributed to trusted peers and no analysis could be preformed on this transmitted data by third party sources.

4.6.4 Opportunity for a Distributed Model to Succeed

The survey results suggested that there are areas of their social networking experience that could be improved by adapting a distributed social networking approach. A distributed social network would only provide personal and private data to trusted peers on the network. The network would use data encryption models to encrypt social messages being sent from P2P giving users peace and mind that their data can only be read and understood by the trusted users that they want to send it to. There would be potential to add advertisements through a stand alone client running the P2P connections, but a pure P2P model would be ad-free and not clog up the user space with distracting advertisements, to which 71% of our participants would prefer to see.

A distributed model surely has many hurdles to overcome with future developments, but a distributed approach to social networking surely presents an interesting alternate model which in the case of the survey research conducted, would benefit the majority of users.

Chapter 5

Application Design

The software that we developed for the project is relatively simple when we break the sections down into each category and describe the overall operation of each function. The application created was developed as a prototype to implement the separate network model, user interface and database design and provide the functionality as described in each model. Each section of the application will be broken down and include diagrams detailing each model.

The success of such an implementation is to provide the P2P networking and data messaging construct while also providing an easy to use and clean user interface. The state of the art review has led the project to use Microsoft's Peer Name Resolution Protocol, to base the backbone of our networking architecture, while also using frameworks such as .NET [Microsoft, 2013c] and Windows Communication Foundation [Microsoft, 2012b] to interact with the network. We will define a complete prototype as one which contains a working implementation of the P2P network, one which contains a database for storing social networking data, and finally one that interacts with a rich user interface to preform social networking interactions.

5.1 Application Goal

The goal of the application was to implement a distributed social network that would implement the main functionality of a popular online social network today. The application by design, is to be functional, secure, and implement

message passing through P2P connections. In the course of building the application, different approaches to the user interface and database design would be considered and tested.

5.2 Requirements

The application was designed to run using Microsoft tools and equipment for the purpose of prototype design and implementation. Each user must be running a copy of Microsoft Windows with an installation of the latest .NET Framework installed.

5.3 Methodology

The application intends to allow a user to create or join a P2P online social network. To do each user is required to create an account, create a profile, and select a social network address to join. Upon joining the network, a user will receive the latest social updates from other users connected in the P2P mesh and any social updates they create themselves is distributed to each user active on the network. The network and underlying management tools track who is online, the messages being sent, database entry and removal, and any queries that another node on the network may have. The user interface will query the data storage to populate a rich and interact user interface that will allow the user interact with the functionality of the overall application. Users are able to view each others profiles, chat amongst each other in group chat, and like and dislike each others posts. Users can upload their own profile picture and write a short biography about themselves and their interests.

5.4 Network Design

The heart of the application is the P2P network model. The model utilizes Microsoft's PNRP (Chapter 3) to create and manage a cloud network connections in the form of peers. Each peer has the ability to register themselves on the network, validate their credentials, and start sending and receiving network

data. Peers connect to the network by assigning their endpoints to a specified address that the social network is broadcast on. Peers then listen for incoming messages and then themselves send any updates to the network that they create.

5.5 User Interface Design

Social network user interfaces must be intuitive to use as they potentially have the reach of any user in the world. The user interface must be designed in such a way as to provide rich social media interactions so that users may have a variety of ways to communicate and are completely abstracted from the background integration with the database and networking layers.

We identified social networking features that we wished to see the user interface and therefore the overall application to feature.

- **View Profile** A user will be able to view the social network profiles of each user on the network. This will be done through a friends page to which the user can click on their friends profile and be redirected to that page. On request, the user interface will query the database for the most recent profile information related to that user. The database will return any data found which will be used to populate a profile page showing that user.

- **Network Status** A user will be able to view which users are currently online and offline in the network. This will be implemented through a separate networks page where the user will be shown a listing of users that are online and offline. The database will be queried based on the status of the network and its members. If a user is online then it is added to the online list and returned the the UI with the same operation occurring for the offline list.

- **Wall-post** A user will be able to view another users profile and leave a message on their profile. The message is of a fixed length and the receiving user or any other user on the network can like or dislike the

post. The wall-post may include links to videos, photos, or just simply a message. Users will be able to click on the sender's name and this will redirect them to sender's profile.

- **Chat** A user will be able to engage in chat with each other user on the network. This is in the form of a distributed chat messaging system. Each chat message that is sent will be received by each user on the network. Users will have a text box entry screen where they may post their chat message to which is will then be distributed to each other user on the network. The user interface queries the chat data and loads a list of recent messages to be displayed to the user.

- **Like** A user will be able to like a wall-post to show their interest in a particular posting or message. Users will be given a "Like" button to which they click and thus increments the like counter.

- **Dislike** A user will be able to dislike a wall-post to show their interest in a particular posting or message. Users will be given a "Dislike" button to which they click and thus increments the dislike counter.

User Interface Model - The user interface will be developed using the model view controller design pattern. This pattern allows functional and rapid development of web based user interfaces.

The model view controller design pattern is comprised of three parts, models, views and controllers.

- **Model** - Consists of application data, business rules, logic, and functions. A model notifies its associated views and controllers when there has been a change in its state. This notification allows the views to produce updated output, and the controllers to change the available set of commands.

- **View** - A view can be any output representation of data, such as a chart or a diagram. A view requests from the model the information that it needs to generate an output representation.

- **Controller** - The controller mediates input, converting it to commands for the model or view. A controller can send commands to its associated view to change the view's presentation of the model (e.g., by scrolling through a document). It can also send commands to the model to update the model's state (e.g., editing a document).

5.6 Database Design

The database design is crucial for storage and access of stored social networking data. The database is central to both the network connection and the user interface. Any validated data or social information that arrives on the network is verified and then added to the database. The user interface directly queries elements of the database to build the UI and provide social information to the user. Database design is very important with regards to social networks due to social networks ability to massively scale and quickly accumulate big data.

The database design below was designed to accommodate system users, profiles, networks, wall-posts and chat messages. The database could easy be expanded or detailed further to accommodate further messaging on the system and thus is flexible and expandable. The database was designed using a model-first approach using Microsoft Entity Framework. A model-first approach is when the model and associations and keys are built in a designer application and then the database is created based on the tables, relations, and models created.

The following tables were designed and created for the application:

User		
Primary Key	Integer	1
Username	String	Conor McEvoy
Password	String	******
Date	DateTime	12/03/2012

Figure 5.1: Network Database Table

Networks		
Primary Key	Integer	1
Network Address	String	Computer Science NW
Network Port	String	8080
Network Name	String	My Network
Date Created	DateTime	12/03/2012
Network Password	String	********
Encryption	Boolean	True

Figure 5.2: Network Database Table

Profile		
Primary Key	Integer	1
Full Name	String	Conor McEvoy
Email	String	mcevoyco@tcd.ie
Location	String	Raheny, Dublin
Bio	String	All About Me
Display Name	String	Conor
Date Created	DateTime	12/03/2012
Display Name	String	Conor
ProfileGUID	GUID	21EC2020-3AEA-1069-A2DD-08002B30309D
isMine	Boolean	True
isOnline	Boolean	True

Figure 5.3: Profile Database Table

Wallpost		
Primary Key	Integer	1
To	String	Michael Hennebry
From	String	Conor McEvoy
Message	String	Hi Michael!
FromProfileGUID	GUID	21EC2020-3AEA-1069-A2DD-08002B30309D
ToProfileGUID	GUID	4NER2730-LMED-0139-P2CD-0640423H3379F
Date Created	DateTime	12/03/2012
WallPostGUID	GUID	91WW20C4-8DDA-29K9-A2FJ-0A7S02B30Z9P
ProfileGUID	GUID	21EC2020-3AEA-1069-A2DD-08002B30309D
Like	Integer	1
Dislike	Integer	0
Likes	Collection(Likes)	Multi
Dislikes	Collection(Dislikes)	Multi

Figure 5.4: Wallpost Database Table

Like		
Primary Key	Integer	1
FromProfileGUID	GUID	21EC2020-3AEA-1069-A2DD-08002B30309D
WallPostGUID	GUID	91WW20C4-8DDA-29K9-A2FJ-0A7S02B30Z9P

Figure 5.5: Like Database Table

Dislike		
Primary Key	Integer	1
FromProfileGUID	GUID	21EC2020-3AEA-1069-A2DD-08002B30309D
WallPostGUID	GUID	91WW20C4-8DDA-29K9-A2FJ-0A7S02B30Z9P

Figure 5.6: Dislike Database Table

Chat Message		
Primary Key	Integer	1
From	String	Conor McEvoy
Message	String	Hi Guys!
NetworkID	Integer	1
Date Created	DateTime	12/03/2012
ChatGUID	GUID	64YW20Z4-8XXA-29K9-N2QK-067502M37Z5L
FromGUID	GUID	21EC2020-3AEA-1069-A2DD-08002B30309D
Like	Integer	1
Dislike	Integer	0
Likes	Collection(Likes)	Multi
Dislikes	Collection(Dislikes)	Multi

Figure 5.7: Chat Message Database Table

The following is the Entity Framework Design Diagram. The diagram is constructed in Visual Studio Entity Framework Designer and then Visual Studio allows us to export and create the database from the generated model. The model below defines the relationships and associations between the tables listed above.

Figure 5.8: Microsoft Entity Framework - Model First Design

Chapter 6

Implementation

6.1 Network

The network layer is responsible for creating the P2P network connection and managing the messages that are send and received via the network connection. By design, the network is bound to an endpoint address which is known to the user of the application and any other member that is participating in the network.

6.1.1 PNRP & Windows Communication Foundation

The network connection is created by binding the P2P address to a "net.p2p://" address using the .NET NetPeerTCPBinding attribute. This binding uses the registered computer IP address and credentials to create a PNRP Peer node on the network and attach that peer node to the PNRP cloud listening on that address. If enabled, the network connection will use HTTPS when sending and receiving messages. The HTTPS uses the network password to encrypt each message as it propagates throughout the network.

6.1.2 User Joining and Leaving the Network

When the application starts up and successfully connects to the network, it will distribute an "Online" message letting each other user know that the user has

successfully joined. Periodically each node will send an Online message as a heartbeat to the network to let users know that they are online. If a heartbeat isnt received in a time-frame, the user is marked as offline in the application.

6.1.3 Message Passing Between Users

Windows Communication Foundation handles the messages being sent and received on the network. Messages are sent through SOAP XML format. Each time a message is sent, the encrypted social message is passed from the database to the network layer, the network layer takes the message and converts it to a SOAP XML packet before being distributed through the open network connection. Once sent, endpoint distributes the social update to any other member in the network.

Once a message is received from the network, it passes through a series of checks to validate the contents. The message under goes an attempted decryption using the network password and the XML headers are checked to validate the message. Once validates, the network passes the message to be stored to the database under the user the message has been received from.

Windows Communication Foundation (WCF) - Windows Communication Foundation is a runtime and a set of APIs (application programming interface) in the .NET Framework for building connected, service-oriented applications. WCF is a tool that is often used to implement and deploy a service-oriented architecture. It is designed to support distributed computing where services have remote connections or clients of the application. Applications that use WCF can communicate using message schemas and choreographies defined in the WS-* specifications. WCF complies with many WS-* specifications. WCF is perfect for the implementation of Peerple's database access layer and the P2P network layer. It allows the application to listen to incoming P2P application and manage endpoint configuration, security and data transactions. This is all done through setup and configuration files and greatly reduces the programming task at hand.

WCF supports many different security models, and makes it easy to implement widely accepted security measures. Because WCF has an extensible architecture, it is also relatively easy to extend WCF security to meet the requirements of a particular application. The default security options range from the traditional transport-centric security to more modern message-based security, as specified in WS-Security [Microsoft, 2004c] and related specifications.

WCF uses SOAP encoding for creating and distributing messages on the network. The SOAP encoding defines a set of rules for mapping programmatic types to XML. XML allows very flexible encoding of data and the serialization of custom network messaging objects.

WCF implements WS-ReliableMessaging [Microsoft, 2004d]to allow messages to be delivered reliably between distributed applications in the presence of software component, system, or network failures.

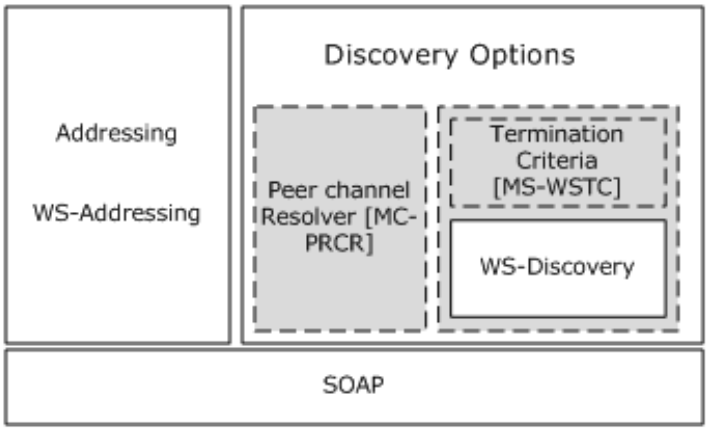

Figure 6.1: WCF Discovery and Addressing Stack

WCF implements WS-Discovery [Microsoft, 2004b] which allows discovery of services in ad hoc networks with a minimum of networking services (for example, where there are no DNS or directory services).

PRCR, the Peer Channel Custom Resolver Protocol REF is a client/server protocol that is used to register and retrieve client endpoint information at a well-known resolver service. The information that is registered and retrieved

is the PeerNodeAddress of clients associated with a named mesh. This information can then be used to establish direct connections among these clients.

Either the PeerChannel resolver (PRCR) or WS-Discovery can be used to get the web service address.

To use WCF and PNRP we must create the P2P connection and bind its endpoints to the WCF infrastructure. We do this by configuring the following:

- **Bindings:**- Specifies all bindings that can be used by any endpoint defined in any service. The binding elements contained in the bindings element can be either one of the system-provided bindings or a custom binding. A binding defines the type of transport, security and encoding used, and whether reliable sessions, transactions, or streaming is supported or enabled. The binding in the case of our application is the "NetPeerTCPBinding" which indicates to the WCF platform that our connection is going to be P2P based and to bind a P2P connection based on service contract, address, and security models defined.

- **Services:** - Contains the specifications for all services the application hosts. Each service specification contains an endpoint element which provides the following information.

- **Address:** - Specifies the service's Uniform Resource Identifier (URI), which can be an absolute address or one that is given relative to the base address of the service. In the case of our application, our address is defined by the creator of the network and is it this address that each node in the P2P network will bind and listen in on.

- **Binding:**- Specifies a system-provided or user-defined binding.

- **Contract:**- Contract: Specifies the interface that defines the contract. In the contract we define the methods and objects that operate on the network, for example, how to send a wallpost over the network, or how to

receive an update from the network. There can be defined through network objects which are all serialised to SOAP packets during transmission and on retrieval on the network.

- **Behaviours:-** Contains a collection of settings for the behaviour of a service like discoverability of service endpoints, settings that authorize access to service operations, the timeout for a service, throttling mechanism of a WCF service, and so on.

WCF provides the perfect framework and development tools for establishing our P2P connection and provides security, connection handling, object serialisation, in combination with PNRP, Peer node discovery and P2P maintenance.

6.2 User Interface

In this section we wish to discuss how the user interface was implemented based on the original goal outlined at the start of the dissertation. The user interface is to not only implement the programming construct designs, but to also contain visual aesthetics which enhance the usability of the application and provide an overall user experience to to the user.
A social network relies heavily on it's ability to be easy to use and to provide users with the right information at the right time. Users must be able to view the latest content and updates to their network and to further interact with the network to share information with friends and connections. The design of the graphical elements do not necessarily have to be clean and in anyway designed as if by a professional digital graphics artist, it does however aid to the whole user experience when interacting with the network.

6.2.1 Technical Design and Approach

The user interface for the final application used the previously defined Model View Controller (MVC) design pattern. The pattern allows the programmer to easily create and integrate a web based user interface with the back end of the application model.

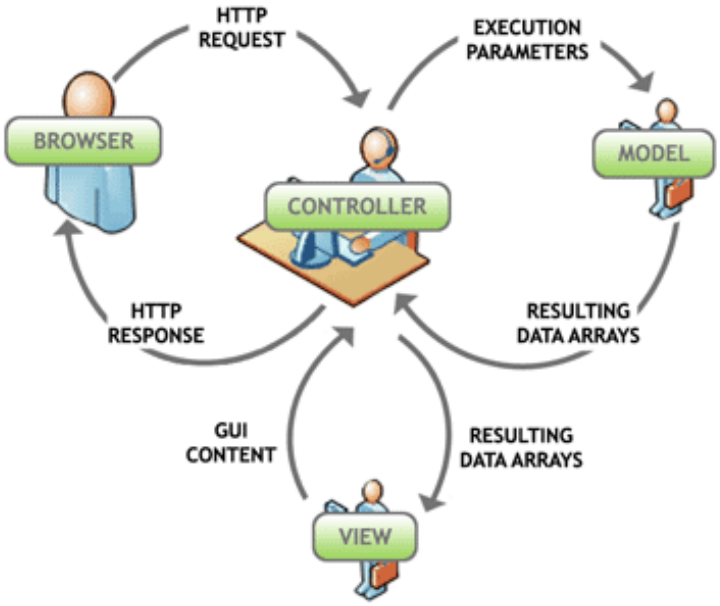

Figure 6.2: Model View Controller Design Pattern

Each view consists of a view, a model and a controller. For the purposes of explaining the implementation we will only describe one of the pages but this model is replicated for each. When a page is requested from the browser, its default controller method is called and the associated HTML view is returned to the browser. Before the view returns the HTML page, it loads any models associated with the page that may be used by the page for building and displaying the user interface to the user. The method returns these models in the form of objects which are passed to the browser alongside the html view. Views are comprised of both HTML code and embedded programming code known as RAZOR syntax [Microsoft, 2010]. RAZOR syntax allows us to create programmed blocks of code that run in the browser and can be used to create UI interactions. When a view is combined with both view-model objects and RAZOR syntax, the HTML page can be built dynamically using the view-model objects and blocks of code.

A page cannot just be dynamically built from GET requests to the controller, but it can also post objects back to the controller or in-fact any other controller in the application. A once a user has completed a form submission, the post

element is redirected to the controller defined in the Form element, and is thus directed to a receiving method defined in the controller. In the POST method, the object is received, correctly formatted and available to the programmer for further processing or use.

When a page that is defined to display user wall-posts is called, the wall-post controller queries any wall-posts to collect and return to the page for viewing by the user. The GET method creates a view model and queries the database for all wall-posts associated with that user. The wall-posts are added to the view model and returned with the view to the web browser. The RAZOR syntax loops through each wall-post and builds the GUI dynamically. For each wall-post the code creates the post, associated profile picture, wall-post message, like button, dislike button and a click-able link to that users profile. If a user clicks a like button or any kind of form submission on the page, a form submission object is created and posted to the associated method with that post in the controller. If a user clicks a like, the Wall post ID is posted to the Like method. The controller queries the database for that wall-post, increments the like count, and updates the view to the correct like count.

By implementing the above programming model throughout the application, it allowed for the creation of rich, interactive, dynamic user interfaces which also fully integrated with back-end services. Each of the pages of the web based user interface were created using this Model View Controller design approach.

6.2.2 Features

A user interface must be designed and implemented around the functionality that the social network features. For the purpose of Peerple, we preformed my own survey on the most popular social networking sites and tried to implement the most used features for my project. After preforming the survey we discovered that the top five features to implement were:

- Profile Page
- News Feed

- Friends
- Like
- Profile Posts

I made it my goal that by achieving the implementation of these five features, that my social network would be usable, workable and manageable.

Application Navigation - Simple application navigation is key to the interface as it gives the user the ease of use of using and navigating the application. It provides each of the applications features in a clear and precise manner. The application features a header and footer bar for site wide navigation. The header bar contains likes to the related pages and users can one-click navigate to the desired page whether it be chat, wallposts, profile page, or whatever pages are available to them. The navigation bar changes to a login bar when the user is not logged in.

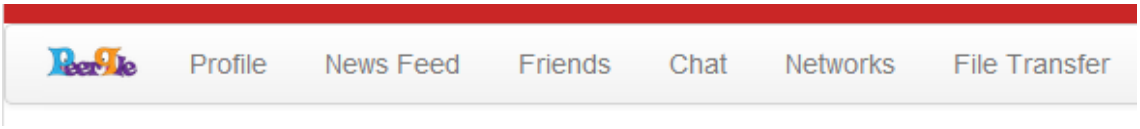

Figure 6.3: Peerple Application Navigation

Register - Each user is required to register with the application so that the application would allow multiple users sign into the same machine. Users are asked to submit a number of details which helps identify them and that can be used to further creating a profile. The application asks a new user for: Full name, Email, Username, Password, and then to confirm their password. After all the information has been validated, being a new user, the user must create a profile page to display their information to other users. The application asks the users for details such as: Display name, Bio, Location, Website and to upload a profile picture. All this information is stored locally on a users machine and be be requested by new members of the network to help identify each user at a later stage. The registration process is quick, simple, and user friendly and both creates an account for the user and builds a profile.

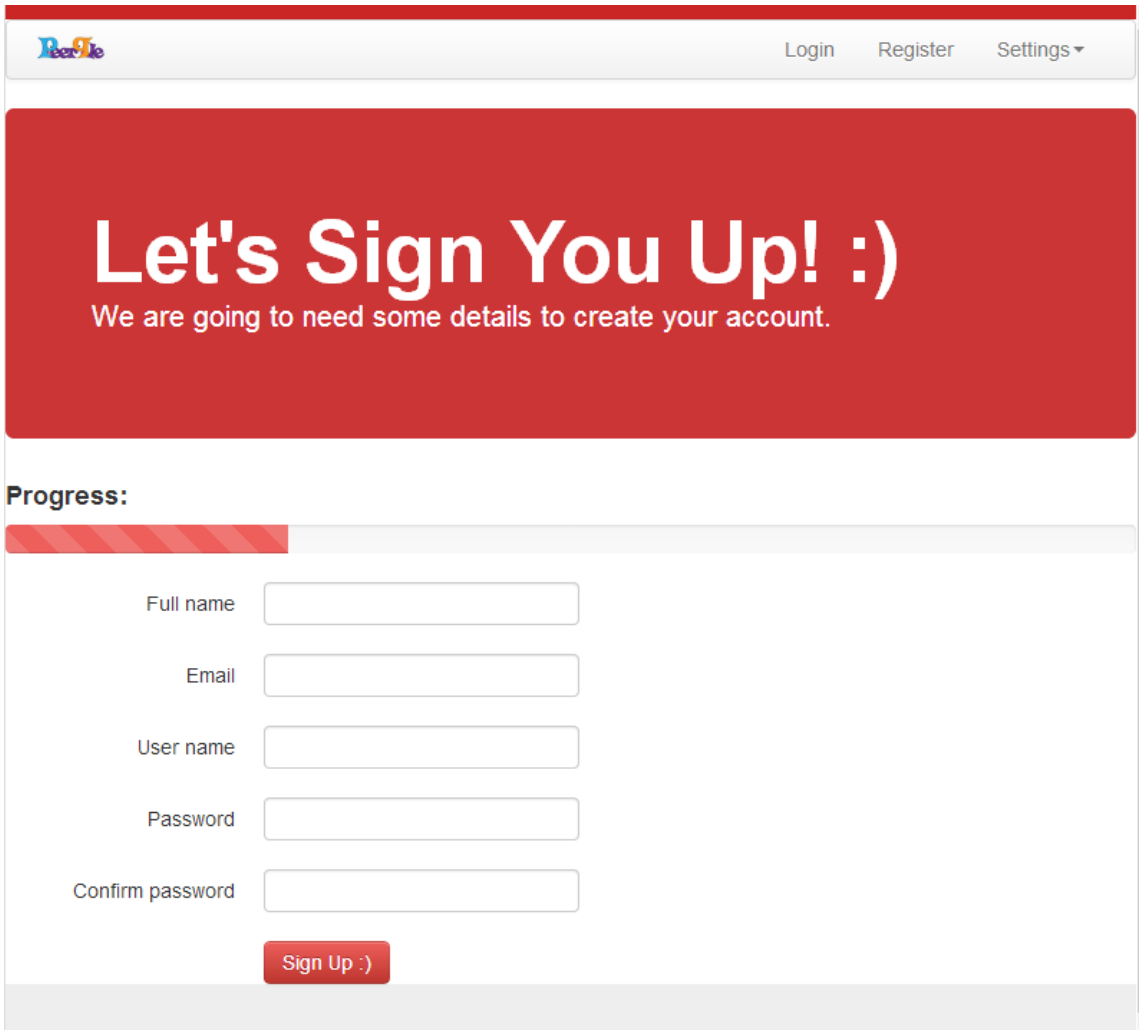

Figure 6.4: Peerple Registration Screen

Login - On the main menu, users are presented with the option to login. Previous users of the system will have created a username and account with the system that binds them to a profile. Once a user enters in details, the system preforms a background check on the data to validate. If the details are incorrect, or the system cannot find a matching profile, then the user is denied access and is asked to re-enter details to log into the system. If the details are correct, then the system logs the user in and presents the users profile page to them , along with any updates since their last log on.

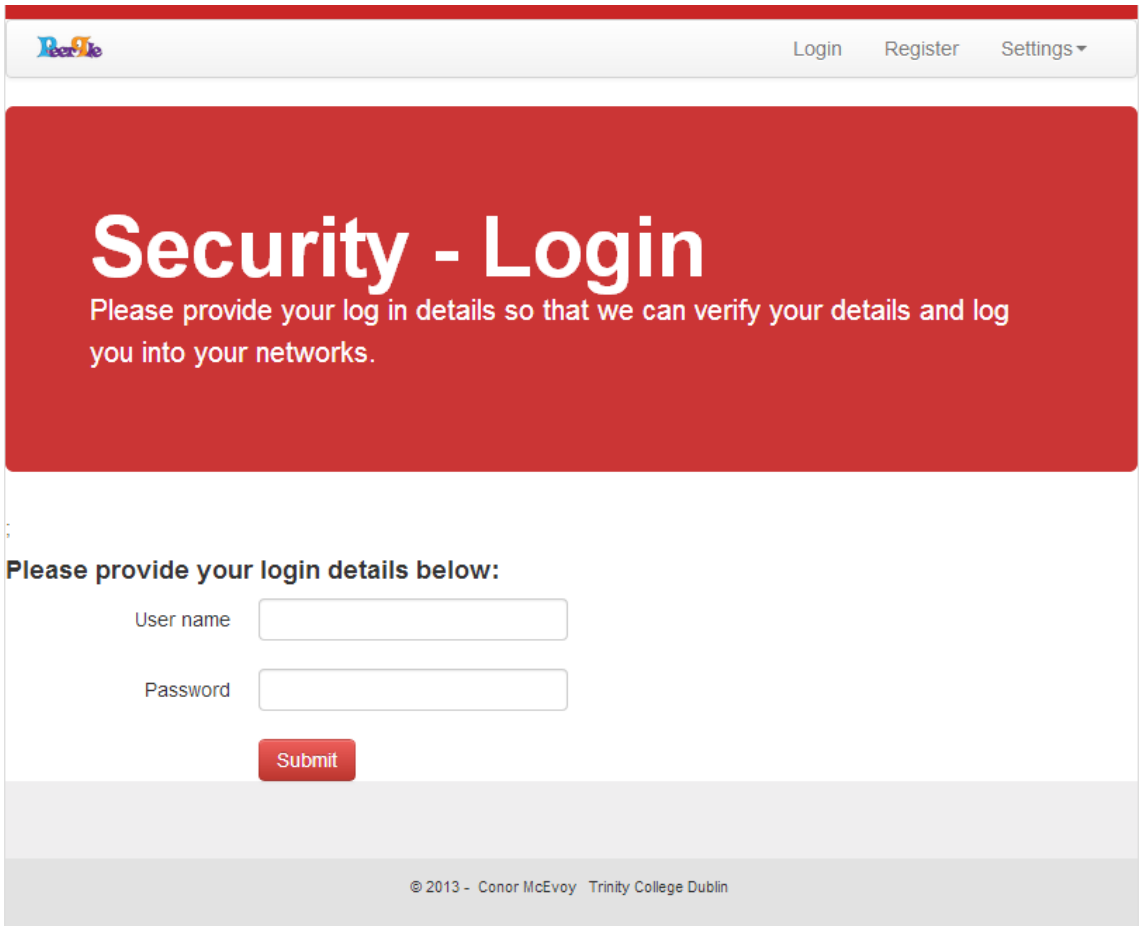

Figure 6.5: Peerple Login Screen

Profile Page - The profile page is point in the social network that lets users define who they are and any particular interests that they have. This helps user identify themselves across the network and promote their views and opinions and particular topics. Users can add their location(optional) to let other users know where they are, or they can customise their profile by changing their account or profile information. The profile page not only displays the users profile details and photo, but also includes a list of the users profile posts.

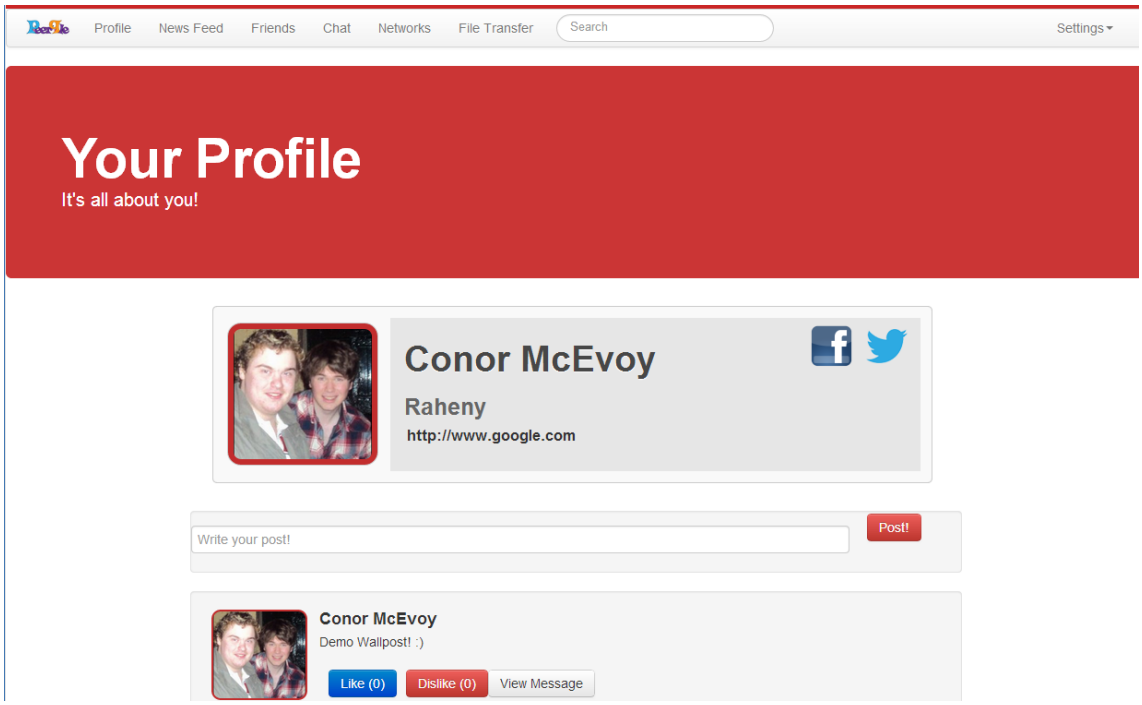

Figure 6.6: Peerple Profile Page

News Feed - The news feed is a feature which correlates all of the social media interactions taking place on the network. It shows users what other users are talking about and then can join, like, dislike or comment on the conversations. It easily allows users to view activity in the network and share information with others.

Friends - One of the key features in a social network is the ability to manage and view a list of your connected friends. In Peerple, the friends are listed as online, or offline to show their status of connection to the network. Users can click on their friends to load their profiles and show their details and profile posts.

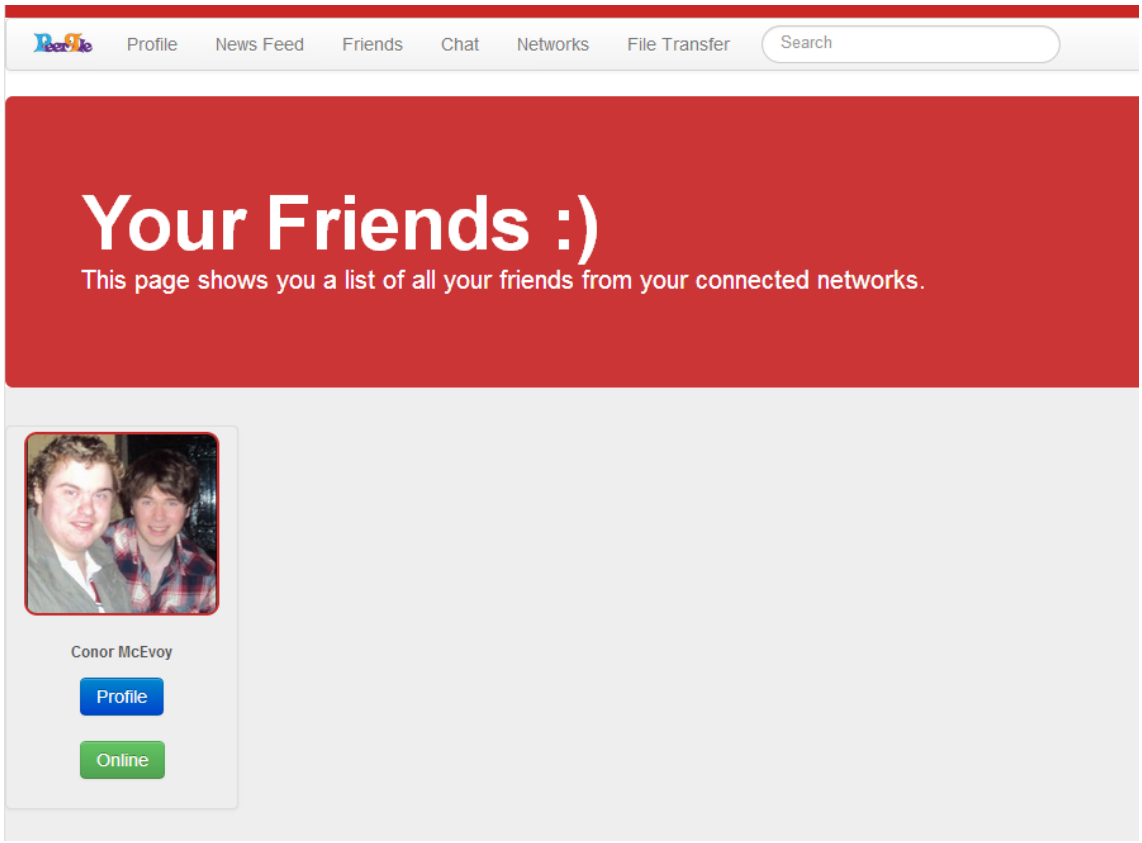

Figure 6.7: Peerple Friends Page

Like - The like button is a button attached to comments or wallposts that allow a user to express that they like that particular post or piece of social media. It's simple for a user to click the like button and instantly feel like they have contributed to a social interaction.Likes can accumulate by how many users click the button and are a clear indication of an approved posting.

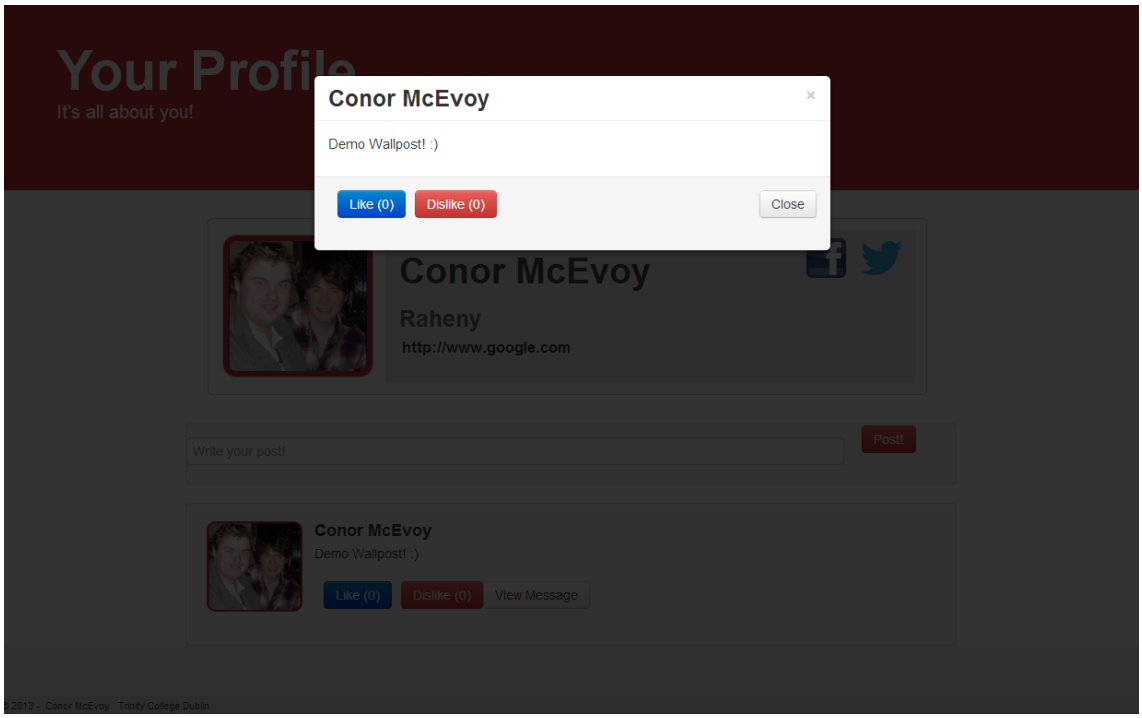

Figure 6.8: Peerple Like & Dislike Button

Profile Posts - Profile posts are messages posted by the user on their own profile to write about what is on their mind or express themselves. Users can post videos, messages and photos to their profile and other users profiles.

Settings - Peerple's setting page allows the user to change any of the settings being used to run the application. Users can change their network settings, profile contents, profile picture, and application settings. Users can also find a link to the application help page to assist them with any troubleshooting they require.

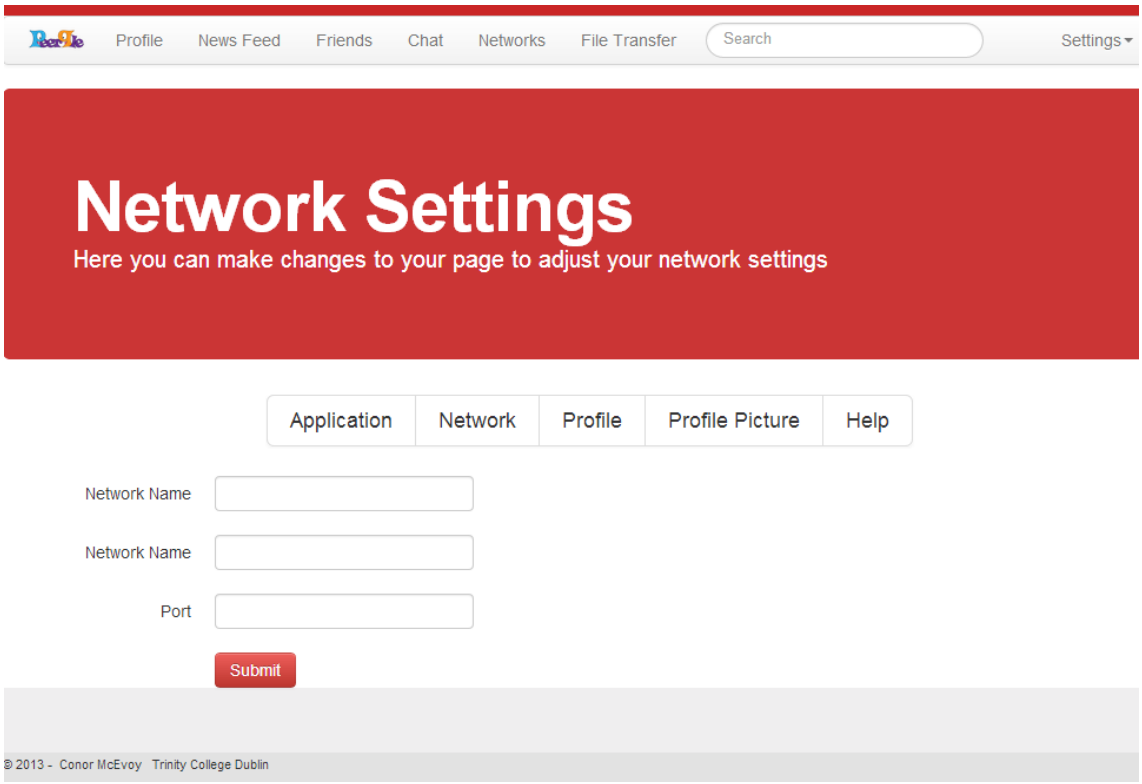

Figure 6.9: Peerple Settings Screen

6.2.3 Packages and User Interface Design

Design Style - Although it is not crucial to the overall result of the research project, the user interface was styled and design with a common theme in mind to be used throughout the site for easy of use and consistency. The site was designed with the wine red theme to add rich colour to the user interface elements. Fonts were chosen to be big, clear, and well defined to make the application easy to use and navigate. Big and colourful buttons are located throughout the site and these act as a combination of navigation and form submission buttons to the user.

GUI - ASP Development and Deployment Environment - ASP.NET is a server-side Web application framework designed for Web development to produce dynamic Web pages. It was developed by Microsoft to allow program-

mers to build dynamic web sites, web applications and web services. It was first released in January 2002 with version 1.0 of the .NET Framework, and is the successor to Microsoft's Active Server Pages (ASP) technology. ASP.NET is built on the Common Language Runtime (CLR), allowing programmers to write ASP.NET code using any supported .NET language.

ASP integrates with Microsoft's RAZOR view engine which allows programming syntax to create dynamic web-pages. Razor was developed in June 2010 and is based on the C# programming language. Razor supports Intellisense, is unit testable, and supports pages as opposed to master html pages. The functionality provided by RAZOR syntax and integration, allowed me to create dynamic web-pages in a much more programmable way. This greatly added to both the speed and quality of the application development process.

GUI - Twitter Bootstrap - In August 2011, Twitter released a set of tools and scripts containing rich user interface elements and graphical templates for creating websites and web applications called Twitter Bootstrap [Twitter, 2012]. The collection contains HTML and CSS-based design templates for typography, forms, buttons, charts, navigation and other interface components, as well as optional JavaScript extensions. Twitter bootstrap is completely open source and free to use for in developing web applications. Peerple uses the Twitter Bootstrap packages to provide rich user-interface models and designs. This greatly aids to the overall visual aesthetics of Peerple and allows easy to use menus, animations, widgets and typography to the web application.

A Twitter bootstrap is an open source package provided by twitter that supplies all the native Twitter UI elements and animations to web developers. Bootstrap is built upon jQuery UI which JavaScript library that provides abstractions for low-level interaction and animation, advanced effects and high-level, theme-able widgets, built on top of the jQuery JavaScript library, that can be used to build interactive web applications.

6.3 Database

The application's database plays a central role to the overall application and is the integration point between the network and user interface layers. The database is used for the storage and retrieval of all social messages. The database is responsible for multiple functions through out the application:

- **Populating the User Interface** - When the user interface is loaded, it's GET call queries the database for any social updates or social messages to populate the UI elements. The database runs the select query and if found, returns a set of objects back to the controller to be used in the UI.

- **Storing Data from the User Interface** - Occasionally the user may create new social updates through using UI elements in the application. These updates are pushed to the controllers where they are validated and then stored in the database. The database takes the social update and stores it in the associated database table.

- **Storing Social Messages Received from the Network.** - When the network receives a social message, it validates it contents, and then sends the message to the database to be stored.

- **Sending data from the Database to the Network** - When a social message is added to the database, it must be distributed across the network to be shared with other users. The database informs the network layers that a new social message has been added and this message, through the network layer, is distributed to each user on the network.

- **Database Security** The database implements security measures to ensure that its contents are safe and private to the network. Each social message is encrypted and decrypted as it enters and leaves the database. The database itself can only be accessed through a shared password on the network to provide a further layer of encryption and security.

6.3.1 Database Creation and Population

Database Creation - The Applications database was created by using a model-first approach using Microsoft's Entity Framework [Microsoft, 2013a]. A model first approach, is where the creator, designs and builds a model of the database using a database design tool and then creates the database from the designer. The database definition was created during the designing phase of the application design. We used Microsoft Visual Studio's Database Designer tool to model the database to specification. Once designed, Visual Studio, compiles the model design to a SQL script which is comprised of a list of SQL operations to build and create the database. When executed, the application connects to the external SQL instance running on a users local machine and establishes the database.

Database Population - The Entity Framework enables you to query, insert, update, and delete data, using common language runtime (CLR) objects (known as entities). The Entity Framework maps the entities and relationships that are defined in your model to a database. The Entity Framework provides facilities to do the following: materialize data returned from the database as entity objects; track changes that were made to the objects; handle concurrency; propagate object changes back to the database; and bind objects to controls.

The primary class that is responsible for interacting with data as objects is System.Data.Entity.DbContext (often referred to as context). The context class manages the entity objects during run time, which includes populating objects with data from a database, change tracking, and persisting data to the database.

6.3.2 Database Technologies utilized

Database - Microsoft SQL Server 2013 - Microsoft SQL Server [Microsoft, 2013b] is a relational database management system developed by Microsoft. As a database, it is a software product whose primary function is to store and retrieve data as requested by other software applications, be it those on the same

computer or those running on another computer across a network (including the Internet). There are at least a dozen different editions of Microsoft SQL Server aimed at different audiences and for different workloads (ranging from small applications that store and retrieve data on the same computer, to millions of users and computers that access huge amounts of data from the Internet at the same time). Its primary query languages are T-SQL and ANSI SQL.

With Microsoft SQL Server 2013, we were able to create an SQL instance to which our database would be created and stored locally. SQL Server provided me with the functionality to create tables, search entries, and manage my whole database. I was able to create and run SQL scripts that allowed me to run complex table operations .

Database - Microsoft Entity Framework - ADO.NET Entity Framework (EF) is an open source object-relational mapping (OR) framework for the NET Framework. The Entity Framework is a set of technologies in ADO.NET that support the development of data-oriented software applications. Architects and developers of data-oriented applications have struggled with the need to achieve two very different objectives. They must model the entities, relationships, and logic of the business problems they are solving, and they must also work with the data engines used to store and retrieve the data. The data may span multiple storage systems, each with its own protocols; even applications that work with a single storage system must balance the requirements of the storage system against the requirements of writing efficient and maintainable application code. The Entity Framework enables developers to work with data in the form of domain-specific objects and properties, such as customers and customer addresses, without having to concern themselves with the underlying database tables and columns where this data is stored. With the Entity Framework, developers can work at a higher level of abstraction when they deal with data, and can create and maintain data-oriented applications with less code than in traditional applications. Because the Entity Framework is a component of the .NET Framework, Entity Framework applications can run on any computer on which the .NET Framework starting with version 3.5 SP1 is installed.

Microsoft Entity Framework help me to construct a model first database for the storage of social messages. Entity allowed me to use a Visual Studio designing tool to craft the database tables and then auto-generated the database once pointed at an SQL 2012 instance running on the local machine.

6.4 Security

6.4.1 Security Model

WCF Network Security WCF (as defined in Section 6.1.3) implements service and message security.

To enable interoperable security systems, companies active in the Web services industry have authored a variety of standards. Specifically regarding security, a few notable standards have been proposed: WS-Security: SOAP Message Security (accepted by the OASIS standards body and formerly known as WS-Security), WS-Trust, WS-SecureConversation, and WS-SecurityPolicy. By adhering to these standards, WCF security can interoperate and integrate with Web services that are hosted on operating systems and platforms other than Microsoft Windows which makes the development of P2P network connections cross-platform capable.

As outlined in the description of PeerChannel (Section 3.5) WCF provides us with *Transport Security Mode*, *Message Security Mode* and *TransportWithMessageCredential*. By design our application uses transport security mode as it has the advantage of being widely adopted, available on many platforms, and less computationally complex. However, it has the disadvantage of securing messages only from point-to-point. Message security mode is several times slower then transport security mode because it has to deal invoking security with the XML nature of the SOAP messages. As described below, our network objects message details are already encrypted using AES, so there is no need to further encrypt the SOAP envelope itself.

Peerple Custom Security Peerple implements its own custom designed security model which aims to focus on securing the network, the database, and the users personal information. Once security is enabled when the network is created, if another user is to join the network they must know:

- **Network Address** - This is the address that endpoint of the application will be pointing to. It is the entry and exit point for social messages. A user cannot find or join the network without a valid network address.
- **Network Password** - The network password is the password that is used in securing the messages transmitted on the network. Peerple's network connection transmits its social messages using HTTPS, and without a valid password, any connections established will simply deliver garbage messages that will be discarded when they get received.
- **Network File** - If a user chooses to use security on the network, the application will generate a file which comprises of an encrypted database password and encrypted network password. On load, Peerple will check for the presence of the file, and if it has a successful password to match the file, will decrypt both passwords and establish the connection to the database and the network. This protects the network and the database, and the user still needs to know network address to gain access.

The network file was created as to create a way to easily and securely distribute the keys for the *Transport Security Mode* which handles the HTTPS point-to-point connection between nodes and the SQL database password which is used to connect to the local DB instance. The creation of the file was inspired through the P2P network BitTorrent and the way it creates .Torrent files to allow the communication of information and network data to begin. When a user of our application creates a new network, it creates a network file with the containing encrypted passwords. The user can then distribute this file to each user that they want to participate on the network. When a user wishes to join an existing network, our application will check to see if the the file is present, decrypt the contents, and use the resulting keys to establish the HTTPS connections between nodes and the database instance. If the keys are invalid, the

user will not be able to create a secure connection to the other nodes in the network and will have no access to any information stored in the local database.

The figure below shows the contents of the network network file, with the passwords encrypted. Although the passwords are identical in this case, a new cryptography cipher could be used to vary the encryption method on each password.

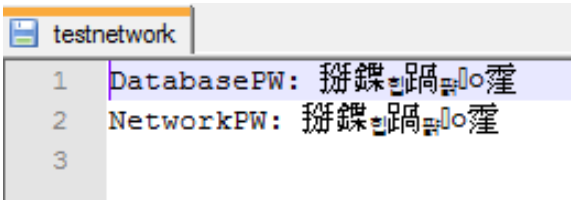

Figure 6.10: Encrypted Database & Network Passwords

Message Encryption - Peerple encrypts and decrypts every social message and profile as it enters and leaves the database. The cryptography key is the network password. This means that everyone who has been given the key or knows the network password should have access to all the information that the database contains. Messages are encrypted with the choice of encryption technique that the user chooses on creation of the network. Messages transmitted on the network are fully encrypted as they leave the database and are attached to the network layer when they are distributed to other users. When another node on the network receives the network message, it tries to decrypt it using it's network password which should be the same as the user's that distributed the message.

Database Encryption - Along with encrypting any information that gets stored in the database (see above), the SQL connection to the local database is also encrypted separately. Users must know the database encryption key to gain access to any of the information stored within.

Multiple Encryption Types - By default, when a user enables security on the network the default encryption technique is Rijndael [Daemen and Rijmen, 2002]

. Users may however select multiple encryption standards for encrypting their data such as AES [of Standards and Technology, 2001] which provides another layer of security by varying encryption methods. Other networks encryption standards would be future work of the application to provide an even more sophisticated layer of security on top of the existing platform.

The figure below shows the result of a user being unable to correctly display encrypted data stored in its database.

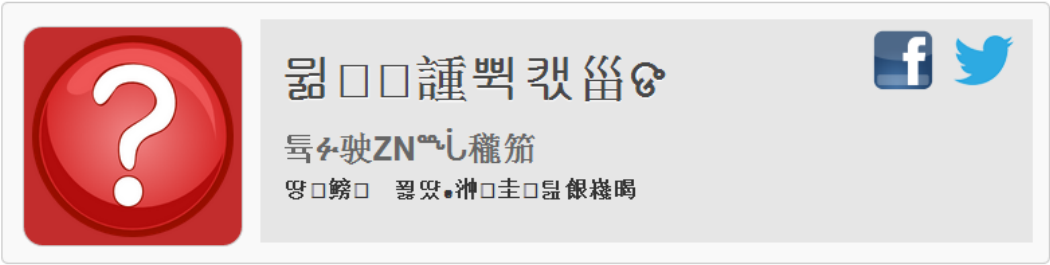

Figure 6.11: Encrypted Profile Protection

6.5 Test & Software

6.5.1 Microsoft Test Suite

Microsoft includes in it's Visual Studio 2012 Ultimate package MSTest, which is a testing suite used to create and run unit tests. Unit testing is a method by which individual units of code have tests written against them to ensure the code still runs as designed. Any chances to what the outcome of the returning code block will cause an error in the unit test and the developer will be notified that something has gone wrong. From the beginning of developing the application it was my goal to provide reliable unit tests to go alongside the application development code. This is common professional coding practice and adds to the quality of the finished code.

6.5.2 Microsoft Visual Studio 2012 Ultimate

Microsoft Visual Studio 2012 [Microsoft, 2012a] is Microsoft's latest version of it's integrated development environment. Visual Studio is used to develop a wide variety of applications from console applications, server applications, mobile applications, games, and fully integrates with Microsoft's .NET framework. Visual Studio provides me with the right development tools and functionality to bring all of the components of Peerple together.

Chapter 7

Results and Conclusion

In this chapter we present an evaluation of the results obtained with the previously detailed techniques, supporting them with quantitative measurements where appro- priate, and identifying their current limitations. We then point out several possible directions for future work and improvement, finally providing some concluding remarks concerning the proposed solution as a whole.

7.1 Evaluation of the Peerple Application & Design

All the measurements presented in this section were performed on an medium-high end laptop, with the following specifications:

- **CPU:** Intel Core i7 - 2.0ghz
- **Memory:** 8GB DDR3 1600mhz
- **Browser:** Google Chrome
- **Network Card:** 1 Gigabit - Broadcom 57765-B0
- **Operating System:** Windows 8

Due to the nature of the application, to fully test the network design and social networking implementation, the dissertation was unable to fully test and

evaluate real-world examination of the designed system. The following section will evaluate the application from a usability and network traffic generated perspective, and how this model would be expected to scale with multiple connections.

7.1.1 Network Evaluation

For the purposes of measuring the network connection, we will measure the size of each outgoing SOAP packet when a update is generated to be distributed to each node in the network. For the purposes of examination, we have turned off point-to-point HTTPS between the nodes so that we can examine the packet.

- **Wallpost** 6KB
- **Chat Message** 5KB
- **Like** 4KB
- **Dislike** 4KB
- **Profile** 450KB (Picture Size Dependant)

The size of each package is generally between 4-6KB each time a user sends an update on the network from its node. If a user sends their profile, including profile picture, the packet becomes much larger, though it can be said that this will be a much less frequent event on the network.

We can calculate that if we were in a perfect network where each user is connected to each other then the amount of traffic generated would be (size of the update * the number of users) on the network. will vary by size and structure of the network, however PNRP guarantees to find the shortest route to eahc nodes on the network and the network will not send a message to a node that has been previously visited, proving the message will be efficiently sent.

7.1.2 Application Evaluation

The Peerple Application designed throughout the course of the research proved to be successful. The application implemented the designed social networking functionality and users are able to create accounts, profiles, join networks, and send each other messages in a distributed P2P manner. The application is only a prototype, but it implements a working network, database, and user interface which is easy to use, observe, and manage.An application such as Peerple, could be further developed to work as a stand alone client and distributed to many users in a further testing stage of the application and models potential.

7.2 Future Work

This section of the dissertation aims to evaluate any future research and work that may be developed after its writing. I have divided the future work into two sections: P2P Network and Peerple Application Development.

7.2.1 P2P Network

Protection Against Rogue Peers

By design, any user that has the network address, network password, and the social networking file created through the security process, has access to the social network. Users can easily share this information with other potential members and they can join the network without each users approval. A P2P voting system could allow each user in the mesh to vote on new members joining or give permissions that only particular peers can invite new peers, such as the creator of the network.

Data Transmission Reduction

The current implementation of Peer Channel broadcasts the message to every user on the network. The messages can be bulky containing images, or vast amounts of social data. Peer Channel already attempts to reduce the number of hops that a message has to take and reduces the repetition of messages

but further research could look into how to reduce the message size through compression techniques, especially as the network scales.

Direct P2P Connection

As described before, Peer Channel broadcasts the message to each user on the network. Peer Channel also allows a user to single out a particular user on the network or group of users. Further work could be done to develop features such as file transfer or private chat from one user to another.

Scalability Testing

At the time of writing, the application and PNRP implementation is not able to be tested due to the large number of hardware requirements. Further work could be conducted to determine how scalable the application and network design is and how that design could be further improved and developed going forward.

Multi OS Support

Peerple's networking and graphical technologies are built and maintained by Microsoft and currently are only adapted to work using Microsoft Frameworks and Microsoft associated technologies. PNRP is an open-source protocol, so future work could be implemented to port the operation of this protocol to Mac, Linux, and other operating systems.

7.2.2 Peerple Application Development

Data Consistency Management

One of the features that we would have liked to implement was a dedicated database consistency management system to manage the updates from the network. The current prototype stores each valid message that is receives and sends a group of messages across the network when a peer requires them. A consistency management system would ensure that each peer has the exact

same copy of the data and that the data would be replicated correctly across the network.

Future Features

Peerple implemented the most common social networking features (Wall-posts, likes, dislikes, chat, profile lookup, security) but there is much more room for research and development into further features. Peer Channel allows the developer to target individual peers on the network instead of broadcasting a message to each user on the network, this would allow for direct P2P file transfers, P2P games, Events, and further applications that may be linked with a social network. A P2P network could implement most of the features that a centralised social network implements today but just in a distributed manner.

Dedicated P2P Client

The Peerple prototype was developed and currently runs from Microsoft's Visual Studio. Further work would could develop Peerple into a stand alone client which manages P2P content such as uTorrent for the BitTorrent network. A stand alone client could allow users to participate in multiple P2P social networks at the time. The current implementation of Peerple is designed to work with multiple distributed social networks with the connections and data transfer all being handled by WCF and PNRP.

Mobile Application

In our survey taken, many users stated that they are starting to use their mobile phone more and more to access their social networking services. This dissertation did not try to create or research the potential for a mobile application for Peerple. Further work would examine porting the PNRP protocol to the Mobile.NET framework currently used in Windows 8 mobile devices, and how mobile phones can operate in a P2P manner.

7.3 Is a Distributed Social Network Model Viable?

The research shown throughout the course of this dissertation, show that there is a demand for a distributed social networking model. The big social networking companies such as Facebook and Bebo, are going to continue data mine user information and opinions for the purposes of targeted advertising and to make a profit.

Like any social network, the key to the success of any new social network is user adoption and activity. Users much prefer to be part of a network that their friends or co-workers are a part of so that they can engage socially with them. It is very hard to shift an existing user from a model where all their friends and connections are to a new model, that said, it has shown that the numbers of increasing users on the likes of Facebook has stagnated and there is potential for a distributed social network to become a niche in the market.

The research carried out in this paper shows that current social networking users are open to new ways to interact with each other and would be interested in a distributed social network. Our participants users in our research survey showed us that there are big issues in how today's popular centralised social networks such as data-trust, data-mining and targeted advertising, and that distributed social networks have the opportunity to solve such problems.

We successfully designed and developed the application Peerple, which is a client platform for the operation and management of distributed social networks. Peerple uses Microsoft's PNRP to successfully connect peers around the world and transmit social updates in a efficient and secure manner. We developed the easy to use GUI and designed a database to store all social messages on a users device. The application provides the basic functionality of a social network, but is open to further development to implement future features.

It is always going to be a difficult task to move users from the popular so-

cial networks onto an alternative model, especially when all of their friends are there. That said, distributed social networks are attracting more and more attention and further into the development of applications such as Peerple and P2P technologies, will only encourage distributed social networking as a viable future model.

Conor McEvoy

Chapter 8

List of Appendices

8.1 Appendix 1 - User Guide

8.1.1 Requirements

The application is built on Microsoft technologies and being a prototype application, is not in a state whereby it could be distributed through executable or via installer. The code may be required through the attached CD along with this report. To run the code, users will have to have the following installed an updated to the latest versions:

- An installation of Microsoft Windows Vista (or higher)
- Microsoft .NET 4.5 Framework installed
- Internet Browser - Google Chrome, Internet Explorer or Firefox.
- Internet Connection
- Microsoft Visual Studio 2008 (or higher)

8.1.2 User Information

Upon loading the application users will be presented with the login screen. Users need to make an account and will be presented with the "Register" button on-screen. Here you create your application account and associated profile

to be used throughout the application and to be displayed to other users on the network.

After a user has successfully created a profile and logged into the application, they must join or create a network. The can do this by clicking the "Networks" tab where they they will be prompted with a choice of joining a network or joining an existing. Users must complete the Network Address, Network Password fields.

Once the user as successfully connected, you must give the network a few moments to query who is online and load the latest social network data from other users. Once loaded, the user is now free to use the applications feature set to create new social messages and interact with other users on the network.

Research Project Proposal

Researcher:
Conor McEvoy (ICS – Integrated Computer Science)

Supervisor:
Hitesh Tewari (SCSS)

Research Method:

This research project proposal is for the dissertation of a study into the application and design of peer-to-peer social networks by Conor McEvoy. The participants will be asked to complete a questionnaire based around the topic of the researcher's dissertation to preform the study. The questionnaire will consist of a variety of questions to evaluate the participant's interaction with current social networks and to evaluate their views and opinion on a distributed social networking approach.

Intended Participants:

The intended participants of this research are participants who participate in social networks such as Facebook, Bebo or LinkedIn. The survey is to be distributed to current Computer Science Students during an agreed time with a lecturer, to ask for their participation in the research.

Participant Recruitment:

I intend to distribute the survey in a fifteen-minute slot in one of my supervisor's lectures and further lecturers until I reach a submission total of one hundred completed surveys. Each participant will receive a copy of the consent form and questionnaire. Each participant that accepts the consent form and the terms and conditions of the survey will be able to participate in the survey.

Researcher Responsibility:

I have a responsibility to make sure that the data entrusted to me by users is communicated and stored in a secure manner. Ethical considerations that must be taken into account include the safe and anonymous storage of this user data.

Parental/Guardian Consent:

If any participant is under the age of 18, then survey will not be able to take part in the survey and will not be given the consent form or the questionnaire.

Researcher's Signature: **Date:**

…….. ……

Questionnaire

Welcome to the survey. Please complete each question below to the best of your ability.Each question is optional. Feel free to omit a response to any question; however the researcher would be grateful if all questions are responded to. When you are finished the survey, please raise your hand and your paper will be collected.

About you

1. **Gender**
   ☐ Male ☐ Female
2. **How old are you?** I am _____ years old.
3. **College Course** _____
4. **College Year**
   ☐ 1st Year ☐ 2nd Year ☐ 3rd Year ☐ 4th Year

Questionnairre

5. **Do you use a social network? If so, please tick below. Multiple choice**
   ☐ Facebook
   ☐ Twitter
   ☐ LinkedIn
   ☐ MySpace
   ☐ Google+
   ☐ Other: _____
6. **How often do you spend using social networks? Choose one**
   ☐ 0-1 hours a day
   ☐ 1-2 hours a day
   ☐ 2-3 hours a day
   ☐ 3+ hours a day
   ☐ Other: _____
7. **What is your main reason for using social networks? Choose one**
   ☐ Finding new friends
   ☐ Connecting with past friends
   ☐ Sharing Media (Photos, Videos)
   ☐ Instant Chat/ Messaging Functionality
   ☐ All of the above
   ☐ Other: _____
8. **What social media do you prefer to use? Choose one**
   ☐ Social Networks
   ☐ Blogs
   ☐ Podcast
   ☐ Instant Chat/ Messaging
   ☐ Video Chat
   ☐ Video (e.g YouTube)
   ☐ Other: _____
9. **How many friends or social connections do you have in the social network that you use the most? Choose One**

- □ 0-10
- □ 10-20
- □ 20-50
- □ 50-100
- □ 100-200
- □ 200+

10. What is the main reason for you joining a social network? Choose one.

- □ All my friends are there
- □ Online Social Functionality
- □ Research based
- □ Event planning

11. What device do you use most frequently to access your social network? Choose one.

- □ Desktop
- □ Laptop
- □ Tablet
- □ Mobile Phone
- □ Other: _____

12. What information do you share with your social networks? Multiple Choice

- □ Name
- □ Age
- □ Email
- □ Address
- □ Phone Number
- □ Profile Photo
- □ College
- □ Work History

13. Are you active in social groups or social pages? (E.g A private group with just your friends)

□ Yes □ No

14. How often do you alter your profile? Choose one

- □ More than once a day
- □ Once a day
- □ Once a week
- □ Once a month
- □ Once a day
- □ Once every few months
- □ Once a year
- □ Never

15. What is your favourite social network and why?

- □ Social Network name: _____
- □ Why?: _____

16. How important are social networks in your everyday life? Choose one

Not Important □—□—□—□—□—□—□—□—□—□ Very Important

17. **How much does social networking effect your social life and lifestyle? Choose one**
No Influence □—□—□—□—□—□—□—□—□—□—□ High Influence

18. **How would you rate the importance of the security of your information with regards to social networking? Choose one**
Not Important □—□—□—□—□—□—□—□—□—□—□ Very Important

19. **How concerned are you that your social networking data could be analysed for the purposes of research and marketing? Choose one**
Not concerned□—□—□—□—□—□—□—□—□—□—□ Concerned

20. **How important is that your social network is accessible through mobile?** Choose one
Not Important □—□—□—□—□—□—□—□—□—□—□ Very Important

21. **How easy is it to access the terms and conditions/data privacy section of your social network? Choose one**
Easy □—□—□—□—□—□—□—□—□—□—□ Difficult

22. **Do you trust social networks with your data? Choose one**
□ Yes □ No □ No Opinion

23. **Are you aware that social networks analyse your data for targeted advertising? Choose one**
□ Yes □ No □ No Opinion

24. **Would you be inclined to switch to a social networking service that does not preform data mining? Choose one**
□ Yes □ No □ No Opinion

25. **Do you use any peer-to-peer based applications? (E.g Bittorrent or Skype) Choose one**
□ Yes
□ No

26. **Do you regularly check to see if HTTPS (encrypted transmission) is being used as opposed to HTTP**
□ Yes
□ No

27. **Do you believe that server-side encryption should be preformed on your social networking data? (e.g Your Facebook Wallposts should be encrypted when Facebook stores them)**
□ Yes
□ No

28. **Would you prefer a system that does not contain advertisements? Choose one**
□ Yes
□ No
□ No Opinion

29. **If you could add features to improve your social network, what would they be and why?**
□ Please describe:

_____

_____

_____

30. **If you feel like you would like to mention anything about this survey, please add below.**
□ Comments:

_____

_____

_____

Thank you for your participation

Bibliography

[Birch, 2005] Birch, M. (2005). Bebo home page. www.bebo.com.

[Buchegger, 2011] Buchegger, S. (2011). Peerson: Privacy-preserving p2p social networks. http://www.peerson.net/.

[Cohen and Navin, 2004] Cohen, B. and Navin, A. (2004). Bittorrent networking client. www.bittorrent.com.

[Cook, 2006] Cook, B. (2006). Oauth- the open authorization framework. http://oauth.net/.

[Daemen and Rijmen, 2002] Daemen, J. and Rijmen, V. (2002). *The design of Rijndael: AES — the Advanced Encryption Standard*. Springer-Verlag.

[Druschel, 2010] Druschel, P. (2010). Pastry - a substrate for peer-to-peer applications. http://research.microsoft.com/en-us/um/people/antr/pastry/.

[Fitzpatrick, 2005] Fitzpatrick, B. (2005). Openid - home. http://openid.net/.

[Gitorious, 2013] Gitorious (2013). Gitorious, listing of distributed social networks. http://gitorious.org/social/pages/ProjectComparison.

[Intel, 2013] Intel (2013). Intel's future processors. http://en.wikipedia.org/wiki/List_of_future_Intel_microprocessors.

[Messina, 2010] Messina, C. (2010). Diso-project. http://diso-project.org/.

[Microsoft, 2004a] Microsoft (2004a). Microsoft pnrp overview. http://msdn.microsoft.com/en-us/library/bb968779.aspx.

[Microsoft, 2004b] Microsoft (2004b). Wcf web services discovery. http://msdn.microsoft.com/en-us/library/dd352335(v=bts.10).aspx.

[Microsoft, 2004c] Microsoft (2004c). Wcf web services security - soap security. https://www.oasis-open.org/committees/download.php/16790/wss-v1.1-spec-os-SOAPMessageSecurity.pdf.

[Microsoft, 2004d] Microsoft (2004d). Wcf ws reliable messaging protocol. http://msdn.microsoft.com/en-us/library/cc485844.aspx.

[Microsoft, 2007] Microsoft (2007). Peer name resolution protocol. http://msdn.microsoft.com/en-us/library/cc239047(v=prot.20).aspx.

[Microsoft, 2010] Microsoft (2010). Microsoft asp.net razor view engine. http://en.wikipedia.org/wiki/ASP.NET_Razor_view_engine.

[Microsoft, 2012a] Microsoft (2012a). Microsoft visual studio 2012. http://www.microsoft.com/visualstudio/eng/.

[Microsoft, 2012b] Microsoft (2012b). Windows communication foundation (wcf). http://msdn.microsoft.com/en-us/library/dd456779.aspx.

[Microsoft, 2013a] Microsoft (2013a). Microsoft entity framework v5.0. http://msdn.microsoft.com/en-us/data/ef.aspx.

[Microsoft, 2013b] Microsoft (2013b). Microsoft sql server 2013. http://www.microsoft.com/en-us/sqlserver/default.aspx.

[Microsoft, 2013c] Microsoft (2013c). .net framework 4.5. http://www.microsoft.com/net.

[Miller, 2010] Miller, R. (2010). Facebook: 50 million a year on data centers. http://www.datacenterknowledge.com/archives/2010/09/16/facebook-50-million-a-year-on-data-centers/.

[Moore, 1970] Moore, G. (1970). Moore's law. http://en.wikipedia.org/wiki/Moores_Law.

[of Standards and Technology, 2001] of Standards, N. I. and Technology (2001). Announcing the advanced encryption standard. http://csrc.nist.gov/publications/fips/fips197/fips-197.pdf.

[O'Neill, 2010] O'Neill, N. (2010). The secret to how facebook makes money. http://allfacebook.com/facebook-makes-money_b9896.

[Parker, 1999] Parker, S. (1999). Napster - join america's largest community of music lovers. http://www.napster.com/.

[Phelps, 2010] Phelps, D. (2010). Hr and facebook: It's complicated. http://www.startribune.com/business/83725197.html?refer=y.

[Rouse, 2006] Rouse, M. (2006). Mesh networking. http://searchnetworking.techtarget.com/definition/mesh-network.

[Schollmeier, 2002] Schollmeier, R. (2002). Definition of peer-to-peer networking for the classification of peer-to-peer architectures and applications, proceedings of the first international conference on peer-to-peer computing. IEEE 2002.

[StatusNet, 2012] StatusNet (2012). What is ostatus? http://status.net/wiki/OStatus.

[Team, 2012] Team, D. D. (2012). The community-run, distributed social-network. https://www.joindiaspora.com/.

[Team, 2010] Team, O. (2010). Creating a free, open and decentralized social network. http://onesocialweb.org/.

[Twitter, 2012] Twitter (2012). Twitter bootstrap ui package. http://twitter.github.io/bootstrap/.

[Weitzel, 2012] Weitzel, M. (2012). Opensocial blog. www.opensocial.org.

[Yang et al., 2006] Yang, H., Piumatti, M., and Singhal, S. (2006). Internet scale testing of pnrp using wids network simulator.

[Zuckerberg, 2004] Zuckerberg, M. (2004). Welcome to facebook. `www.facebook.com`.

[Zuckerberg, 2012] Zuckerberg, M. (2012). Facebook - welcome to my facebook. `http://www.facebook.com/zuck`.