Social Network Services as development platform
An industrial case study
Master of Science Thesis in Software Engineering and Management Programme

NAMRATA MUKHERJEE
HUI ZHOU

Social Network Services as development platform
An industrial case study

NAMRATA MUKHERJEE,
HUI ZHOU

Examiner: MIROSLAW STARON
Supervisor: AGNETA NILSSON

University of Gothenburg
Chalmers University of Technology
Department of Computer Science and Engineering
SE-412 96 Göteborg
Sweden
Telephone + 46 (0)31-772 1000

[Cover:
5 Ways to Use Social Networking to Meet People in Your Niche
Provided by WHITEC0DE.]
Department of Computer Science and Engineering
Göteborg, Sweden April 2013

Abstract

In recent years online social networking services like Facebook and Twitter have become increasingly popular with a large number of active users. The introduction of developer platforms like the Facebook Developer Platform encourages developers to develop and integrate third party applications on the huge user base of these services. The role and use of the information generated by these social network services and applications is not yet well understood in software engineering although they influence software development practices.

In this thesis, we investigated the feasibility of developing corporate solutions on social network services. We used Facebook to develop an application for Commute Greener (Volvo IT) using its open API for traffic data. Further interviews were conducted targeting product owners and developers at Volvo. Based on the outcomes of the study eight key themes, which determine the benefits, risks and limitations of such social network based applications are presented and discussed.

Volvo IT grants to Chalmers University of Technology and University of Gothenburg the non-exclusive right to publish the Work electronically and in a non-commercial purpose make it accessible on the Internet.

To our families and our friends whose love and support is the reason behind our every achievement.

Acknowledgement

Table of Contents

List of Figures and Tables

1. Introduction

Popular social network services have recently opened their doors to developers across the world to create social applications by leveraging their underlying social graphs referred to as "the global mapping of everybody and how they're related" (Wikipedia a). For example Facebook announced the Facebook Developer Platform on May 24, 2007 enabling third party developers to launch and expose their engaging social applications using Facebook's library of open source APIs. The introduction of these third-party applications has led to even higher traffic on Facebook. It grew to more than 800,000 developers around the world building applications for Facebook Platform (Wikipedia b). Facebook kept on gaining a huge number of users and has become increasingly popular with more than 955 million active users as of June 2012 (Wikipedia d). The platforms are still evolving and hence subjected to regular upgrades. With this popularity and increasing number of applications being developed on social network platforms, it becomes important for product managers, developers and testers in a software company to know about the benefits and risks of application development against these regularly updating platforms.

Social network services have an inherent viral property meaning that the applications' user base can undergo exponential growth bringing about quick spread of information similar to real-world social networks (Nazir et al., 2008). Business today is focused on content sharing and social marketing. Several companies are using these social network services to support the creation of brand communities (Muniz, Guinn, 2001) exploiting their viral spread. Companies never before had this kind of opportunity to talk to millions of their customers, send out information and get feedback so quickly. Even consumers never had the ability to talk to each other, criticize or recommend products and services being completely independent of the influence of a company. Thus, a social media presence is becoming increasingly important today. According to a new survey by Harvard Business Review (2012), more than three-quarters (79%) of the 2,100 organizations surveyed said they are either currently using social media channels (58%) or preparing to launch social media initiatives (21%).
Volvo IT shares the same view and believes that an early social presence is important.

This thesis reports from a qualitative study investigating the benefits and risks of developing applications on these platforms for product managers, developers and testers in a software company. This study focuses on application development in the Facebook Developer Platform based on a running application on Facebook and interviews conducted at Volvo IT.

The rest of the thesis is structured as follows: section 2 presents the background and related work; section3 describes the research site; section 4 describes our research approach; section 5 presents the results obtained and our analysis of the results followed by a discussion and conclusion in section 6 and 7 respectively.

2. Related Work

2.1 Background

Before we present the studies which have been conducted on social network services, some definitions are presented in this section as background study.

Web 2.0 is a term that was first used in 2004 to describe a new way in which software developers and end-users started to utilize the World Wide Web; that is, as a platform whereby content and applications are no longer created and published by individuals, but instead are continuously modified by all users in a participatory and collaborative fashion (Kaplan, 2010). With the launch of Web 2.0, companies can reach other companies and end users in exciting new ways and thereby bring about new innovation towards better user interaction using social network services, blogs, wikis, video sharing sites, hosted services, web applications and mashups. Web 2.0 has more content creators compared to the original Web 1.0 where users were limited to the passive viewing of content that was created for them. The differences between Web 1.0 and Web 2.0 are based on technological, sociological, and structural sides (Bernal, 2009).

Along with web 2.0, the concept of social media is widely discussed today. Social media refers to the means of interactions among people in which they create, share, and exchange information and ideas in virtual communities and networks (Ahlqvist, 2008). Kaplan defines social media as "a group of Internet-based applications that build on the ideological and technological foundations of Web 2.0, and that allow the creation and exchange of user-generated content. Social media technologies can be in the form of magazines, Internet forums, weblogs, social blogs, microblogging, wikis, social networks, podcasts, pictures, video, rating and social bookmarking.75% of Internet surfers used social media in the second quarter of 2008 by joining social networks, reading blogs, or contributing reviews to shopping sites; this represents a significant rise from 56% in 2007 (Kaplan, 2010). It is therefore interesting to find out that social media is a revolutionary new trend which should be of interest to companies operating in online space.

Online social networks, a category of social media consists of users who communicate with each other in an online setting in diverse ways (Krishnamurthy, 2009). People form a social network using an online social network on the internet to share and organize contents, and find contacts. The online social networks have some common features, which are usually combined in websites that are known as social networking sites (SNSs) or social sites and function like an online community of Internet users with similar interests and activities. Today these SNSs are rapidly growing over the internet with people constantly desiring to interact with each other for business and personal reasons. The concept of SNSs existed in the 90s with Geocities (1994) and Tripod.com (1995) etc. but their popularity has exploded over the past decade. They have attracted a massive following with most people using SNSs such as Friendster (2000), MySpace (2003) Facebook (2004), Twitter (2006) etc. on a regular basis. As a result SNSs are now among the top visited websites on the internet. Our study is based on these SNSs investigating the benefits and risks of application development in them, the result of which could be used by product managers, developers and testers in a software company.

2.2 Social Network Studies

In case of social networks some researchers have mainly focused on graph theoretic properties of social networks (Granovetter, 1973), (Milgram, 1967), (Mislove et al., 2007), while some on the usage of individual network patterns (Meeyoung, 2007). For instance, Golder et al., 2007 examined an anonymized dataset consisting of 362 million messages exchanged by over four million Facebook users for insight into Friending and messaging activities. These kinds of data also lend themselves well to analysis through network visualization (Adamic et al., 2003).In this thesis we study similar aspects but focusing on the massive user base created by these underlying network of social network services.

Boyd et al., 2007 in their article 'Social Network Sites: Definition, History and Scholarship' have highlighted the fact that SNSs are increasingly attracting the attention of academic and industry researchers. In their introductory article, they describe features of SNSs and propose a comprehensive definition. Then they present the history of such sites, discussing key changes and developments. Finally they summarize the existing scholarship concerning SNSs and discuss some important articles in this domain and conclude with considerations for future research. Among others, some of the key articles they have mentioned and that relate to our study are discussed in following three paragraphs.

Boyd et al., 2007 have pointed out potential privacy concerns in SNSs, mainly concerning the safety of younger users (George, 2006). There has been some research work investigating the potential threats to privacy associated with SNSs. In an academic study on this privacy domain, Acquisti et al., 2005 analyzed 4,000 Carnegie Mellon University Facebook profiles and outlined the potential threats to privacy contained in the personal information included on the site by students, such as the potential ability to reconstruct users' social security numbers using information often found in profiles, such as hometown and date of birth. In another study examining security issues in SNSs, Jagatic et al., 2007 used freely accessible profile data from SNSs to craft a "phishing" scheme that appeared to originate from a friend on the network. The targets were much more probe to give away information to this "friend" than to a perceived stranger. Similarly in our study, privacy of data in SNSs was one of the most crucial factors as we were dealing with confidential data.

Ahmed et al., 2011 also state that an important research area is protection of users profile information from social applications. With the introduction of developer platforms like Facebook's Developer Platform millions of third party developers are building thousands of social applications for existing massive user base. This introduces serious privacy risk because current social networking platforms do not provide any mechanism to control disclosure of user's information to social applications and their developers. In their paper, they investigated two very popular social networking platforms, i.e. Facebook Developers Platform and OpenSocial. They demonstrated inherit flaws in these social networking platforms, pointing out limitations in existing solutions to the problem suggested by researcher community in the area. They concluded with the need for extension of current APIs available for these social networking platforms so that extended version of APIs provide fine grained access control to the user of social network services. Maximilien et al, 2009 also argue that SNS privacy primitives are insufficient in particular, ease of use, too plentiful, do not fully apply to third party applications, and do not take full advantage of the social graphs that SNS users build on these platforms. Their paper provides a first step in resolving these issues. In our study we have developed a third party application and have investigated the ways our data was handled by the Facebook Developer Platform.

In addition to the identified themes above, some scholarships have addressed other aspects of SNSs users and practices connected to them. For example, the ways in which race and ethnicity (Gajjala,

2007), gender (Geidner et al., 2007), age (Quinn et al., 2011) and sexuality connect to and are affected by raise interesting questions about how identity is shaped within these sites. These studies have helped us in understanding the user base of Facebook and if its effects on developed third party applications.

Boyd et al., 2007 in their article contribute to the importance of SNSs, both for practitioners and researchers. According to them methodologically SNS researchers' ability to make causal claims is limited by a lack of experimental studies. They further point out that scholars still have a limited understanding of users of these sites and the purposes of their participation. Such questions require large-scale quantitative and qualitative research. They also believe that richer, ethnographic research on populations more difficult to access would further aid researchers' ability to understand the long-term implications of these tools. Our study is an experimental study for Volvo IT where the product managers would like to know more about the benefits and risks of application development in SNSs.

David, 2008 has written an article as a response to the above article of Boyd et al. He points out that although these SNSs have already moved into the cultural mainstream, they have not received sustained analytical attention. According to him we should not just think about SNS users and their profiles but we should also be thinking about capitalist interests, of third parties using the SNS data, of privacy made public, of the motives and agendas of those that build these technologies, and, finally, how SNS can be understood as archives that represent rich source of transactional data about a vast population of users. To provide guidance to research we might take inspiration from their information flows on one hand and business informing itself through mining, harvesting and analysis of the data on the other. Our study focuses on the capitalist interests and third parties using SNS data investigating benefits and risks of such applications to product managers, developers and testers in a software company.

David, 2008 also points out the fact that the information that the SNS holds is of immense value in the context of a knowing capitalism and we can look at the valuation of these sites to see that. This is illustrated by the recent developments at Facebook, valued at $15bn, where they have reconfigured the business model to capture further profits from its established social networks with the development of 'social ads'. These social ads will be guided and targeted by information held about people and their connections with brands. By focusing solely upon the user, we are overlooking the software and concrete infrastructures, the capitalist organizations, the marketing and advertising rhetoric, the construction of these phenomena in various rhetorical agendas, the role of designers, metadata and algorithms, the role, access and conduct of third parties using SNS, amongst many other things. Lash, 2006 has pointed out the need to address these gaps in knowledge through an engagement with software designers and those constructing brands and working-up algorithms. They further add that additional interviews can be complemented with such research, ethnographies to fill in some of the gaps in the information. While doing our study we also identified these gaps in information and conducted interviews to broaden our research scope.

One of the very important aspects of SNS is the changing nature of the relations between consumption and production. Today companies are using SNSs to collaborate with consumers on product development, service enhancement and promotion. SNSs have become more intuitive and user friendly to provide information that is easily reached by the end user (Salvatore et. al, 2008). Companies are using these tools to get the consumers involved, inviting them to participate in marketing-related activities from product development to feedback to customer service among many others. For example, Zabin et al. (2008) have pointed out that as major companies are increasingly coming to realize this, these consumer voices will have considerable influence in shaping the opinions

of other consumers and, finally, their brand loyalties, their purchase decisions, and their own brand advocacy. But the basic reason for this as pointed out by Krishnamurthy (2009) is not just the large number of users, but the network induced by the popular external applications that use the distribution platform provided by SNSs to grow virally. The Facebook Developer Platform is a similar kind of distribution platform where software applications can undergo viral spread based on theories of Golder et al., 2007 and Krishnamurthy, 2009.

Todd (2006) has also pointed out that there has been little attempt to standardize academic contributions of SNSs and have been generally ignored by academia mainly because the well-known sites were not recognized for their academic discussions. While doing this research we found that very few detailed studies of corporate solutions or applications developed and launched on SNSs exists, Nazir et al. (2008) believe their paper to be the first to provide a first look at the usage and subscription characteristics of SNS (Facebook)-based applications as well as the nature of interaction between users in the context of such applications. They gather data from their three applications developed using the Facebook Developer Platform and show that applications once popular remain strong and tend to retain their rankings. However, they recognize the wide variety of categories for these SNSs. They concluded mentioning that further studies are needed to investigate the SNS application usage.

Colin et al. (2010) mention that SNSs have given the end users digital social experiences by the help of social features. Social interaction helps user to meet psychological needs. Social features provided by these platforms give the user faster, wider and more frequent interactions creating a digital social experience. Suraworachet et al. (2012) has done a study about the impact of social features from end users' perspectives. The results of his study show that people using Facebook's social features and connecting with Facebook page have respectively significant impacts on consumers' attitude. Juste et al. (2008) claims that the availability of social networking APIs makes it possible for third-party developers to tap into these social relationships and design socially-aware applications. The availability of open source social features is a big reason to build more and more applications on these platforms.

According to O'Reilly, 2007, web 2.0 supports lightweight programming models that allow for loosely coupled systems. However, the traditional web-services are usually very complex to enable tight coupling. Though this is necessary in some cases, many of the most interesting applications can indeed remain loosely coupled, and even fragile. Similarly in case of SNS-based applications, the design of the applications is highly affected by the design of the SNSs or platform. The availability of social networking APIs, such as the Facebook Developer Platform, makes it possible for third-party developers to get hold of these social relationships and design socially aware applications. Juste et al., 2008 in their paper describe how social networking groups can be used to help create virtual private computing clusters that consist of nodes spanning different administrative domains and organizations. They developed a prototype implementation and present the quantitative results of virtual private network. Their design integrated with virtual machines, peer-to-peer virtual private networks, and the Facebook Developer Platform API, to enable a user community in which they could easily deploy their application. There is also an interesting article related to application design for Facebook applications where Dmitriy (2010) discusses that applications developed on the Facebook development platform should be aware of the upcoming updates and it should be possible to forecast respective modifications of the application in advance. Dmitriy's study claims that the application design of social network platforms may impose certain restrictions on the platform therefore cause problems when adding new functional features while running the application. From software

engineering perspective, the design of an application is derived from functional and non-functional requirements of software. In our study a Facebook application was developed according to the requirements from Volvo IT but later the design was changed according to the Facebook Developer Platform design standards.

Another interesting feature of SNSs is the availability of analytical data, for example Facebook Insights. This data can be measured, collected, analyzed and reported for purposes of understanding and optimizing application usage. Lakhwinder et al. (2012) explained that analytics has become an important component of web-based system and helps in taking business decisions. The study of Bukhari et al. (2012) has explored the potential usage of social media analytics tools and found that social media analytics can be of great value for organizations in both special events and routine activities. Hossmann et al., 2011 goes a step further with the data collected from SNSs. In their paper they report an experiment called Stumbl for collecting rich datasets comprising social, mobility and communication ties. Stumbl is a Facebook application that provides participating users with a user-friendly interface, which is used to report their daily face-to-face meetings with other Facebook friends. Stumbl logs user interactions on Facebook (e.g. comments, wall posts, likes). This way the compared and analyzed the contact graph, social graph, and activity graphs for the same set of users. In software engineering we often talk about this kind of data used for analysis based on various input parameters.

For our study we needed to test our third party application for Facebook and hence we needed to study the testing framework of the Facebook Developer Platform. There is a lack of methods and tools to achieve quality of a web application (Di et al., 2002). Software testing for third party involved environments may also be more complex as it requires more technical specifications for specific environment (Träskbäck et al., 2003). Stanton in his blog expressed some thoughts on the challenges of testing SNS based applications (Stanton, 2008). These applications need a lot of good testing, especially when it relies on the underlying social graph of users. Performance testing in production environment with real users is very important. Additionally, these applications need to be tested in different environments like browsers, operating systems etc.

Software applications once developed and deployed, need maintenance. Ashish (2012) discusses about companies increasingly facing maintenance issues related to social media applications. According to his study, companies should try to address potential problems before they arise otherwise they may face expensive discovery problems. This has also been approved by Kajko- Mattsson's study which shows the organizations must create a new role responsible for evolving and maintaining high-level web services. Apart from that, the maintenance process has to be slightly adapted to manage investigation of problems and impact analysis which have to be made between applications and the involved platforms (Kajko-Mattsson, 2004).While executing this study, we came across some of the maintenance issues that originate from SNS-based applications.

It is challenging to find the right set of research methodologies to study the impact of SNSs on software engineering as these environments are advancing rapidly and their development context tends to crosscut traditional software development categories. Being powerful tools having massive impact on society it makes it necessary to continue research work on this area.

2.3 Social Network Platform

The social network platforms allow developers to create their own SNS and run them on a server (Wilson, 2010). For example the Facebook Developer Platform is a software environment provided by the SNS for third-party developers to create their own applications and services that access data in Facebook (Wikipedia e). The social network platforms are different from other traditional web applications mainly because these platforms have built in social features which are easily implementable. For example Facebook eliminates the need of having an authentication and authorization module which traditional web applications required. Here the single sign on handles the entire authentication and security aspects. Secondly application design is much simpler in case of social network platforms because most parts of the application design are pre-defined and not many changes are required. Features like analytics are included and available for free whereas in traditional web applications, analytics features need to be added as separate module thereby increasing costs.

There are different social network platforms supporting various SNS for example LinkedIn Developers for developing applications using LinkedIn's open APIs, Facebook Developer Platform for developing and integrating applications on Facebook etc. For our study Facebook Developer was chosen as per Volvo requirements and hence our discussions would be based mainly on the Facebook Developer Platform. Facebook Developer Platform was launched in May 2007(Facebook Developer Platform, 2012) with about only eight applications in its roster. The platform underwent a phenomenal growth when more than 35,000 applications were deployed within July 2008. At the end of 2011, there existed more than 7 million applications and websites integrated with Facebook (Facebook statistics, 2012). According to Facebook, one way of modeling a social product is to allow people to create an identity and let them share it and build a community over time.

2.3.1 Facebook Platform Core Components

The Facebook Developer Platform enables making websites more social by offering a number of social features which can be easily integrated with applications developed in this platform. Social Plug-ins, such as the Like Button can drive user engagement with a single line of HTML. For example the Login Button and Registration Plug-ins lets simplify or eliminate applications own user registration and sign-in. lastly; the Graph API allows access to the full social graph of a given user, allowing creating a truly deep personal experience. Therefore applications developed on can focus more on business logic and social features can be by platform (Facebook Developer Platform, 2012).

Figure 1: Facebook Platform Building Blocks, (2012)

Figure 1 above shows the building blocks of the Facebook Developer Platform. The foundation of the developer platform is the so-called social graph. The social graph is basically a model of users and objects that they interact with and of those interactions. Facebook has introduced one early version of an open graph at 2010, which is an extension of the social graph. Recently, Facebook is trying to extend the open graph to also include actions and objects. These actions and objects can actually be created by third-party applications. The open graph allows applications to model user activities based on actions and objects (Facebook Open Graph, 2012). As Facebook end users play around with the application, social activities are published. Actions would be displayed on user's individual page. In this case, developers could create a persistent connection and relation between the application and application users. This also could drive new users to the application. Along with social graphs there is another component called the Social Channels. These social channels are ways apps can reach users on Facebook. Some examples are news feed, application requests and bookmarks. Building on top of this foundation the following are the two ways by which an application can connect with users and access the social graph(Facebook Developer Platform, 2012).

FBML

FBML is an acronym for Facebook Markup Language. FBML is a tag based language based on HTML and it is able to provide many of the tags supported by that language. It also provides a large set of Facebook user interface and programmatic primitives (Maver et al. 2009). This markup language gives developers the ability to make applications look like they were developed as part of Facebook. For an application created on the Facebook platform, the developer must choose either that the application will support FBML or it will be an IFrame-base application. It is always suggested that developers new to Facebook applications use FBML. However, if the application uses third-party JavaScript libraries that are over 64K in size, developers are not able to use FBML; they must use the IFrame-based application instead.

Facebook JavaScript

FBJS is the Facebook's version of JavaScript. It supports most of the DOM-based manipulation methods but it differs from JavaScript in several important ways. Firstly, its syntax is slightly

different from normal version of JavaScripts. It is one way to protect Facebook itself from malicious JavaScript code. Apart from that, many DOM properties used in normal JavaScript are replaced by get/set property methods. Lastly, many widely-used event handlers in normal client-side JavaScript are not available. For example, when a web page loads, Facebook does not allow any use of the ubiquitous onload() event handler to execute code(Maver et al. 2009).

Facebook Query Language
Facebook Query Language offers a host of SQL-like features and language elements that allow applications to directly query Facebook's internal data tables. However, not all SQL's syntax is supported. Facebook Query Language is powerful in the case that the application needs to get access and return the same data provided by many of the Facebook API calls. This enables the API response in one less round trip to the Facebook servers. Facebook Query Language is also able to "preloaded" for all the pages in an application. This is useful when the application needs to show similar Facebook data on every page (Maver et al. 2009).

Facebook JavaScript Client Library
Facebook JavaScript Client Library allows applications use the Facebook API on the client rather than the server which means Facebook applications are able to run anywhere JavaScript could. Websites that use this library is able to get access to most of the Facebook API as well as a subset of FBML (Maver et al. 2009).

2.3.2 Facebook Platform Application Architecture
Facebook platform provides a Facebook Developer application to allow developers to fill in the details about the application they are about to create. Facebook does not actually host the developer's applications; Facebook creates an outer frame and then calls the application's Canvas Callback URL to get information to display to the users. The workflows will be explained in this section (Maver et al. 2009).

Secret Keys
Facebook platform gives the application a public and private key pair when this application is created using the Facebook Developer application. This key pair allows the Facebook API to know each API call is related to specific application. This key pair is essentially important to developers, because anyone who knows the key pair could start making calls masquerading as that application, or even deleting user data (Maver et al. 2009).

Canvas Callback URL
Canvas Callback URL is the location on the server where the application resides by the developer. The server hosting the Canvas Callback URL is able to handle the callbacks using any web server or language the developer prefers. Facebook only passes a set of data as POST variables to the Canvas Callback URL, and this set of data only contains information about the viewing user, the session, and the application (Maver et al. 2009).

Canvas Page Workflow
Facebook would call the application's Canvas Callback URL when the end user goes to an application canvas page. Developers have to decide if they want their applications to have FBML or IFrame based canvas pages (Maver et al. 2009).
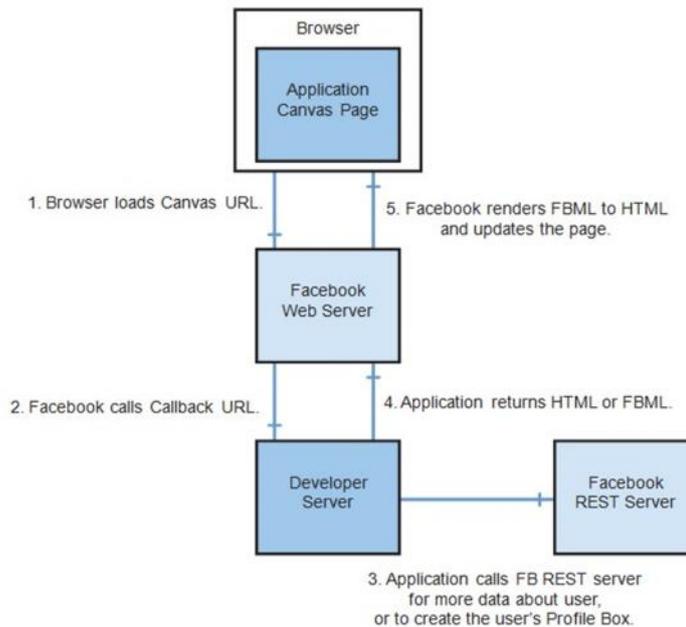
Figure 2: Workflow for displaying an FBML application canvas page (Maver et al. 2009)

Figure 2 shows the workflow for displaying FBML application canvas page. In FBML application, the web browser loads the canvas URL firstly; Facebook will call the callback URL to connect the application resided server. Application will obtain the user data by calling the FB REST server and return FBML to the Facebook web server. Finally the browser display the FBML rendered by Facebook and updates the browser page.
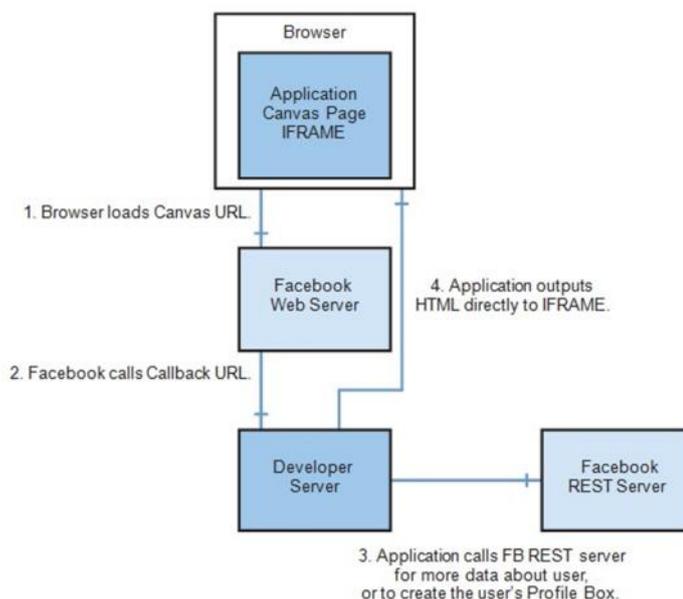
Figure 3: Workflow for displaying an IFrame application canvas page (Maver et al. 2009)

Figure 3 shows the workflow for displaying an IFrame application canvas page. In IFrame application, the web browser will load the canvas URL as same as the FBML application. The different process happens after the application obtains the user data by calling the FB REST server. The browser displays the IFrame-based canvas pages and shows its content directly.

2.3.3 Choosing Between an FBML and IFrame Canvas

Maver's study has pointed out the differences between FBML and IFrame canvas in six categories: Learning Curve, migration of existing applications, cross-platform portability, application design, application performance and testing (Maver et al. 2009).

Learning Curve

FBML pages require developers to master three things: FBML tags, FBJS, and the Facebook API as stated by Maver in his study, Although Facebook has provided the documentation for the FBML tags, application developers need to learn the nuances of each tag if they choose the FBML integration. Apart from the FBML tags, it also can be painful to figure out which subset works and how to get access to the application needs because of FBJS is a limited form of normal JavaScript. The Facebook API is used for both canvas types, although FBML can reduce the number of calls needed. IFrame pages' learning curve was limited to standard web technologies of cascading style sheets (CSS), HTML, JavaScript, and the Facebook API. Overall, the learning curve for an IFrame page is lower than for FBML pages (Maver et al. 2009).

Migration of Existing Applications

For software developers who already have the intent to deploy the application into Facebook, IFrames are easier than FBML integrations. Developers could reuse the existing HTML and JavaScript rendering, and use the Facebook API to get the data that the application would need. To convert applications to FBML requires software developers to rewrite JavaScript as FBJS. The application also needs to remove the dependence on any external JavaScript libraries. For example, YUI, jQuery etc. (Maver et al. 2009).

Cross-Platform Portability

If the application needs to support multiple platforms, IFrame is able to provide a common display model. The developers only need to change how the application gets the data for each platform. For FBML application, the FBML requires rewriting the FBJS into normal JS and reuse normal JS libraries (Maver et al. 2009).

Application design

Applications that have the Facebook look and feel design fit in well with the Facebook Platform and Facebook users expect applications to work like Facebook. Compared to IFrame applications, FBML provides easier method to do this because it gives applications access to the same controls and styles that Facebook itself uses. For IFrame applications, in this case with Facebook look and feel design, developers need to manually design the application in a way to fit in well in the Facebook Platform. Due to most of the Facebook tags don't have publicly downloadable equivalents; this can be time consuming for developers (Maver et al. 2009).

Performance

FBML applications can be faster than IFrame applications because they are able reduce the number of API calls needed by using FBML tags. However, in some specific functions provided by Facebook, the IFrame applications would perform better than FBML applications. For example, the chat bars function, which has to be created on each full page load. IFrame applications could avoid this because they only have to change the inner IFrame source instead of the entire page. Furthermore, when an FBML page does an AJAX call, it must pass through a Facebook proxy to parse any returned FBML. IFrame AJAX calls go directly to the callback URL. IFrame that use the Facebook JavaScript Client Library can receive results of preloaded FQL queries and get cached access to the user's friend list that

is not passed to the canvas page by default. This also helps IFrame catch up with some of FBML's speed advantages (Maver et al. 2009).

Testing
It is more difficult to test FBML applications because FBML must be rendered by Facebook before it is displayed. However, the developer could copy the expecting outcomes of the FBML to the FBML Test Console as well as get access to the application through Facebook. Compared to FBML testing, IFrame testing can be much simpler. Developers can access the application directly on the application local server. In these cases, developers might have to again emulate the GET variables that Facebook normally passes to IFrame pages to allow the application to function correctly(Maver et al. 2009).

3. Case Study

3.1Volvo IT and Commute Greener

This study was conducted at Volvo Group Telematics department of Volvo Information Technology, Goteborg, Sweden from Jan 2012 – June 2012. The study was conducted within the project Commute Greener, a Volvo IT solution for sustainable transport.

3.2 Commute Greener - A value-driven Investment towards sustainable development

The Commute Greener initiative is defined as follows:
Commute Greener provides a strong foundation to reach both your environmental targets and business objectives. The solution includes an application that gives you direct measurements and feedback through mobile devices, the web and social media channels. This makes it ideal for usage with customers, employees, or local community citizens. Through the power of Commute Greener, you contribute to sustainable development, save time and money as well as improve health – for your employees, customers or local community citizens (Commute Greener, 2012).
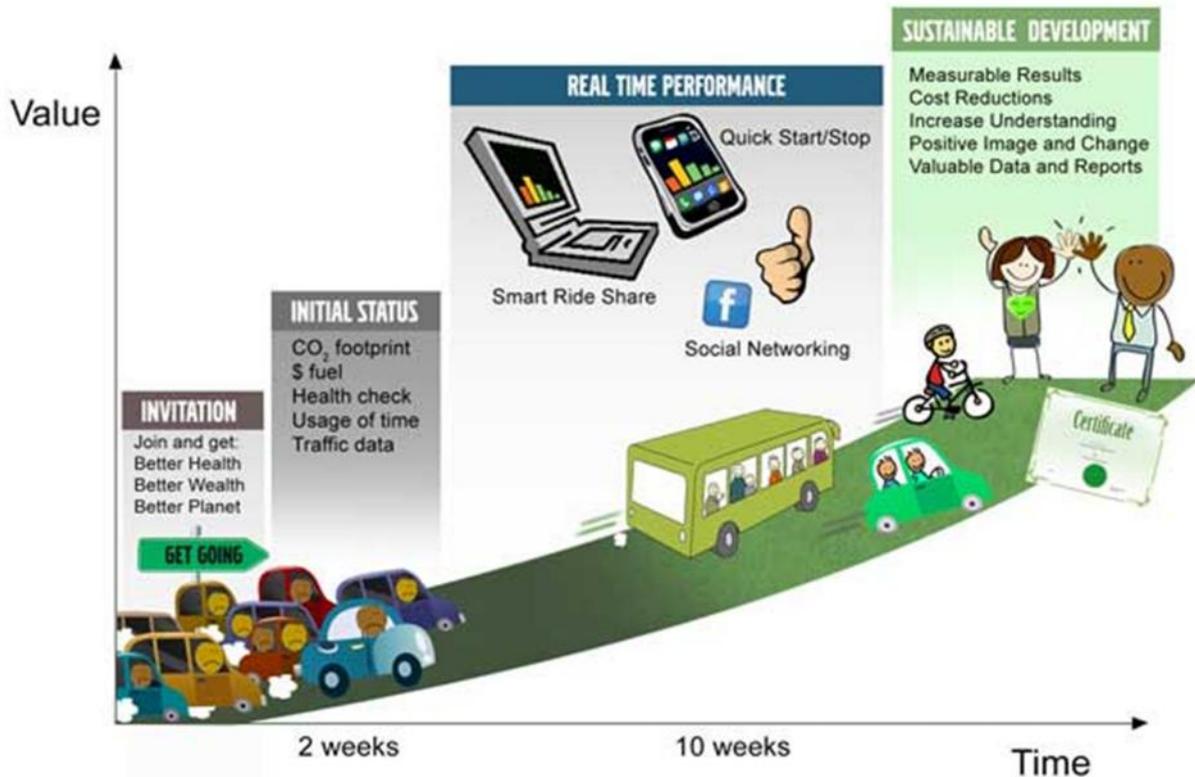
Figure 4: CommuteGreener Progress (Commute Greener, 2012)

Commute Greener started as an idea from a group of Volvo employees. Their vision was to develop an IT solution that would motivate people to do something good for the environment. In 2009 Volvo IT released this application to help employees measure the time, efficiency, and environmental impact of their commuting habits. By using this small application on mobile phones calculating the environmental impact of each commute, employees were able to reduce their CO_2 footprint by 30% in just one month's time. Later with support from key partners Commute Greener launched as a global service helping companies, cities, and individuals around the world understand a greener way of commuting. The main objectives are:

- *Actions to reach environmental and Corporate Social Responsibility targets*

Whether a small organization or a multi-national corporation, Commute Greener can deliver benefits that help deliver actions against environmental and corporate Social Responsibility targets.

- *Change commuting behavior and improve health!*

Instead of stressing around in a car it can often be possible to take a relaxing walk or a healthy bicycle trip.

- *Cost savings and efficiency for business*

Commute Greener offers value for the environment, as well as opportunities to reduce costs and increase efficiency for a number of corporate-wide initiatives (Commute Greener, 2012).

Today a lot is heard about intelligence transport systems when working with traffic and traveling information. These systems are based on open data resources example traffic information, commuting patterns, bus schedules etc. In 2011, Commute Greener opened its data through public APIs, aiming to help other applications reuse Commute Greener's existing features and focus on new feature developments. The first two open APIs were published in TrafikLab (TrafikLab, 2012). The first API was used to develop an application in the West Coast Travel Hack 2011 competition which is nominated as one of five nominees for Best Sustainability Project in Sweden at the CIO Awards 2012.

21

The other API called Commuting Pattern API was more complex. In this thesis, this API was used. We developed a Facebook application which read data from the above API to display commuting patterns of users. The details of the application are discussed in section 3.4.

3.3 New Trends and shift to Social networking site

Two and a half years ago Commute Greener was an SMS service. Users registered their commuting patterns from home to work and back called the baseline data and this data was stored in a database for comparison. The next release of Commute Greener was a web based application and an iPhone application. This web based application was kind of a standalone SNS with features like login, invite friends, chat, compete towards lower CO_2 emissions etc. Commute Greener was developed on a platform called Pocket Life (Pocket Life, 2012) which offers the above mentioned social features. Commute Greener aimed to take advantage of getting access to the entire existing user base of Pocket Life and all the other applications using the same platform and social features. Soon it was realized that with Facebook, Twitter, LinkedIn etc. offering a number of attractive and advanced social features, people would not like to create a separate community of friends in Commute Greener rather use one community which most of their friends are already using for example Facebook. Hence it was decided to embed Commute Greener in a SNS. Today the main features is to help people commute efficiently in an environment friendly way and ride share with trusted people while Facebook handles the social part like friends who have accepted this Commute Greener Facebook application get connected to each other as friends instead of inviting people. Having a Commute Greener Facebook application increases the possibility to accumulate data traffic and offer better transport solutions to transport companies etc. The aggregated data can be used to discover commuting patterns to develop and market new and enhanced products. The reason for choosing Facebook development platform according to Commute Greener is because they have the most mature platform APIs.

3.4 Development of CommuteGreenerIT

We developed our Facebook application CommuteGreenerIT using CommuteGreener's commuting pattern open API which is discussed in section 3.4.2. Even though the API was open to public, Commute Greener found it difficult to propagate it. The API response JSON files were complex and people without knowledge of JSON could not understand them. This led to our thesis application where we developed a java web application retrieving data from the API and displaying result on Facebook client side as per requirements. This was built using the IFrame integration method of Facebook. The data while developing the application was systematically collected and analyzed to provide answer to our research question.

3.4.1 CommuteGreenerIT

This application enables users to search commuting patterns based on parameters like locations, time interval, start and end positions, and region and transportation modes. The application calculates the CO_2 emissions for each journey according to the above parameters. These emissions can be compared to encourage users to improve their commuting patterns.

Another typical use could be a traffic organization planning to shut down a road on any day of a week. To gather information on which time or date the road of a particular location should be closed in order to create fewer problems to commuters, they can use this application to see the commuting patterns of people in the Google map along with their corresponding transport modes.

On the technical side, we used the gson library to fetch the API data and convert them into java objects and finally display them on the java server pages. We connected our application to Facebook via the canvas URL method of the Facebook platform. This helped us to reuse Facebook social features. Figure 5, 6 and 7 below display the working of the application CommuteGreenerIT in steps.
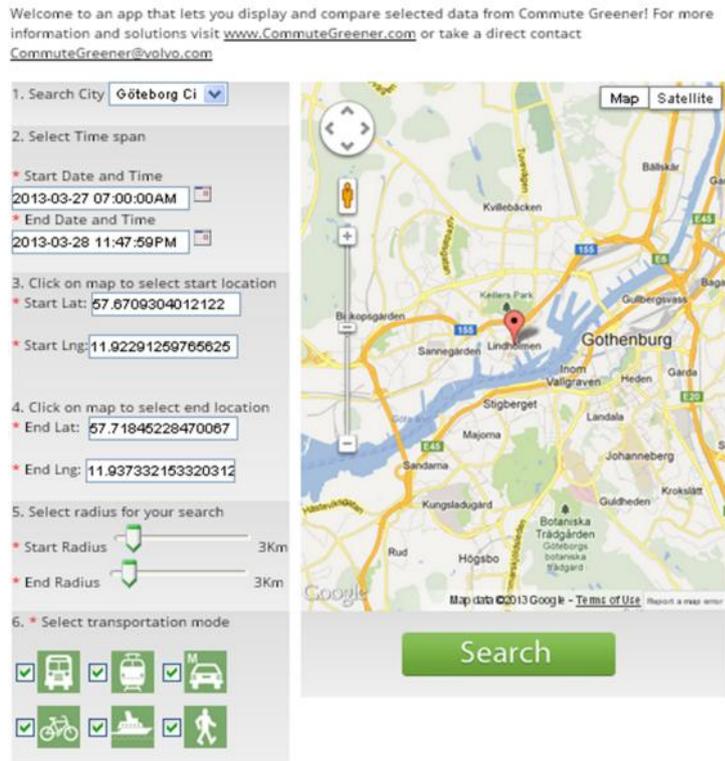
Figure 5: CommuteGreenerIT (Facebook Developer Platform, 2012)

When a user has access to the application, on clicking the canvas url in Facebook, this application opens. Here the Facebook user can select a city, time span, start latitude, start longitude, end latitude, end longitude of his journey, radius of search and transportation modes for searching a journey. The result of journeys retrieved from the API is displayed as markers on the Google maps color coded according to transport modes. A detailed list of the journeys and their respective CO_2 emissions is also displayed (Figure 6).

Figure 6: CommuteGreenerIT (Facebook Developer Platform, 2012)

Figure 6 shows how this result data can be compared to another set of input parameters.
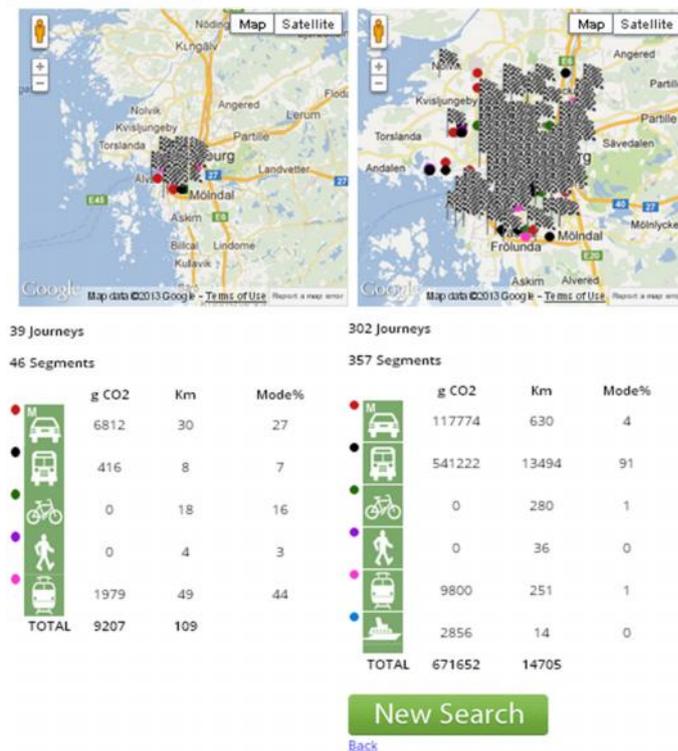
Figure 7: CommuteGreenerIT (Facebook Developer Platform, 2012)

In figure 7 we can compare the output from the two searches done in figure 5 and 6.

3.4.2 Commuting Pattern API

With this API one can get selected information about how commuters using Commute Greener are commuting within an area example within a region, city or suburb. Right now moment there is no versioning of the API, only a single version is being supported (TrafikLab, 2012).

Data Format

The table below shows the request message parameters, their types and if they are mandatory or optional. The startLat and starLng are the starting latitudes and longitudes of a journey. The startRadius is the radius around the startLat and startLng. Similarly the endLat and endLng are the end latitudes and longitudes of a journey around the endRadius radius. The request message Format parameter is HTML and the response message format is HTML and JSON. The default response message format is HTML, i.e. if the request message contains no format argument (TrafikLab, 2012).

Argument	Type	Comment
startLat	double	mandatory, start latitude
startLng	double	mandatory, start longitude
startRadius	int	mandatory, start radius (in meters)
endLat	double	mandatory, end latitude
endLng	double	mandatory, end longitude
endRadius	int	mandatory, end radius (in meters)
Format	string	default 'html', or json
Callback	string	optional callback function for JSON

Table 1: API Request Message Arguments (TrafikLab, 2012).

3.4.3 CommuteGreenerIT Configuration

Applications need to be configured in the Facebook Developer Platform. Figure 6 below shows the application configuration page of CommuteGreenerIT in Facebook. As each app in Facebook gets a unique App ID generated by Facebook, below we can see CommuteGreenerIT also has an App ID. Facebook users get access roles as administrators, developers, tester or insights user. In our case we had three administrator roles. CommuteGreenerIT is the IFrame app, the canvas URL and https secure canvas URL was configured. The height and width of the Canvas page URL can also be configured. For details of app settings one can read the documentation at Facebook Developers.
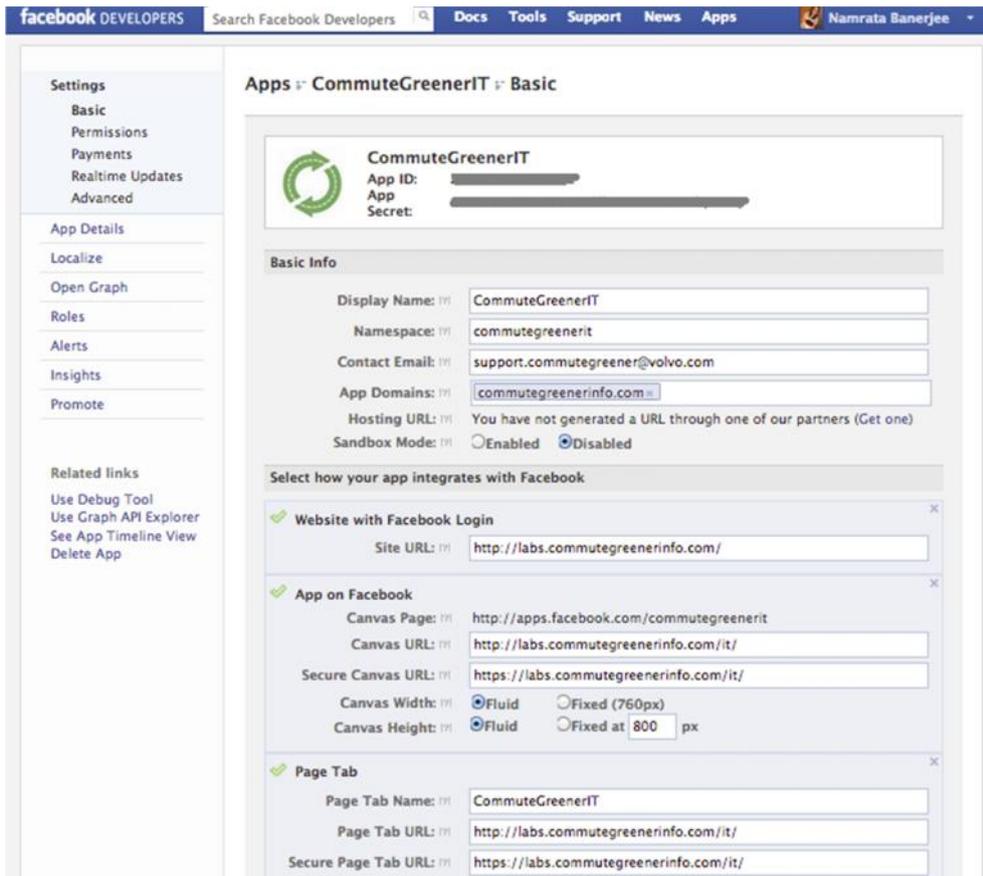
Figure 8: Facebook Developers App Setting Page for CommuteGreenerIT

Figure 9 illustrates the architecture of our application. Our application user sends request tokens to the platform. The platform server forwards user's request to our application server. The application server handles the user requests and sends back responses through the platform server. In this case, Facebook platform server is not able to modify the data from application side or change any response data to the application user.

26
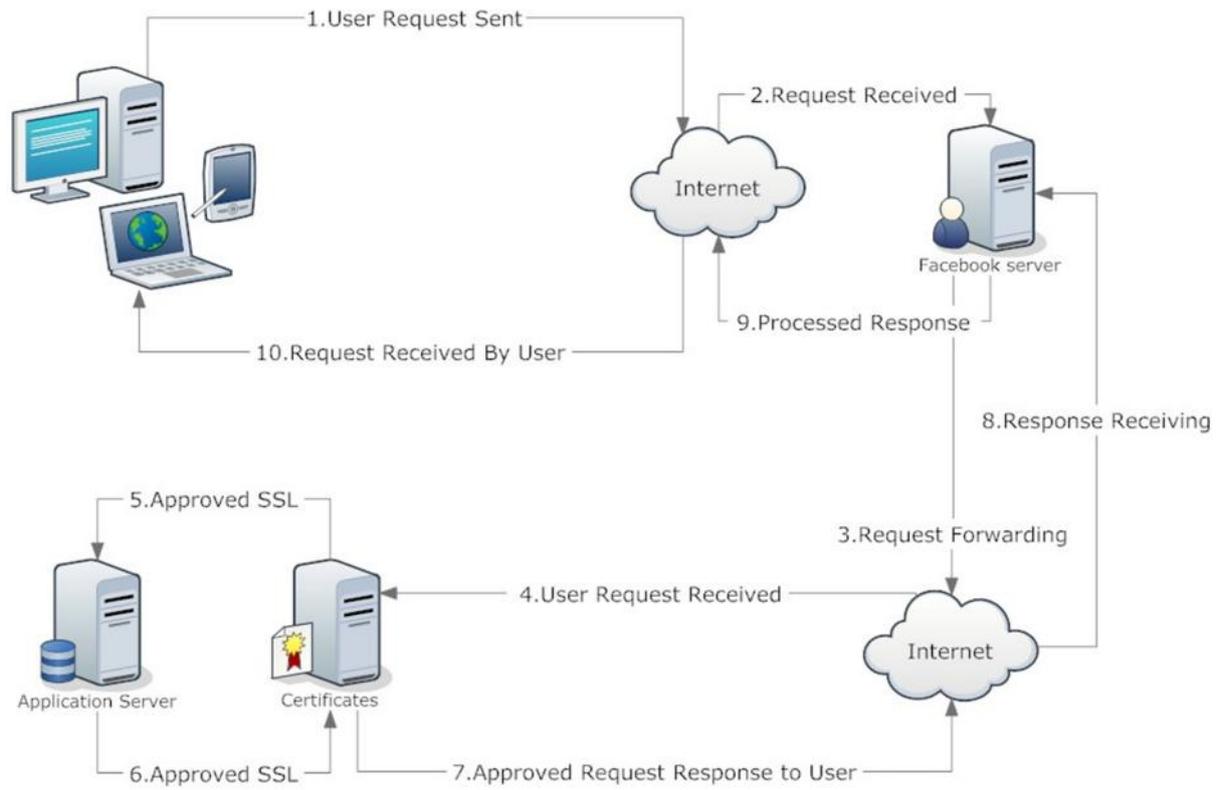
Figure 9: Facebook Developer Platform's Architecture Used by CommuteGreenIT

4. Method

Our research aims to investigate the benefits and risks of applications developed and integrated with social network platforms. In this research we have concentrated on Facebook. Our research question is: What are the benefits and risks for product managers, developers and testers in a software company to use social network platform services as application development platform? We have used design research method to accomplish our research.

4.1 Design Research

Design research is a set of analytical techniques and perspectives for perfo4rming research. It involves the analysis of the use and performance of designed artifacts to understand, explain and frequently to improve on the behavior of aspects of software (Vaishnaviet al., 2004). The process steps of using design research method are awareness of problem, suggestion, development, evaluation and conclusion.

We reviewed the study Essential Facebook Development: Build successful applications for the Facebook platform (Maver et al. 2009). However, this study mainly focus on build FBML based applications using Facebook platform and showing the benefits and risks related to this type of integration applications. In order to investigate what benefits and risks were involved in software development using social network platforms. We developed the application "CommuteGreenerIT" and deployed it into Facebook platform. The application was developed by using the IFrame service provided by Facebook platform which is different FBML based integration method. In this case, we identify the benefits and risks for IFrame integration between the application and social network platform. In order to widen and gather a holistic view of this integration method we also conducted interviews.

4.2 Data Collection

Application development process
The application development phase started from November 2011 and ended at March 2012. During the entire duration of the study we had access to Volvo Group Telematics environment for example laptops, networks, software etc. We could also access the intranet site as well as use required Volvo documents. For CommuteGreenerIT we received the requirements from the product owner. This was followed by an analysis of the requirements and the Facebook Developer Platform. During the development phase, 15 group meetings were held among software solution manger, product manager and two developers where we discussed and reflected on the application's requirements, architecture design, implementation and testing outcomes and how these were affected by the use of social network platform. During each meeting we took notes of our discussions on the challenges and successes we faced during the development cycle. From these notes we analyzed the benefits and risks of using IFrame integration method from a developer's perspective. The product manager and solution manager of Commute Greener were leading these weekly meeting.

Interview data
The data we gathered from application development meetings represent the thoughts mainly from application developer's perspective. We needed to further the research data to include product manager's perspective as well. We also need to evaluate our application development findings by interview other SNSs developers. This led us to conduct interviews within Volvo Group Telematics with people working with SNSs. We designed our interview questions based on the findings from our

development weekly meeting notes and from the reviewed literatures related with the Facebook Developer Platform. We chose to conduct semi-structured interviews as it enabled us to ask questions around our identified areas and motivated the interviewee to express their independent perspectives (Hennink et al., 2011). Our interview questions covered four areas: Present trend of social network platform based application development, advantages a company or a stand-alone developer would get from the platforms, problems of using social network platforms and the future of these platforms and applications developed on them.

In total six interviews were conducted with different roles inside the organization, one application developer with experience in both social network platform applications and traditional web application development, three software developers with limited experience of social network platform, one solution manager of Commute Greener, and finally one product manager of Commute Greener. To be able to get detailed description from our interviewees related to interview questions (Steinar, 1996), we used open questions and relied on follow-up questions. We used probing question when we needed help to understand a statement our interviewee had made (Boyce et al., 2006). All interviews were recorded and transcribed.

4.3 Data Analysis

We analyzed the collected meeting notes and grouped the data in three categories: 1) Benefits we gained by using IFrame integration method, 2) Problems we came across by using IFrame integration method, 3) and risks we were facing after the application deployment. For example, from the meetings notes analysis we could examine that we benefitted from using free social features provided by the platform. However, we also came across design conflicts between the application and the platform as well as the end user security concerns about third party involvement. Based on these three groups of data, we identified the benefits and risks related to social features, application design and security by using IFrame integration method.

Our coding phase for the interview data started after we have read and familiarized ourselves with the transcribed data. Firstly, we marked essential topics expressed by our interviewees as interesting units. Secondly, these units were gathered by specific category such as user base, social features, application design, maintenance and security. We worked systematically through the interview data set and identified interesting aspects in the data. For example, we found that both solution manager interviewee and product manager interviewee expressed their opinion about the large installed user group. We marked the user base as one interesting data unit for later discussion. After we listed the different codes that we have identified across the transcribed data, we started discussing the different codes and find benefits and risks related to these codes.

5. Results & Discussion

This chapter presents this research project's main results, namely the key themes determining the benefits and risks of using SNS as a software development platform. These key themes are identified by the analysis from the data collected from CommuteGreenerIT development process and interview data. All these results have been discussed and connected with our related literature review.

5.1 User Base

Benefits:
- Both FBML and IFrame applications could benefit from the large install social network platform user base.
- Users outside social network platform could get access to the IFrame application as well.
- It is easier for a software application to gather attention by using the installed user base than creating a new social community surround it.
- Social network users need less training to get familiar with the FBML and IFrame application.

One offered explanation to the huge popularity of SNSs is based on the fact that they have so many users using their services. As expressed by the solution manager interviewee: "I actually think it's mainly because people are not interested in developing separate communities, but they are willing to invite people to the one they use". The product manager also expressed that Volvo IT believes this huge installed user base of Facebook will be beneficial for the spread of the application. The product manager has repeatedly mentioned that the main reason for their venture into SNSs is its huge installed user base. They intend to target the existing Facebook users and not lose out on its massive user base. According to the solution manger and product manger's perspectives, it is easier to utilize and leverage an established user base instead of trying to make a new social network community surround the application with social features. The established user base also reduces user training because of the users are already accustomed to using those social features. The large installed user base can help application providers drawing more attention to their application. From end users' perspective, they are more likely to accept the application in a case they are already familiar with the social network. So for applications like CommuteGreenerIT, where the user community takes an important role, it is very easy to spread and popularize itself inside a social network community. As soon as we published our application on Facebook we could invite our existing friends on Facebook and ask them for feedback. Hence it was very easy to spread the application.

Risks:
- FBML applications are completely connected to the platform and cannot be separated.
- Users who do not register the social network platform cannot get access to the FBML application.
- Chain effects for both FBML application and IFrame application.
- Specific age range.

The installed user base can also have some negative influences on applications. As the developer and solution manager expressed that social network users are often tightly connected with friends, so there is a chain effect caused by these users. The fact that a user has stopped using an application can be easily found out by other connected users resulting in some related users losing interest in the application. Therefore there is an evident risk that if users stop using the social network platform, the application tightly coupled and deployed within the platform will be abandoned along with the platform. In addition end users in SNSs often belong to a specific age range. Company decision makers need to take into consideration parameters such as age, gender and location to understand

whether the specific user base would accept the application. The product manager mentioned that it is necessary to classify the users in order to identify the most relevant ones. The solution manager and product manager also pointed out that for services like Commute Greener, the spread of the application via SNSs would help Volvo in achieving its corporate social responsibilities.

5.2 Social features

Benefits:
- Social features provided by the development platform are reusable and free to use.
- FBML applications and IFrame applications have the same accessibility to the social features provided by the platform.
- New social features are continuously released by the platform.

Social features are key attributes of SNSs. The large user base of these SNSs as discussed above is fascinated by the digital social experience. In order to grasp and maintain this large user base, applications newly developed in these platforms also need to include social features. Today most of the SNSs are developed using open source software and they open up APIs providing open source social features. The open source availability of these features makes them really attractive to interested companies as expressed by the developer interviewee: "We get a lot of social features for free. We could focus on core business value of the application instead of being concerned about the social features". Our application CommuteGreenerIT uses Facebook's authentication module for single sign on i.e. users log in to the application via Facebook login eliminating the need of a separate login module. This is one of the privacy protections provided by SNSs. We also found that a lot of developing time and effort can be saved by using social features from Facebook. The benefits from using social features showed in our study holds the same opinion with Juste et al. (2008). The availability of open source social features is a big reason to build more and more applications on these platforms. Social features provided by these platforms are also beneficial for developers in the software development process.

The product manager expressed that the platform can be utilized to build great social experience with a number of available social features. From a developer's perspective, they can reuse code of social features and therefore reduce the developing time. This was also experienced in the development of our application CommuteGreenerIT, in which we saved significant time by reusing the existing social features. And all our interviewees agreed that people using Facebook's social features and connecting with Facebook page have respectively significant impacts on consumers' attitude, they also shared the same opinion with the study of Suraworachet et al. (2012) that exploring social features is a promising direction to solve similar problems in this domain.

Risks:
- It is difficult to modify the features provided by the platform for both FBML and IFrame applications.
- Lock-in effect.

Although features provided can be easily implemented by developers, modifications or improvements are very difficult to implement, posing limitations to applications developed in the platforms. As pointed out by the solution manager during the interview: "the applications actually depend on other's roadmap". In other words, the social features provided by the platforms can be used easily but they cannot be modified easily. As the experienced developer interviewee has mentioned, individual developers are small players within these platforms, which is different from traditional web applications where each developer has greater control of the application. Our interviewee also stated

that there exist lock-in effects for different social network platforms. Each platform has its own set of special features, and these features are not widely shared among the platforms. In order to deploy an application in multiple social network platforms, developers need to spend extra time on integrating the application with the specific set of social features. Lastly, it is also brought to attention that the online documentation of social features sometimes is not easy to understand by developers. Different integrations between applications and platforms give the developers different social features to choose from. However, as confirmed by all our interviewee, social features are definitely the core of the social experience of SNSs. However for both FBML and IFrame, product managers, developers and testers should be aware of the limitations of customization.

5.3 Application design

Benefits:
- Serves as templates both FBML and IFrame applications.
- Developers choosing FBML integration method can easily adopt the platform's design.
- IFrame application's design is more under control of developers.

One developer interviewee stated that the positive side of the platform is that they behaving like templates for gathering application design requirements thereby reducing efforts on application design specification.

Risks:
- Both FBML and IFrame applications are restricted by the platform API.

Developers as well as product manager interviewee stated that application design using SNS platform is limited by the platform API. Application design needs to be created according to the social network platform properties. Our experience from developing the application CommuteGreenerIT confirmed this. The first deployment of the CommuteGreenerIT application encountered several design flaws which forced us to redesign the application. While we developed CommuteGreenerIT, we observed that though the application design was easier compared to developing it outside the platform, there was always a need to abide by the design restrictions of the platform. Since these applications reside within Facebook's user interface, they are restricted in the user interface design and aesthetic experience as well. However, Vanderdonckt et al. (2009) in their working paper presented a model-based approach for developing user interfaces to an interactive web application. It exhibits the capability of being rescaled in any dimension without losing the application information. The interface under this model-based approach is beneficial for the developers and end users, because it facilitates platform adaptation and makes the web application accessible in a ubiquitous way. All of our developer interviewees agreed that IFrame applications may face less design constraints compared with FBML applications.

As the solution manager expressed during his interview, Facebook users are accustomed to rapidly clicking around Facebook and switching from one page to the other. So a similar user experience needs be maintained for IFrame applications like CommuteGreenerIT. Maver et al. 2009 explained in their study, FBML applications are having faster loading speed than IFrame applications. To give the end users a smooth experience while using our application on Facebook we had to make the required architecture changes. This confirms Dmitriy's study which states that the application designed for social network platforms may impose certain restrictions on the platform therefore cause problems when adding new functional features while running the application. It also happened to us during our development phase that we had to mandatorily enable Secure Socket Layer (SSL) certificates for our application because of Facebook platform's security requirement update. And this newly added

security requirement update affects both FBML and IFrame applications. Thus designing an application for Facebook requires these constraints to be taken into consideration.

5.4 Testing

Risks:

- Immature and evolving as development platform itself.
- Extra testing work for IFrame applications outside the platform.
- It is not easy for testers to create new test account through social network platform for both FBML and IFrame applications.

The product manager expressed that the development platform was recently released by the SNSs, so there is no mature testing environment constructed. Developer interviewee also mentioned that internal application functions have to be tested separately by developers without deployment. This would increase the risk of creating more uncovered bugs before final release. This specific result hold the same opinion as the study of Träskbäck et al. (2003) about software testing for third party involved environments may also be more complex as it requires more technical specifications for specific environment. In addition, it is not easy to create test accounts to accomplish the integration test after the deployment because of the platform will take away accounts they believe which are not real users. Maver et al. 2009 also expressed this testing problem for FBML applications. And so far there is no complete solution to overcome this limitation. However, the solution manager described that during development phase developers could release the application within some connected friends or targeted users. These users could use the application before the entire installed user base of the platform. Due to that, the acceptance tests are controlled by developers and they could benefits from this type of internal release. Testers can concentrate on testing the core application features.

Our study holds a similar view as Stanton (2008) about testing challenges. As the product manager expressed for the testing topic, the application's performance relies on the platform. It is difficult to validate the performance individually without deploying the application into the social network platform. On the other hand, the application deployment process would again increase the testing difficulty. The large installed user base makes it more complex since people are using different combination of operating system, browsers, security revisions, etc. Developer interviewee has pointed out that for IFrame integrated applications, they are difficult to test in an automated way because the platform prohibits from hooking up automation systems to their servers. In other words, one cannot just replace the browser accessing the platform server with any load test tool. However, Alshahwan has provided one solution for developers to improve the application testing by using testability measures (Alshahwan et al., 2009). For the IFrame integrated applications, developers are able to test the application before deploying it into the platform. In this case, developers could use an automated test code generation method to test the application. And the oriented approach provided by Turner improves the tests of the stand-alone application before deployment (Turner et al., 2008).

5.5 Analytics

Benefits:

- Analytic functions provided by the platform are free to integrate for both FBML and IFrame applications.
- Analytic functions are attached directly with the application and no extra integrating needed.
- New analytic functions are continuously released by the platform connected with social features.

Apart from those free social features provided by social network platform, the product manager and the solution manager mentioned that there are analytic functions which enable developers as well as product managers to track how end users are interacting with the application. From application owners' perspective, firstly they don't need to spend extra costs on developing specific analytic tool to keep track of their applications. This result hold the same opinion as Lakhwinder's study which explained that web analytics has become an important component of web based system and helps in taking business decisions. By the help of some analytic functions, for example, the feedback for stream stories, referral traffic to the application, Application owners could have a clear view of different circumstances the application is facing. Besides that, it included a breakdown of what users actions contribute to active user count, demographics on authorized users and active users, and the number of times permissions are prompted and granted.

As the developer interviewee expressed that based on these data provided by the analytics, developers could improve their application easily to meet end users' need. These functions could also help those application owners to analysis the application easily even they are not professional in this domain. This result confirms to Bukhari's study that social media analytics can be of great value for organizations in both special events and routine activities (Bukhari et al., 2012). In addition, it also provides diagnostics for the application so that developers can easily track API errors as well as get access to allocations and throttling information. The analytics feature provided by the platform is helpful for both FBML and IFrame applications owners.

Risks:
- The privacy of the analytic data.
- Stability of the analytic data.

All interviewees have pointed out that the analytic data are created by the platform which means the platform actually owns the analytic data. The application owner could not ensure the privacy of the data. Besides this, the Facebook internal analytic tools may miss recording data as well. Many applications' statistics were missing for days in between when developers acquired daily usage activity data for the top 200 applications on Facebook (Nazir et al., 2008). So application owners have to address the risks while using free analytic functions provided by the platform.

5.6 Maintenance

Risks:
- Constant updates from the platform side.
- Maintenance with multiple developers

The maintenance of the application consists of two parts while using social network platform. The first part is the maintenance of the application itself. For example, bug fix, new features, additional components, new version release, etc. The second part is the maintenance according to the social network platform. As stated by the product manager and developers in their interviews, the API provided the platform is not steady enough. New updates are constantly added to the platform. So in order to keep the application alive in the platform, developers need to work with every update which might affect the application performance. This result has been confirmed by Ashish's study about companies increasingly face maintenance issues related to social media. And companies should try to address potential problems before they arise. Otherwise, they may face expensive discovery problems (Ashish, 2012).

The developer interviewee also expressed that the application must follow the rules set up by the platform. The platform would not roll back the update to keep ensure application running. Application, in this case, is small player which has to catch up the big player's step. Developers are only able to

know what update components will stop the application working after the platform actually deploy the update. So the application sometimes is not available during the update fix. This increases the maintenance difficulty as well as the costs. Beside this, maintenance the Facebook application with multiple developers is also complicated, because it is difficult to verify local code changes without pushing to the live server as well as application deployed into the platform.

Based on our development process data and interview results, we found that application developers and companies need to do extensive work with the maintenance of SNS based applications. In case of SNSs the APIs developed for the platforms are not yet steady enough; developers need to work with every update which might affect the application. This has been approved by Kajko- Mattsson's study which shows the organization must create a new role responsible for evolving and maintaining high-level web services. Apart from that, the maintenance process has to be slightly adapted to manage investigation of problems and impact analysis which have to be made between applications and the involved platforms (Kajko-Mattsson, 2004).

All of our interviewees have claimed that maintenance is one of the major risks for application deployed in social network platforms, especially for FBML applications. IFrame applications are relatively easier to maintain because the application is just embedded as IFrame, making changes to it is easier.

5.7 Security and Confidentiality

Benefits:
- Secured data transfer between the application and the platform.
- Applications update notifications for application administrators.
- Whitelist of IP addresses for updating application settings.

We found out that there are two different operations happening when users are using applications provided Facebook during our application development phase. Firstly, Facebook is able to process and strip undesirable content coming from the application server before forwarding them to end users. This actually enables Facebook to protect users from dangerous content that may be embedded in the response data by the application servers. Facebook developer platform also introduced the Secure Canvas URL and Secure Tab URL fields for developers to serve their apps through an HTTPS connection. This ensured that users browsing IFrame applications browse over a secure connection. Secondly, the developer interviewee mentioned that the application servers have the right to not transmit data back to Facebook API server. In this case, the data from application server are actually only viewable by Facebook API. The Facebook API platform will not have the ability to store as well as modify the data. These two data traffic filtrations enhanced the security for end users as well as the platform. However, when dealing with sensitive corporate data, there should be clear corporate policies to define what kind of information employees can share in these public online platforms via the developed applications.

FBML and IFrame application developed for the platform configured to have different role; serve whitelists of IP addresses, and have update notifications for application settings, developers can set up IP address whitelists for making requests or setting changes to their applications, and use an access control model to establish which privileges each of their team members has though there exists considerable risks (Facebook developer platform 2012).

Risks:

- Leakage of the application data outside corporate circle.

Regarding confidentiality of data within these platforms, people outside the corporate circle may get access to various discussions which can be dangerous for any organization. So there still remains a need to have internal SNSs within the corporate circle.

5.8 Brand Reputation

Benefits:

- Promotes easy and effective branding.

The product manager expressed that applications deployed in the social network platform benefit from the social network brand. The loyal installed users under the brand directly increase the potential users of the application. And the application could be promoted by the platform. From the product owner's perspective, he believes that the SNSs are still growing. He has the vision that his business applications deployed in the social network platforms would benefit from the social network branding.

Risks:

- Dependence on SNS brand reputation.

The solution manager has mentioned" the other side of the social network brand is attached as well: risk." This has been supported by Pekka's study. Pekka described that there are connections between reputation risk and SNS as a business environment. He also described that social media expands the spectrum of reputation risks, which must be considered in the modern business and marketing environment (Pekka, 2010). Our interviewees have expressed their similar concerns about SNS brand reputation effects. If end users start to lose interests on the SNSs, the benefits would be decreased simultaneously. This brand effect exists in all the three integration types of applications.

7. Conclusion

The intention of this thesis is to investigate the benefits and risks of developing applications on social network services for product managers, developers and testers in a software company.

We used Facebook as SNSs and developed an application for Commute Greener (Volvo IT) using its open API for traffic data using the Facebook Developer Platform. The benefits and risks we gathered by launching the application using the platform represents a valuable resource since it provided us with first-hand data about the benefits and risks of such applications launched on an SNS. Further semi-structured interviews were conducted targeting product owners and developers at Volvo IT to widen our research data. Based on the data collected from our development process and interviews, eight key themes where identified: user base, social features, application design, testing, analytics, maintenance, security & confidentiality and brand reputation, which could determine the benefits, risks and limitations of such social network based applications. The awareness on the usage of SNSs as development platform can be raised by providing the knowledge of the benefits and risks to software development teams. Companies should take advantages of the large user base and integrate the social features to provide enhanced social experiences for end users. Meanwhile, application developers need to take into consideration the risks of security, testing and maintenance.

More case research is needed to enhance the answer for the research question. The conclusion of this research provides a preliminary guideline about the benefits and risks of having corporate solutions on SNSs. More empirical data analysis needs to be explored to support the findings. In this study we focused on the benefits and risks of applications built on SNSs, this research method could be extended to a large number of topics like SNSs and product development, SNSs based application maintenance, analytics and popularity of SNSs based applications etc.

References

Acquisti, A., & Gross, R. (2006). Imagined communities: Awareness, information sharing, and privacy on the Facebook. Proceedings of 6th Workshop on Privacy Enhancing Technologies (pp. 36-58). Cambridge, UK: Robinson College, 2006.

Adamic, L. A., Büyükkökten, O., & Adar, E. (2003). A social network caught in the Web, Found at website, http://www.firstmonday.org/issues/issue8_6/adamic/index.html

Ahlqvist, T., Bäck, A., Halonen, M.,&Heinonen, S.(2008),"Social media roadmaps exploring the futures triggered by social media", VTT Tiedotteita - ValtionTeknillinenTutkimuskeskus (2454): 13.

Ahmed, J., & Shaikh, Z.A.(2011), " Privacy issues in social networking platforms: comparative study of facebook developers platform and opensocial", Proceedings of International Conference on Computer Networks and Information Technology (ICCNIT), Abbottabad, 2011.

Alshahwan, N., Harman, M., Marchetto, A., &Tonella, P. (2009), "Improving Web Application Testing using testability measures", Proceedings of the 11th IEEE International Symposium on Web Systems Evolution, Edmonton, AB, 2009.

Ashish, S. P. (2012), "Cloud Computing And Social Media: Electronic Discovery Considerations And Best Practices", The Metropolitan Corporate Counsel, 2012.

Bernal, J. (2009), Web 2.0 and Social Networking for the Enterprise: Guidelines and Examples for Implementation and Management Within Your Organization, 800 East 96th Street, Indianapolis, Indiana, IBM Press.

Boyce, C., & Neale, P. (2006), Conducting in Depth Interviews: A Guide for Designing and Conducting In-Depth Interviews for Evaluation Input, 9 Galen Street, Suite 217 Watertown, USA, Pathfinder International.

Boyd, M., Ellison, B., Journal of Computer-Mediated Communication Volume 13, Issue 1, pages 210–230, October 2007, Article first published online: 17 DEC 2007, DOI: 10.1111/j.1083-6101.2007.00393.x

Bukhari, I., Wojtalewicz, C., Vorvoreanu, M., & Dietz, J. E. (2012), "Social media use for large event management: The application of social media analytic tools for the Super Bowl XLVI", Proceedings of IEEE Conference on Technologies for Homeland Security (HST), Waltham, MA, USA, 2012.

Canvas URL (2012), found at website, https://developers.facebook.com/docs/appsonfacebook/tutorial/ [accessed Sept 20th, 2012].

Colin, S., Qaalfa, D., & Steven, W. (2010), Customer Experience: Future Trends and Insights, Chapter 6, the human social media experience, 175Fifth Avenue, New York, USA, Palgrave Macmillan.

Commute Greener (2012), found at website, http://commutegreenerinfo.com/?page_id=86 [accessed Oct 10th, 2012].

Commuting pattern API (2012), found at website, http://www.trafiklab.se/api/commuting-pattern [accessed Oct 10th, 2012].

David, B. (2008), International Communication Association, Journal of Computer-Mediated Communication, Article first published online: 11 FEB 2008, DOI: 10.1111/j.1083-6101.2008.00408.x,Volume 13, Issue 2, pages 516–529, January 2008.

Di, L.G.A., Fasolino, A.R., Faralli, F., & De C.U. (2002), "Testing Web applications", IEEE Conference Publications, Proceedings of International Conference On Software Maintenance ICSM, 2002.

Dmitriy, K. (2010), "Developing Facebook Applications: Tips for Owners and Designers", found at website, http://www.acceptic.com/articles-and-reviews/facebook-application-development-tips.html [accessed Dec 4th, 2010].

Facebook (2004), found at website, http://en.wikipedia.org/wiki/Facebook [accessed May 15th, 2012].

Facebook Authentication (2012), https://developers.facebook.com/docs/authentication/ [accessed May 15th, 2012].

Facebook Canvas (2012), found at website, https://developers.facebook.com/docs/guides/canvas/ [accessed May 15th, 2012].

Facebook Developer (2012), found at website, https://developers.facebook.com/docs/reference/api/ [accessed May 15th, 2012].

Facebook Developer Platform (2012), found at website, http://developer.facebook.com/ [accessed May 15th, 2012].

Facebook Platform Building Blocks (2012), found at web site, http://pluralsight.com/training/courses/TableOfContents?courseName=fb-platform/ [accessed Nov 16th, 2012].

Facebook Statistics (2012), found at website, http://developer.facebook.com/ [accessed May 15th, 2012].

FacebookOpen Graph (2012), found at website, https://developers.facebook.com/docs/opengraph/ [accessed May 15th, 2012].

Friendster (2000), found at website, http://en.wikipedia.org/wiki/Friendster [accessed May 15th, 2012].

Gajjala, R. (2007). Shifting frames: Race, ethnicity, and intercultural communication in online social networking and virtual work. In M. B. Hinner (Ed.), The Role of Communication in Business Transactions and Relationships (pp. 257-276). New York: Peter Lang.

Geidner, N. W., Flook, C. A., & Bell, M. W. (2007, April). Masculinity and online social networks: Male self-identification on Facebook.com. Paper presented at Eastern Communication Association 98th Annual Meeting, Providence, RI.

Geocities (1994), found at website, http://en.wikipedia.org/wiki/GeoCities [accessed May 10th, 2012].

George, A. (2006, September 18). Living online: The end of privacy? *New Scientist, 2569.* Retrieved August 29, 2007 from http://www.newscientist.com/channel/tech/mg19125691.700-living-online-the-end-of-privacy.html

Golder, S., Wilkinson, D., &Huberman, B.A. (2007), "Rhythms of Social Interaction: Messaging within a Massive Online Network", Proceedings of 3rd International Conference on Communities and Technologies, Michigan USA, 2007.

Granovetter, M. (1973), "The strength of weak ties", American Journal of Sociology, Volume 78, Issue 6, 1973.

Gson (2012), found at website, http://code.google.com/p/google-gson/ [accessed May 10th, 2012].

Harvard Business Review (2012), found at website, http://hbr.org/web/slideshows/social-media-what-most-companies-dont-know/3-slide [accessed Dec 10th, 2012].

Hennink, M., Hutter, I., & Bailey, A. (2011), Qualitative research methods, 55 City Road, London, SAGE publications Ltd.

Hossmann, T., Legendre, F., Nomikos, G., &Spyropoulos, T. (2011), "Stumbl: Using Facebook to collect rich datasets for opportunistic networking research", Proceedings of IEEE International Symposium on World of Wireless, Mobile and Multimedia Networks (WoWMoM), Lucca, 2011.

Jagatic, T., Johnson, N., Jakobsson, M., & Menczer, F. (2007). Social phishing. Communications of the ACM, 5 (10), 94-100.

Juste, P.S., Wolinsky, D., Jiangyan X., Covington, M., &Figueiredo, R.(2008),"On the Use of Social Networking Groups for Automatic Configuration of Virtual Grid Environments", Proceedings of Conference on Grid Computing Environments Workshop, Austin, TX, USA, 2008.

Kajko-Mattsson, M. (2004), "Evolution and maintenance of web service applications", Proceedings of 20th IEEE International Conference on Software Maintenance, Chicago Illinois, USA, 2004.

Kaplan, M., Haenlein, M. (2010), "Users of the world, unite! The challenges and opportunities of social media", Business Horizons, Vol. 53, Issue 1 (page 61)

Krishnamurthy, B.(2009), "A Measure of Online Social Networks Paper", Proceedings of the 1st International conference on Communication Systems And Networks, Bangalore, 2009.

Lakhwinder, K., Hardeep, S., &Ramandeep, K. (2012), "Web analytics and metrics: a survey", Proceedings of the International Conference on Advances in Computing, Communications and Informatics, Chennai, India, 2012.

Lash, S. (2007). Power after Hegemony: Cultural Studies in Mutation. Theory, Culture & Society, 24(3), 55–78.CrossRef, PubMed,Web of Science®

Maver. J., & Popp C. (2009), Essential Facebook Development: Build Successful Applications for the Facebook Platform, ADDISON-WESLEY, 1st Edition.

Maximilien E.M, T. Grandison, T. Sun, D. Richardson, S. Guo, & K. Liu, (2009)"Enabling Privacy as a Fundamental Construct for Social Networks" Proceedings of IEEE conference on Social Computing, Vancouver, BC,2009.

Meeyoung, C., Haewoon, K., Rodriguez, P., Yong-Yeol, A., &Moon, S. (2007), "I Tube, You Tube, Everybody Tubes: Analyzing the World's Largest User Generated Content Video System", Proceedings of the 7th ACM SIGCOMM conference on Internet measurement, San Diego, CA, USA, 2007.

Milgram, S. (1967), "The small world problem", Psychology Today.Sociometry, Vol. 32, No. 4.

Min-Feng, W., Sie-Long, J., Meng-Feng, T., & Cheng-Hsien, T. (2011), "Enterprise Email Classification Based on Social Network Features", International Conference on Advances in Social Networks Analysis and Mining (ASONAM), Kaohsiung, Taiwan, 2011

Mislove, A., Marcon, M., Gummadi, K., Druschel, P., &Bhattacharjee, B. (2007), "Measurement and Analysis of Online Social Networks", Proceedings of Internet Measurement Conference, San Diego, CA, USA, 2007.

Muniz, A. M., &O'Guinn, T. C. (2001), "Brand community", Journalof Consumer Research, 27(4), 412—432.

MySpace (2003), found at website, http://en.wikipedia.org/wiki/Myspace [accessed Oct 10th, 2012].

Nazir, A., Raza, S., &Chuah, C. (2008), "Unveiling Facebook: A Measurement Study of Social Network Based Applications", Proceedings of the 8th ACM SIGCOMM conference on Internet measurement. Vouliagmeni, Greece, 2008.

O'Reilly, T. (2007), "What Is Web 2.0: Design Patterns and Business Models for the Next Generation of Software", International ", Proceedings of the International Journal of Digital Economics, No. 65, pp. 17-37.

Pekka, A. (2010), "Social media, reputation risk and ambient publicity management", Journal of Strategy & Leadership, Vol 38 Iss: 6, pp.43-49.

Plural Sight (2012), found at website, http://pluralsight.com/training/ [accessed Oct 10th, 2012].

Pocket Life (2010), found at website, http://www.pocketlife.com/#Login-1220 [accessed Oct 10th, 2012].

PocketWeb (2012), found at website, http://www.pocketweb.com/platform.html [accessed Oct 10th, 2012].

Quinn, D., Liming, C., &Mulvenna, M.(2011), " Does Age Make a Difference in the Behaviour of Online Social Network Users?", Proceedings of 4th International Conference on Cyber, Physical and Social Computing Internet of Things (iThings/CPSCom), DaLian, China, 2011.

Salvatore, P., Guinan, P. J., & Weinberg, B.D. (2008), "The Secrets of Marketing in a Web 2.0 World", The Wall Street Journal, Europe Edition, 2008.

Stanton, C. (2008), "Thoughts on Testing Social Networking Apps. Software Testing Trends", http://blog.utest.com/thoughts-on-testing-social-networking-apps/2008/10/ [accessed Sept 10th, 2012].

Steinar, K. (1996), An Introduction to Qualitative Research Interviewing, Thousand Oaks, California, SAGE Publications Inc.

Suraworachet, W., Premsiri, S., &Cooharojananone, N. (2012). "The Study on the Effect of Facebook's Social Network Features toward Intention to Buy on F-commerce in Thailand", Proceedings of IEEE/IPSJ 12th International Symposium on Applications and the Internet (SAINT), Izmir, 2012.

Todd, B. (2006), "Social software in Academia, Educause Quarterly Magazine", volume 29, number2, found at website, http://www.educause.edu/ero/article/social-software-academia [accessed May 15th, 2012].

TrafikLab (2012), found at website, http://www.trafiklab.se/api [accessed Oct 10th, 2012].

Träskbäck, M., Koskinen, T., &Nieminen, M. (2003), "User- Centred Evaluation Criteria for Mixed Reality Authoring Application", Proceedings of HCI International, Crete, Greece, 2003.

Tripod.com (1995), found at website, http://en.wikipedia.org/wiki/Tripod.com [accessed Oct 10th, 2012].

Turner, D.A., Park, M., Kim, J., &Chae, J. (2008), "An Automated Test Code Generation Method for Web Applications using Activity Oriented Approach", IEEE/ACM International Conference on Automated Software Engineering, L'Aquila, 2008.

Twitter (2006), found at website, http://en.wikipedia.org/wiki/Twitter [accessed Oct 10th, 2012].

Vaishnavi, V., &Kuechler, W. (2004), "Design Science Research in Information Systems" January 20, 2004, last updated September 30, 2011. URL: http://desrist.org/desrist [accessed Oct 12th, 2012].

Vanderdonckt, J., Guerrero-Garcia, J., & Gonzalez-Calleros, J.M. (2009), "A Model-Based Approach for Developing Vectorial User Interfaces", Proceedings of Web Congress Latin American, Merida, Yucatan, 2009.

Wikipedia a, found at website, http://en.wikipedia.org/wiki/Social_graph [accessed May 19th, 2012].

Wikipedia b, found at website, http://en.wikipedia.org/wiki/Mark_Zuckerberg [accessed May 19th, 2012].

Wikipedia d, found at website, http://en.wikipedia.org/wiki/Facebook [accessed May 19th, 2012].

Wikipedia e, found at website, http://en.wikipedia.org/wiki/Facebook_platform [accessed May 19th, 2012].

Wikipedia f, found at website, http://en.wikipedia.org/wiki/Web_analytics [accessed Dec 11th, 2012].

Zabin, J., & Jefferies, A. (2008), "Social media monitoring and analysis: Generating consumer insights from online conversation", Aberdeen Group Benchmark Report, 2008.