



LAUREA
UNIVERSITY OF APPLIED SCIENCES
Together we are stronger

Integrating external authentication service in Viope Solutions Oy's web platform

Malla Pradhananga, Rabi Raj

2015 Leppävaara



Laurea University of Applied Sciences
Laurea Leppävaara

Integrating external authentication service in Viope Solutions Oy's web platform

Rabi Raj Malla Pradhananga
Degree Program in Business
Information Technology
Bachelor's Thesis
April, 2015

Rabi Raj Malla Pradhananga

Integrating external authentication service in Viope Solutions Oy's web platform

Year 2015 Pages 53

In today's world, surfing the internet and using various web services has become commonplace. Most of the web services require users to create an account in order to use their services. Studies show that an average internet user has at least 25 different web accounts. This means the user has 25 different set of username and password combinations to remember in order to access those accounts. This could lead to the problems like forgetting username or password or both and password mismatch. External authentication service, EAS, solve this problem at certain extent by letting users to use their existing set of username and password for more than one web accounts.

This thesis work has aimed to integrate EAS in the case company, Viope Solutions Oy's web platform. There are different types of EAS providers available from social networking sites to payment solution services. It was essential to choose the most suitable EAS provider with which Viope's potential users already have account and they feel comfortable using it. Different existing researches and survey reports were analyzed and Facebook was chosen as the most suitable EAS provider.

A development environment was created to test the implementation of EAS. Following recommendations and best practices from Facebook's documentation and other sources, EAS was successfully integrated in the test website. But the actual implementation in Viope's platform will be handled by the company itself.

The company CTO is satisfied with the result of this thesis work. The thesis work has created a basic implementation of Facebook login which mainly focused on the back end portion. The front end portion and required modification in code will be done by the company before implementing it in the live site.

External authentication service, Social login, Facebook login, OAuth

Table of contents

1	Introduction	6
1.1	Background of case company	6
1.2	Problem statement	7
1.3	Objectives	7
1.4	Scope and limitations.....	8
2	Research method.....	8
3	External authentication service (EAS).....	9
3.1	Benefits of EAS	10
3.1.1	Benefits of EAS to users	10
3.1.2	Benefits of EAS to companies	11
3.1.3	Security benefits of EAS.....	12
3.2	Pitfalls of EAS	13
3.2.1	Privacy concern of user	13
3.2.2	Misuse of user data.....	13
3.2.3	Dependence on EAS provider's availability	14
3.2.4	Security flaws	14
4	Concepts and Technologies used	14
4.1	Authentication	15
4.2	Authorization	15
4.3	OAuth.....	15
4.4	Perl	16
4.5	Perl Dancer	17
4.6	Nginx	17
4.7	PostgreSQL	18
4.8	Git.....	18
4.9	Debian Wheezy	19
4.10	HTML	19
5	List of EAS providers.....	19
5.1	Social media providers	19
5.2	Non-social media providers.....	21
6	Selection of EAS provider	23
7	Implementation of EAS.....	26
7.1	Social login best practices	26
7.2	Creating development environment	27
7.2.1	Installation of required technologies and tools	27
7.2.2	Creating simple Dancer app	28
7.2.3	Setting up Nginx	29

7.2.4	Setting up Git	29
7.2.5	Setting up database.....	30
7.2.6	Creating test website.....	31
7.3	Integrating Facebook Login.....	34
7.3.1	Log in with Facebook	38
7.3.2	Sign up with Facebook.....	40
7.3.3	Link Facebook account	42
7.3.4	Unlink Facebook account	42
8	Evaluation	43
	References	44
	Figures	47
	Tables	48
	Appendixes	49

1 Introduction

This is a bachelor's degree thesis report. The thesis aims to solve particular issue of a case company, hence can be categorized as a functional thesis. The project work or practical part of this thesis is to implement the external authentication service provided by most suitable service provider in the case company's website. To test the implementation actually works, a test environment was created using relevant technologies. Various electronic reference sources such as research papers, blogs, articles, official websites, guides and surveys were used to collect required fact and figures.

This report is divided into two main sections: theoretical part and practical part. Theoretical part consists sections 3 and 4 of this report which include theoretical background and main concepts. Practical part consists sections 5, 6 and 7 which describe how and what was required to implement the project work. Research method used for this thesis work is briefly explained in section 2. Section 3, External authentication service (EAS) explains what EAS is and what are the benefits and drawbacks of it. Section 4 highlights the technologies and main concepts used for the thesis work. Section 5 list out number of EAS providers. In section 6, how and why one EAS provider was selected is described. Section 7 explains the implementation of EAS. The last section evaluates how the overall thesis work went. In the first section, "Introduction", the background of case company, problem statement, objectives and scope and limitations of the thesis work are discussed.

1.1 Background of case company

Viope Solutions Oy is a Finnish company founded in 2001. The company is located in the capital city of Finland. Viope has been providing e-learning platform since its establishment. It mainly focuses on developing better learning solutions. As mentioned in Viope's website (About Viope 2015), the company is pioneers on the field of automated assessment technology in computer programming and mathematics subjects. Services and products of the company has been used in over 70 countries. The mission of the company is to improve education by integrating digital learning with the world's best education system.

The platform currently provide two services targeted for specific group of users: Viope Pro and Viope Learning Environment. Viope Pro is a tool for professional individuals who want to study specific course. This tool offers badges and certification and can be accessed throughout the life time of the user. The second tool is more focused on educational institutes rather than an individual user. This tool provide more efficient ways of teaching and learning environment with its unique automated assessment system.

1.2 Problem statement

In today's world of internet and web, everyday millions of websites are used for various purpose such as communication, networking, bank transactions, watching videos and movies, listening songs, reading e-books and finding information. In a Microsoft research carried out by Florencio and Herley (2007, s.3.1.5), it was found that an average online user has 25 different accounts and they use 6.5 different passwords. This is quite old data, so today the number of online accounts and passwords used to access them must have increased. With the increased online accounts users may feel burden to remember all the usernames and passwords they have used.

As a solution to the problem caused by remembering many username and password sets, nowadays users are offered an option to use existing identity (a username/email address and password combination) by vast number of websites. At Viope, currently new user is registered by filling out the form where email address, username and password are entered. It makes a user to remember new username and password combination for further use of Viope's services. This thesis work has aimed to add a new feature in Viope's website which offer an option to use existing identity for registration and logging process, making it easier for users to use Viope's services.

1.3 Objectives

The main objective of this thesis work is to find the most suitable external authentication service provider and implement its authentication service in Viope's website. Below is the list of other objectives which will help achieving the main objective.

- Analyze the benefits and pitfalls of external authentication service. It is important to know whether something is truly useful and valuable before adopting it.
- After the analysis, list out various external authentication service providers and the basic user information they provide.
- Compare all the listed service providers and then select the most suitable one to integrate in Viope's website.
- Create a development environment for testing purpose.
- Find out the best possible way to integrate authentication service in Viope's website so as to get maximum benefit from it.
- Ultimately, integrate an external authentication service provided by the chosen service provider.

1.4 Scope and limitations

The implementation part of this thesis project was done in a separate server where minimally set up website was used as a development environment. Because of this, the code written to integrate external authentication service might not be directly used in the case company's live site. For example, handling of user registration might include sending a verification link to the email address in the live site which is omitted in the test site.

This thesis work has created just a basic working implementation which mainly focused on backend scripting leaving the front end side and actual implementation in live site to the case company.

2 Research method

Even though this is not a research based thesis, at certain extent research was also done to complete it. After all these years of studies what I have learnt about research is that it is something which is done to find out facts and figures, solution to some problem and exploration of ideas behind some topics in a systematic way. In general, it is a process to collect the information which would either support the existing idea or generate completely new idea.

There are myriad of sources from where required information can be accumulated. Some examples of those sources include books, interviews, observations, surveys, photographs, newspapers, research papers, videos, movies and experiments. Depending upon the nature of project one or more of these sources are used in any kind of research. With the rapid growth of technologies, World Wide Web or simply web has become big source of information which is stored in electronic form. Nowadays, electronic version of books, interviews, newspaper and research paper can be found in web/internet.

According to Eriksson and Kovalainen (2008), electronic research, e-research and online research means research activities that rely on communication that is mediated through computers and other new technologies. It has three categories: research literature on electronic media, existing electronic material and online research methods. Research carried out during this thesis work is heavily based on the existing electronic sources such as online survey, online research papers, articles published on various websites, blog posts, companies' information from their respective official websites and electronically published guides and instructions.

3 External authentication service (EAS)

External authentication service, EAS is a general term which basically means the use of external service provider for handling user authentication. It is also referred to as “social sign in”, “social login” and “single sign-on”. Social login or social sign on is an authentication service provided by social media sites or other web services. With this service, users can authenticate to third party websites using their login credentials from the service provider’s account. Social login service is implemented using protocols like OpenID, OAuth and SAML (Security Assertion Markup Language). Depending upon the protocol used, social login may also offer authorization services.

Single sign on (SSO) is access control method that enables users to log in once and gain access to multiple software systems without being prompted to log in again (Service now 2015). For example, Google offers different services which includes Youtube, Gmail, Google play, Google plus, Google drive, Google maps and Google search. Once the users log in in one of these services they will not be asked to log in again if they want to use another service, meaning they are automatically logged in to all other services as well. Protocols like LDAP (Lightweight Directory Access Protocol), SAML, Kerberos and Radius are used to implement SSO. In the context of this thesis work, EAS means social login not SSO. The external authentication process in very basic level is depicted in the following figure.

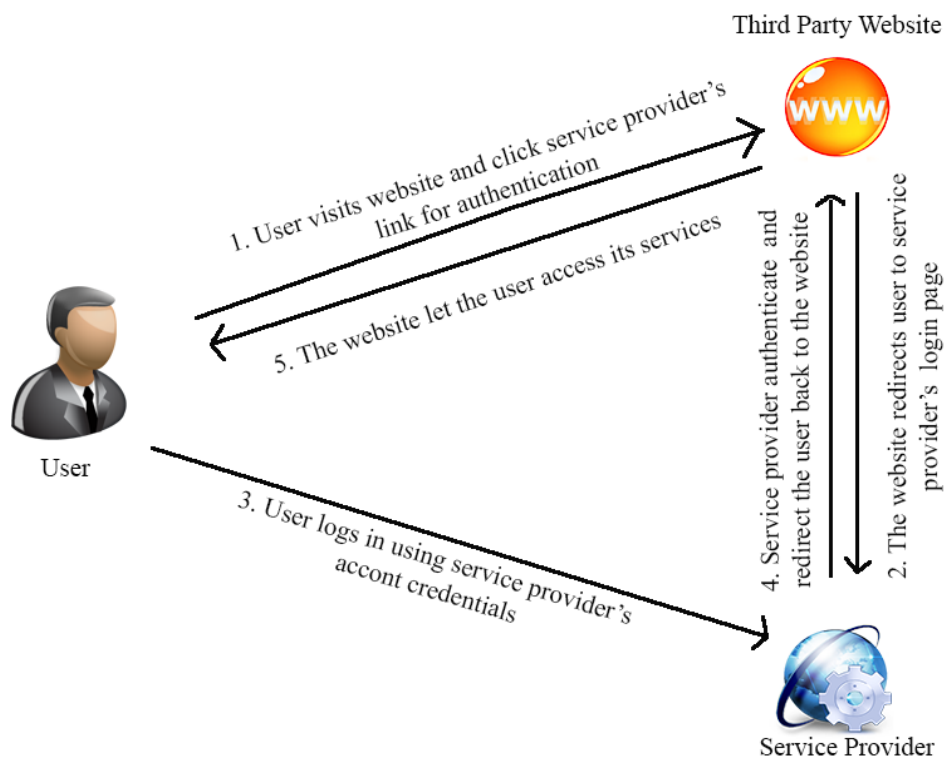


Figure 1: Basic external authentication process

As shown in above figure, a user visits some website which offers authentication with external service provider. It is possible that the website provides more than one service providers. User chooses the suitable service provider where he/she already has an account. The website redirects browser to login page of the chosen service provider. The user logs in using his/her account credential for that service provider. Depending upon the protocol used, user may be asked to authorize the website to access his/her information from service provider on his/her behalf. The service provider redirects the user back to the website. It also sends response to the website confirming the user's identity and a way to obtain user's information. Using the response from service provider, website authenticates the user and redirects the user to appropriate page.

3.1 Benefits of EAS

According to the survey conducted by Blue research for Jainrain (Goings & Abel 2013, 9), 87% of the respondents have come across a website that offers social login and 52% of them have used it. Gartner (Gartner 2013), has predicted that by the end of 2015 half of new retail customer identities will be based on the customer's social network profile. While in 2013, only 5% of the new customer's identities were based on their social network profile. These figures indicate social login is getting popular and it can be beneficial. In this section advantages of EAS are described in further three sub-sections:

- Benefits of EAS to users
- Benefits of EAS to companies
- Security benefits of EAS

3.1.1 Benefits of EAS to users

Now a days, user experience is considered very important part of any sort of services, hence web service also. Better the users feel more they use the service. This sub-section describes what kind of benefits social login offer to the users which enhance their experience.

As mentioned earlier in this report, there are many online accounts used by a single user and it can be very tough to remember all account credentials, especially passwords. To overcome the hassle caused by forgetting username or password, user tend to use the same password for different accounts or they use more easy and obvious password (Pickard 2008). This is not a recommended practice of using password as these type of accounts can easily be hacked. Social login allows users to reuse their single username and password combination to login across various web services. They do not need to remember all those different login credentials. Hence, reducing the problems caused by forgetting and mismatching passwords.

While creating new account a user has to go through lengthy process of filling out the registration form. Besides numerous input fields, complex password requirement and in some websites captchas (input field designed to distinguish human and system generated bots) could make it a frustrating experience for users. To solve this issue EAS integrated websites provide a single click sign up option which automatically fetch required user information and carry out the registration process. Users are provided with the option to register in the new website using their preferred identity provider's account. For example, codeschool.com offers three different identity providers, Github, Facebook and Googleplus for signing up a new user.

Users logged in or signed up via social login get personalized experience. User information (shared by user themselves) such as location, likes, address and relationship status makes it possible for the websites to provide relevant and useful services or products to the user. For example, a movie rental service can offer a bunch of popular action movies to the user who likes or follows action movie stars in his/her social network.

3.1.2 Benefits of EAS to companies

Not only users but also companies get number of benefits from EAS by integrating it in their official website. In this sub-section some of the advantages are discussed.

Conversion rate in general means the percentage of users who take a desired action (Marketing Terms 2014). Desired action may have different meaning in different situations. It may mean sales of products, membership registrations or like/sharing in social media networking sites. Social login helps businesses to increase conversion rate. For example, comfortable and effortless registration process provided by social login boosts the number of new registered members.

In most of the e-commerce sites, buyers are required to fill number of mandatory fields and their credit card information to check out items from the cart. Some buyers might find it frustrating or inappropriate and may abandon the cart instead of filling out the form. E-commerce sites can avoid this kind of issues to a certain extent by providing options to login via well-known and reliable payment providers like PayPal which simplify the payment process by pre-populating mandatory fields.

Companies require reliable and useful user data to make an effective marketing plan. Relevant data helps companies to ensure that they place right products or contents in the right place so the right user gets it easily. But getting the reliable user data could be very tough. Social login can help companies gather relevant data about their potential customers. By providing social

login option to customers, companies can request access to the useful information like relationship status, location, likes and hobbies from customer's social networking sites.

Taking advantage of rich user data with the help of social login, more personalized contents and product information can be offered to the user. When users find more personalized contents they automatically spend more time on exploring those contents. Once user get interested in spending time, chances of them viewing other related contents and pages from the website also increase. Patrick Salyer from Gigya (2012) writes in the blog that users logged in with a social network spend 50% more time on site and view twice as much pages as viewed by traditionally logged in users.

Companies want their customers to return frequently to their site and use their services over and over again. But it is not easy to make customers return again and again. Besides other reasons, forgotten login information can halt customers at certain extent from coming back to the website. According to the survey conducted by Blue research for Jainrain (Goings & Abel 2013, 9), 92% of the survey respondents have left the site instead of resetting or recovering login information. This issue can be resolved with the integration of social login. It automatically logs the user in to the site if they are already logged in in the service provider's site.

Socially logged in users do not have to remember new set of username and password. Because of this companies may avoid the hassles of dealing password related problems which reduces company's IT support costs.

3.1.3 Security benefits of EAS

It is clear that both users and companies get numerous advantages from EAS. The following paragraphs explain what security benefits EAS offer.

If not all, some of the top EAS providers such as Google, Facebook, Yahoo, Twitter and LinkedIn have implemented high level of security with full-time security teams. For instance, Yahoo's security technology analyze every single sign-in attempt in real time which checks the previous behavior of the user, reputation of IP address and the geographic location of sign-in attempt. Yahoo also let users to review their recent sign-in activity. In a similar fashion Google also analyze the user's login behavior. It alerts user if any unusual activity is detected to the user's account. Among other security tools, Facebook's Session Classifier application uses location, device and other account details to detect suspicious activity. Facebook Notification alert users if any suspicious activity detected. This kind of advanced security measurements and tools with dedicated security team can hardly be employed in small com-

panies. But integration of EAS make it possible for small companies to get the benefits of those highly sophisticated security tools.

Facebook, Google, Yahoo and Amazon are few representative EAS providers which have implemented multi factor authentication (also known as 2 step verification). Two step verification process asks users to give an additional password (mostly one time use 6 digits number sent as a text message to the user authenticated phone number) before giving access to their account. This prevents an unauthorized access even if the invader knows the username and password. It can be tough as well as expensive at the same time to implement multi factor authentication. With the integration of EAS, companies can take advantage of this security measure. They simply let their users to use one of those EAS provider which has implemented this security feature.

3.2 Pitfalls of EAS

Certainly EAS has not only advantages but also some downsides. Some users may find it irritating to link their social profile to the third party website while some may not mind doing it. In this sub section, some issues that may stop users or companies from using EAS are described.

3.2.1 Privacy concern of users

Users nowadays are getting more and more privacy conscious. They don't want to share their private information to anything that they do not trust. When it comes to using social login for registering or logging in to third party website, they want to know what kind of data is being shared by EAS provider to that website. According to the survey conducted by Survey Monkey on behalf of Gigya (2012), 49% of consumers skip social login because they do not want to transfer personal information to a website. Same survey shows that 41% of consumers skip social login because they are concerned about their web activity being shared to social networks without their permission.

3.2.2 Misuse of user data

One of the main reason companies offer social login is to get user data from social networking sites. But companies may not disclose explicitly how they are going to use that data. On the same survey from Survey Monkey which is referenced in the previous sub section, 63% of the respondents think that businesses sell their social data to other companies. Larry Drebes (2012) wrote on an article published in Forbes that Groupon (a website that features dis-

counted gift certificates usable at local and national companies) surprised its users by posting their purchases to Facebook without users being aware of it.

Though few in number, websites or companies which misuse user data exist. So users should be cautious enough before letting any websites use their information.

3.2.3 Dependence on EAS provider's availability

After integrating EAS, authentication system of the website depends on the up time and availability of that EAS provider. For example, let's assume that some website or web app integrate Facebook login and for some reason government banned Facebook from that country. That website/web app will definitely be affected as it would lose significant number of users and their data if the website can register only via Facebook login. This kind of incident might rarely happen though.

3.2.4 Security flaws

Nothing in the web is absolutely secure. They are secure until some bugs are found by researchers or hacked by some intruders. A research carried out by Or Peles and Roei Hay (2014) from IBM found serious bug in implementation of social login in some of the EAS providers including LinkedIn, Amazon and MYDIGIPASS.com. But before disclosing their finding they first contacted the vulnerable EAS providers to fix it. Wang Jing (2014), a PhD student in mathematics from Nanyang Technology University, found bug in OAuth and OpenID standards. Rui Wang, Shuo Chen and XiaoFeng Wang's (2012) study on the security of single sign on using OpenId protocol also found 8 security threats which allows access to an unauthenticated users. In this study also some of the top EAS providers including Google, PayPal and Facebook found vulnerable. But EAS providers were informed before disclosing the loopholes.

4 Concepts and Technologies used

Various technologies were used to implement EAS. To build the development environment for testing purpose, Perl, Nginx, Debian Wheezy, Perl Dancer, PostgreSQL and HTML were used. Authentication and authorization are important concepts used in computing systems which require identification of a user and inspection of his/her access permission. In addition to these technologies and concepts OAuth protocol is briefly explained in this section.

4.1 Authentication

Authentication is a process of determining whether someone is actually the one who he/she claims to be. In computing system, authentication is normally done with a username and a password combination. Username can be considered as the identity of the user and password is a proof of his/her identity. If user gives the correct set of username and password he/she is authenticated and let in to the system. Besides a set of username and password, biometric measures such as finger print, retina, face recognition and palm veins are also used for authentication purpose. A real world example of authentication can be proving that a person is actually the one to whom the parcel has been sent by showing a valid citizenship card or driving license while picking up the parcel from post office.

4.2 Authorization

Authorization is a process of determining whether a person or a system has permission to do what it is trying to do. Authentication must be done before authorization because without first identifying who is trying to access resources or functionality of a system it is not possible to determine if the action should be permitted or not. In multi user computing systems, generally a system administrator defines which users are allowed to access a system, what kind of actions they are allowed to take, which resources they can access and other privileges. Different authorization systems can be used depending upon the system requirements. Some examples of authorization systems include, Active Directory Security Groups, Apache Require Directive and External Access Control List. A real world example of authorization can be issuing a visa to travel abroad. It means that a person is authorized to enter a country for which visa has been issued.

4.3 OAuth

OAuth is a protocol which is used for authorization purpose. According to OAuth Core 1.0 documentation (2007), OAuth protocol enables websites or applications to access a protected resources in behalf of a user from a service providers via an API. With the use of this protocol user do not disclose his/her service provider credentials to the websites. For example, if user wants third party website to access basis profile information from his/her Facebook account, then OAuth makes it possible for that website to gain access to those user information without getting any knowledge of user's Facebook login credential. OAuth does not care about how the user is authenticated in Facebook before sharing his/her information. Amazon, Facebook, Dropbox, Dailymotion, Google, Microsoft, Yahoo, Evernote, Github and LinkedIn are some of the OAuth service providers.

First version of OAuth, OAuth Core 1.0, was published in December 4 2007 which was obsoleted by OAuth Core 1.0 Revision in July 24 2009 to address session fixation attack. OAuth 2.0 was published on October 2012 as the next version of OAuth which obsoleted the previous version.

The high level overview of how OAuth 2.0 works is listed below (OAuth 2.0 2012, 6).

- First a website or application which requires user information (referred to as client in OAuth specification) requests authorization directly from user or indirectly through authorization server (a server issuing access token which is a credential required to access private resources)
- Client then receive authorization grant(a credential required to get access token which represents a resource owner/user's authorization) from user
- Using that authorization grant client asks for access token from authorization server
- Client get access token if authorization server validates the authorization grant
- Client asks for protected resources from resource server (a server hosting the protected resources) using that access token.
- If valid access token is provided resource server grant access to the protected resource.

4.4 Perl

Perl is a group of high level general purpose programming languages. It consists of two programming languages namely Perl 5 and Perl 6. It was developed by Larry Wall as a general purpose Unix scripting language in 1987. Perl has undergone many changes and revisions. In 2000 Perl 6 was created as a redesign of Perl version 5. Later Perl 6 became a separate language in itself which is still in development phase. Perl 5 has been widely used in rapid prototyping and large scale development projects. It is an open source and free software. Below is the screenshot of sample Perl script.

```
1 use strict;
2 use warnings;
3 use diagnostics;
4 use Crypt::SaltedHash;
5 use Data::Dumper;
6
7 my $empty_pss = "";
8 my $salted_empty = Crypt::SaltedHash->new->add($empty_pss)->generate;
9
10 if(Crypt::SaltedHash->new()->validate($salted_empty, $empty_pss)){
11     print "This is empty password."
12 }
```

Figure 2: Perl sample code

4.5 Perl Dancer

Dancer is one of the web frameworks written in Perl programming language. It is an open source and free software written by Alexis Sukrieh. It was inspired by Sinatra, a web framework written in Ruby programming language. There are two versions of Dancer: Dancer (first version) and Dancer2 (latest and current version). Some of the key features of Dancer are mentioned below (Perl Dancer, 2015):

- Extremely simple: Intuitive, minimalist and very expressive syntax
- Flexible: PSGI (Perl Web Server Gateway Interface) support, strong scalability
- Few dependencies: It depends on as few CPAN (Comprehensive Perl Archive Network) modules as possible which makes it easy to install.

Following screenshot shows the sample of code written with Dancer framework.

```
#!/usr/bin/perl

# make this script a webapp
use Dancer;

# declare routes/actions
get '/' => sub {
    "Hello World";
};

get '/hello/:name' => sub {
    "Hello ".param('name');
};

# run the webserver
Dancer->dance;
```

Figure 3: Sample script written using Dancer framework

4.6 Nginx

Nginx is an HTTP (Hyper Text Transfer Protocol), mail proxy and reverse proxy server which was written by Igor Sysoev. It has been running huge websites including Yandex, Mail.Ru, Netflix and Wordpress.com. Out of several Nginx features only few are listed below (Nginx 2015):

- Serving static and index files
- Accelerated reverse proxying with caching
- SSL(Secure Sockets Layer) and TLS (Transport Layer Security) support
- Flexible configuration
- Name based and IP based virtual servers
- Validation of HTTP referrer
- The PUT, DELETE, MKCOL, COPY and MOVE methods

- FLV and MP4 streaming
- Embedded Perl

4.7 PostgreSQL

PostgreSQL also known as Postgres, is an object-relational database management system. It is an open source software and is free to use. It runs in different operating systems and platforms such as Linux, FreeBSD, Solaris, Mac OS X and Microsoft Windows. Like other software PostgreSQL is also revised frequently, resulting into creation of new and improved versions. The latest version of the software as of writing this report is PostgreSQL 9.4.1. Some of the key features of PostgreSQL as listed in its website (PostgreSQL 2015) are as follows:

- Complex queries
- Foreign keys
- Triggers
- Updatable views
- Transactional integrity
- Multi version concurrency and control

4.8 Git

Git is a free and open source software which is popularly known as a distributed version control software. It was developed by Linus Torvald, the creator of Linux operating system. It is mainly used in software development projects for storing different versions of the source codes. But it can also be used with any other files or documents where version controlling is required. Some of the features of Git as described in the about section of its official website are (About Git 2015):

- **Branching and Merging:** The feature that makes Git stand apart from almost every other Version Control Software.
- **Small and Fast:** Nearly every operations are performed locally, giving it a huge speed advantage on centralized systems which constantly have to communicate with a remote server.
- **Distributed:** It is distributed, which means that it allows many developers to work on a given project without requiring them to share a common network.
- **Data Assurance:** Git ensures the cryptographic integrity of every single piece of a project.
- **Staging Area:** It is an intermediate area where commits can be formatted and reviewed.

4.9 Debian Wheezy

Debian is a Linux operating system which is free and an open source software. 12 stable versions of the software has been released since it was first announced by Ian Murdock in 1993. The current and the latest stable version is Debian 7 which is given a code name of Wheezy. Debian comes with more than 37400 packages (precompiled software which can be easily installed), which are also free to use (The Debian Linux FAQ 2014).

4.10 HTML

Hyper Text Markup Language, HTML, is a building block of a webpage. It is the standard markup language which is written in the form of HTML elements or tags enclosed in angle brackets. Web browsers can read HTML files. It converts all the HTML tags and scripts and present it in human friendly format. There are different versions of HTML, the latest and current one being HTML 5. Below is the sample code written in HTML 5.

```
<!DOCTYPE html>
<html>
  <head>
    <title>Sample HTML file</title>
  </head>
  <body>
    <h1>The body of the webpage goes here.</h1>
  </body>
</html>
```

Figure 4: HTML code sample

5 List of EAS providers

There are different types of EAS providers including social media networking sites, payment solution companies, cloud computing solution companies, email providers and many more. Most of the social login providers use OAuth while some use OpenID and some use both for authentication services. There is a dominance of social media networking sites and micro-blogging sites among EAS providers. This section is further divided into two sub sections: social media providers and non-social media providers, which list out some of the popular EAS providers.

5.1 Social media providers

There are many social networking and micro-blogging sites that provide social login service. Some sites are globally popular while some are popular only in few countries. Below is a list

of some of those social media providers which are popular among particular communities or in particular countries.

- **RenRen** and **Kaixin001**, two Chinese social networking sites which are popular in China and other countries where Chinese is widely spoken.
- **Vkontakte (VK)** is Russia based popular social networking site mostly used in Russia and Russian speaking countries like Ukraine, Latvia, Kazakhstan, Belarus and Uzbekistan.
- **Live Journal** is a popular blog-platform in Russia. Large number of people from USA also use Live Journal. Russia and USA are two countries which cover huge portion of total active users in Live Journal. It supports OpenID standard for authentication.
- **Mixi** is a Japanese social networking site which requires valid Japanese cell phone number for registration.
- **MeinVZ** also known as **StudiVZ** and **SchülerVZ**, is a Germany based social networking site which is popular in German speaking countries like Germany, Switzerland and Austria.
- **Sina Weibo** and **Tencent Weibo** are two popular Chinese micro-blogging websites which are also considered as “Twitter of China”. Sina Weibo has huge user base, 503 million registered users, in China and in Chinese communities in more than 190 countries (Sina Weibo 2014). And Tencent Weibo has 469 million registered users (Tencent Weibo 2014).

Popular globally used social networking sites providing EAS are listed below.

- **Facebook:** Facebook is one of the globally renowned social networking site which was established in 2004. According to the statistics from Facebook (Facebook Statistics 2014), there are 864 million daily active users on average for September 2014 and approximately 82.2% of daily active users are from outside USA and Canada.
- **Twitter:** Twitter is a micro-blogging site which was founded in March 2006. It is based in San Francisco, United States. Twitter’s official website (About Twitter 2015) shows that they have 288 million monthly active users and 77% of accounts are from outside the USA.
- **LinkedIn:** LinkedIn is a well-known social networking site for professionals. It is different than other social media networks as it is specifically meant for professional networking. LinkedIn announced (LinkedIn Press 2014) that it has reached 300 million members (67% from outside USA) in more than 200 countries and territories.
- **Google+:** Google+ / Google plus is a social networking site owned by Google. Google plus is integrated with other Google services like Gmail and Youtube.
- **Instagram:** Instagram is a social networking site that allows its users to take pictures and videos and share them with their friends and family.

- **Xing:** Xing is a Germany based social network for business professionals. As of September 2014, it has more than 14 million members out of which 7.7 million are from German-speaking countries (Xing 2014).

These social networking sites have huge amount of information about their users. Using OAuth specifications, these sites authorize third party websites to access some of the user information. Basic information such as name, age, user Id and address can be requested without special permission while some information such as email address, likes and hobbies require special permission. Depending upon the EAS provider, information they share may vary. The following table shows list of service providers and basic information they share. NOTE: There are much more information that can be requested from these service providers.

	Ren Ren	Kaixin001	VK	LiveJournal	Mixi	Mein VZ	Facebook	Twitter	LinkedIn	Google Plus	Instagram	Sina Weibo	Tencent Weibo	Xing
User Id	×	×	×		×	×	×	×	×	×	×	×	×	×
Address	×	×	×		×	×	×	×	×	×		×	×	×
Birthday	×	×	×		×	×	×		×	×			×	×
First Name			×		×	×	×	×	×	×			×	×
Last Name			×		×	×	×	×	×	×			×	×
Full Name	×	×	×		×	×	×	×	×	×	×	×	×	×
Gender	×	×	×		×	×	×			×		×	×	×
Email						×	×		×	×				×
Mobile number			×						×					×
Profile Photo	×	×	×		×	×	×	×	×	×	×	×	×	×

Table 1: List of social media EAS providers and basic user information they provide

5.2 Non-social media providers

Different types of web services other than social media sites also provide EAS. Below is the list with brief description of some of the popular non-social media EAS providers.

- **PayPal:** PayPal, owned by e-bay (e-commerce site), is a web service providing payment and money transfer solution through internet. In other word, it is an online money transfer service which serves as electronic alternative to paying with traditional paper method such as checks and money orders.

- **Yahoo!:** Yahoo! is an internet corporation specially recognized for its search engine, Yahoo Search and mailing service, Yahoo mail. Besides these Yahoo services include Yahoo Directory, Yahoo News, Yahoo Finance, Yahoo Answers, advertising, online mapping and video sharing platform. Yahoo is one of the trusted and popular OpenId providers.
- **Microsoft account:** Microsoft account, previously known as Windows Live Id is a web service provided by Microsoft which allows users to use a single account or identity to log into other windows products like web-based email service (Outlook.com), Windows 8 computers and tablets.
- **Amazon:** Amazon is an e-commerce company, one of the largest online retailers in the USA. Besides online retailer, it is also known as one of the major cloud computing service provider.
- **AOL:** AOL Inc, also known as America Online, is an American mass media corporation based in New York City. AOL has numerous products and services in the field of content, advertising, local, membership and ventures.
- **Flickr:** Flickr is an image and video hosting website created by Ludiocorp, now owned by Yahoo. It is popular among users for sharing and embedding personal photographs. It allows developers to create non-commercial application by using its Application Programming Interface (API).
- **MYDIGIPASS.com:** MYDIGIPASS.com is a product of Vasco, the authentication company. It provides two-factor authentication platform to protect the web applications. Websites can integrate MYDIGIPASS.com as an EAS by using Secure Connect API provided by MYDIGIPASS, which uses OAuth 2.0 (draft) protocol.
- **QQ:** QQ, also known as Tencent QQ is an instant messaging service developed by Chinese company Tencent Holdings Limited. It provides its services not only in Chinese but also in English, French, Japanese, Spanish, German and Korean language. With the international edition of QQ, it has increased its user base in other countries where Chinese is not widely spoken.
- **Wordpress:** Wordpress is an open source blogging tool and a Content Management System (CMS) based on PHP and MYSQL. 23.3% of all the websites in the internet are built using Wordpress (W3Techs 2014).
- **Github:** Github is a web-based Git repository hosting service which offers all of the distributed revision control and source code management functionalities of Git in addition to its own features.
- **Disqus:** Disqus is a comment hosting service for websites and online communities. It can be integrated with any type of website and comments from those website can easily be imported to it.
- **Foursquare:** Foursquare is a local search and discovery service which provides a personalized local search experience for the user based on user's location information.

- **Blogger:** Blogger is one of the services offered by Google which allows users to publish blog posts. The blog posts are hosted at a subdomain of blogpost.com. It is available in 62 different languages including Hindi, Spanish, Portuguese and Finnish. It supports OpenID for authentication.

Like social media providers, the above listed EAS providers also let third party websites to access bunch of user information. Following table lists only the basic profile data of user.

	Pay Pal	Yahoo	Microsoft	Amazon	AOL	Flickr	MYDIGI PASS	QQ	Github	Word press	Disqus	Four-square	Blogger
User Id	x	x	x	x	x	x	x	x	x	x	x	x	
Address	x	x	x	x	x	x	x	x	x			x	
Birth-day	x	x	x		x	x	x						
First Name	x	x	x		x	x						x	
Last Name	x	x	x		x	x						x	
Full Name	x	x	x	x		x	x	x	x	x	x		
Gender	x	x	x		x	x		x				x	
Email	x	x	x	x	x	x	x		x	x	x	x	
Phone Number	x	x	x			x	x					x	
Profile Photo		x						x	x	x	x	x	

Table 2: List of non-social media EAS providers and basic user information they provide

6 Selection of EAS provider

There are plenty of EAS providers from different origin and background as listed in the previous section. Among those service providers one of the most suitable provider had to be selected to integrate in the case company's website. To do that different reports on global social login trends were analyzed. Based on the analysis and company's requirement which service provider was selected is explained in the subsequent paragraphs.

According to Gigya (Gigya Infographics 2014), a company which provides social login solution along with other services, Facebook is the most used EAS provider on the web. 58% of their

consumers used Facebook for social login. Google plus is at second position with 24% and both yahoo and twitter were used by 6%. The same infographics present various data based on devices (such as pc/web or mobile devices), type of websites (for example, ecommerce site, media/publisher, consumer brands, education/non-profit and travel/hospitality) and geographic location (like Africa, Europe and Asia/Pacific). In every category, Facebook has topped the list with highest percentage of usage and with few exceptions Google plus has maintained its second position. Twitter is competing with other EAS providers such as Yahoo, LinkedIn, Amazon and VK to achieve the third position. A complete infographics is attached in the appendixes section of this report.

Michael Olson from Janrain, a company like Gigya, writes in a blog (2014) that Facebook is preferred by 46% of their consumer for social login while Google Plus is preferred by 34% and 7% of the consumers preferred Twitter. Since 2009 Facebook and Google Plus are dominating the social login trend as shown in the following graph taken from Janrain.

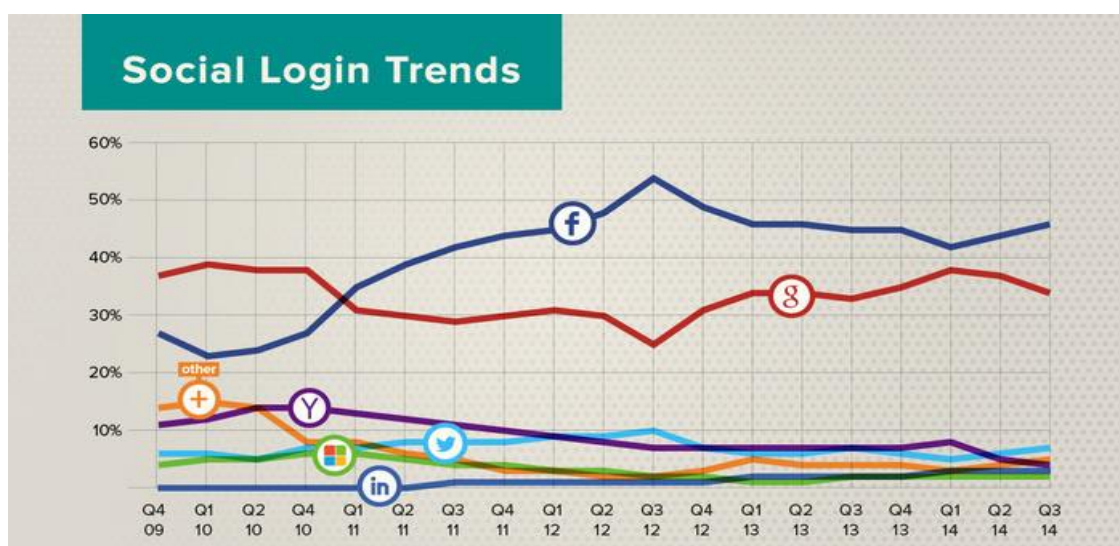


Figure 5: Social login trend analysis by Janrain

A quarterly report published by Loginradius, a company like Jainrain and Gigya (2014), also shows the dominance of Facebook and Google Plus for social login. The following pie chart shows popular EAS providers with the percentage of user preference in the third quarter of 2014.

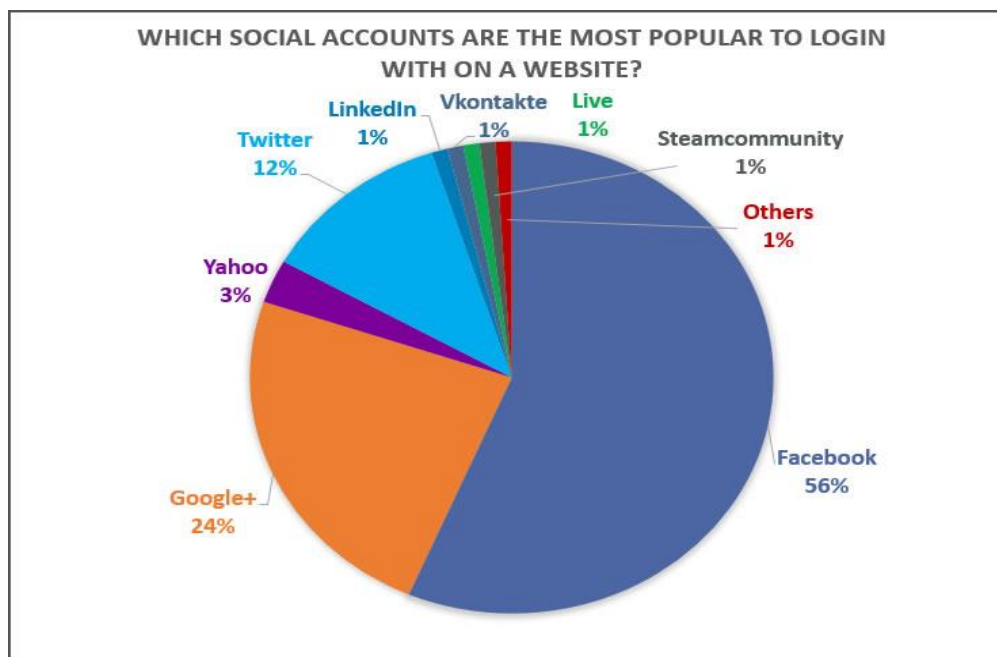


Figure 6: Social login trend report by LoginRadius

These data from above mentioned reports show that Facebook, Google Plus, Twitter, Yahoo and LinkedIn are the top 5 mostly used EAS providers. On the base of these facts and figures, all the other EAS providers were eliminated. Among these 5 EAS providers also, it was required to filter out the most suitable provider for the case company.

All the above mentioned reports were discussed with the Chief Technology Officer of the case company. He showed his interest in Facebook and he advised me to do research on popularity of Facebook in the context of Finland. Following paragraphs highlight some of the existing reports/studies on social networking site's popularity in Finland.

It is mentioned in an article from eMarketer (2014) that Facebook is the most dominant social media site in Finland which is visited by 82% of the internet users at least daily. According to the same article, Instagram is second most popular with 15% internet users which is followed by Google Plus (9%), 5% Twitter users, 3% Pinterest users and 2% LinkedIn users. It is mentioned in the article that source of these data is Buzzador. In the same article, it says that data from DNA Oy also found Facebook is the most popular social media site. In January 2014, Facebook was used by over 66% of the internet users polled in the country, while Google Plus was used by 20%, around 10% used LinkedIn and Twitter and 8 % used Instagram.

Recently, news article on Yle (Finns favor Facebook 2015) reported that 56 % of a Taloustutkimus survey respondents used Facebook, 18% used Google Plus and 10% used Twitter. While Instagram was used by 16% of the respondents. The same survey suggests that 90

percent of the respondents who were active on Facebook, were 15 to 24 years old users. Har- to Pönkä (2014), writes in his blog that number of Facebook users in Finland in the beginning of 2014 was 2193000. Toni Nummela's (2015) website shows the number of twitter users who tweet in Finnish language has reached 285659 since February 7, 2013. Quite old data from Yle (2013) shows that there were approximately 50000 Finnish google plus account out of which only about 20 000 were expected to be active.

All of these data clearly indicate that Facebook has huge user base and it is the most popular among the five selected service providers in Finland. Based on these data and case company CTO's interest, Facebook was selected for integrating EAS in Vioppe's website.

7 Implementation of EAS

This section covers all the details of Facebook's authentication service implementation. There are three sub sections, first sub-section highlights the best practices of EAS integration in existing login system. Second sub-section covers how a basic development environment was created. The third and the final sub section shows how the best practices were applied to integrate EAS in the test website.

7.1 Social login best practices

As Facebook was going to be integrated with Vioppe's website which already has custom authentication system, it was worth checking Facebook's recommendation on using Facebook login (an authentication service which uses OAuth 2.0) with existing login system. The following paragraphs describe some of the possible scenarios and Facebook's suggestion on handling those scenarios (Developers Facebook 2015).

First case: A user sign up or register to the website using existing registration process, but later he/she wants to use Facebook to log into the website instead of using custom login credentials. In this kind of scenario, there is a chance of creating duplicate account for the same user. One already created using the website's registration system and other created by Facebook login process. To handle this case, Facebook suggest that email address should be asked from the user while completing Facebook login. Then use that email address to check if any account with the same email address exists in the database. But there is no guarantee that the user would use same email address in both Facebook and the website. In that case, user should be offered an explicit account merging option.

Second case: A user logs into the website through its login system and while still logged in he/she wants to associate his/her Facebook account. By associating the website's account

with Facebook account the user can later use Facebook for logging in to the website. It is suggested to create new table in the database for storing user's Facebook information separately. An alternative for that would be to add relevant columns in the table where user's information are stored.

Third case: New user creates an account in the website using Facebook login, but later he/she wants to be able to login using custom login credential. In this scenario, first of all, email address retrieved from Facebook should be verified if email address is used as a unique credential which identifies each account. To do this, a verification email can be sent to that email address. Once email is verified, a user can be asked to supply new password. After this a user can be informed that he/she can use email and password combination for further login. If username is used as unique identifier then the user can be asked to supply username and password instead.

Fourth case: A user who had associated his/her Facebook account to the website's account wants to disassociate it. In this situation a user should be offered an option to remove his/her Facebook account. One way to do that is by simply deleting the Facebook information of that particular user from the database.

7.2 Creating development environment

It is very important to have a test or development website to test new features before implementing it in the live site. The CTO of the case company said that Viopé's test site is so complex which could take lot of time and resources to get familiar with. So it was not convenient to work on company's test site. Because of that I had to create a development environment myself to test the new feature, social login.

Company provided me an empty server, which was a Debian Wheezy machine, where I had to create a test website. The test website should be built in such a way that the code used for integrating Facebook login can be directly used in Viopé's live website. So I had to use exactly same tools and software used by the live website. In the following sub sections, how the test environment was created is described.

7.2.1 Installation of required technologies and tools

In order to interact with the server I had used SSH (Secure Shell) connection through the terminal. Once the connection to server was established, I started to install all the required technologies which were not already installed in the server.

Perl programming language is used in the backend section of Viopé's website. Perl version 5.14.2 was already installed in the server. The latest stable version of Perl is 5.20.1 but it was not upgraded as the older version was the one used in the live site. For database server, PostgreSQL was installed. Perl Dancer framework which is used to create Viopé's website was then installed. Nginx web server is used as reverse proxy server in live site. It was pre-installed in the server. The next tool required to install was Git. It was required and recommended by the company CTO to pull and push code from and to the server. All these tools were also installed in the local computer in order to create local development environment as it was not convenient to code directly in the remote server. Ubuntu operating system was used instead of Debian because of two reasons: first, it was already installed in the local machine and second, both are Linux operating systems.

7.2.2 Creating simple Dancer app

Once all of the required software were installed, a new Dancer web app was created by typing the following command in the server's terminal:

```
dancer -a DemoWebsite
```

This command creates a new Dancer app called DemoWebsite and it includes following files and directories:

```
+ DemoWebsite
+ DemoWebsite/config.yml
+ DemoWebsite/Makefile.PL
+ DemoWebsite/environments
+ DemoWebsite/environments/development.yml
+ DemoWebsite/environments/production.yml
+ DemoWebsite/lib
  DemoWebsite/lib/
+ DemoWebsite/lib/DemoWebsite.pm
+ DemoWebsite/MANIFEST.SKIP
+ DemoWebsite/t
+ DemoWebsite/t/001_base.t
+ DemoWebsite/t/002_index_route.t
+ DemoWebsite/public
+ DemoWebsite/public/dispatch.cgi
+ DemoWebsite/public/javascrpts
+ DemoWebsite/public/images
+ DemoWebsite/public/css
+ DemoWebsite/public/css/style.css
+ DemoWebsite/public/css/error.css
+ DemoWebsite/public/404.html
+ DemoWebsite/public/dispatch.fcgi
+ DemoWebsite/public/500.html
+ DemoWebsite/bin
+ DemoWebsite/bin/app.pl
+ DemoWebsite/views
+ DemoWebsite/views/index.tt
+ DemoWebsite/views/layouts
+ DemoWebsite/views/layouts/main.tt
```

Here, DemoWebsite is the root folder of the app inside which all the configuration, HTML, CSS, images and other relevant files are created and stored. The core application file is app.pl which is located inside the bin sub directory. All the HTML files are stored inside the views sub-directory. All the static files like images and CSS files are stored inside public sub-

directory. The main configuration file of the app is config.yml file. Environment (development and production) specific configuration files are located inside environments sub-directory. DemoWebsite.pm inside lib sub directory is a file that is mainly used for routing (handling browser requests). Once the Dancer app is created, it can be run using Dancer's built in web server by typing "perl bin/app.pl" in the terminal. Default Dancer app runs at port 3000. If the server's IP address is 1.2.3.4, then the Dancer app can be accessed in the browser by typing <http://1.2.3.4:3000> or if it is installed in the local machine <http://localhost:3000> should be used instead.

7.2.3 Setting up Nginx

After creating a Dancer web app, Nginx was configured as reverse proxy server for the Dancer's built in web server. It was done in order to access Dancer web app without using the port 3000. It means, <http://1.2.3.4> can be used instead of <http://1.2.3.4:3000>.

First a file, nginx-demo.conf (*nano nginx-demo.conf*) was created in the directory where Dancer app was located and filled that file with the following information.

```
server {  
    location / {  
        proxy_set_header Host $http_host;  
        proxy_set_header X-Forwarded-Host $http_host;  
        proxy_set_header X-Real-IP $remote_addr;  
        proxy_set_header X-Forwarded-For $proxy_add_x_forwarded_for;  
        proxy_pass http://localhost:3000;  
    }  
}
```

The default nginx configuration file was then deleted and a symbolic link from the nginx configuration directory to nginx-demo.conf file was created. Nginx server was restarted to apply the changes in configuration file.

7.2.4 Setting up Git

The next thing needed to be configured was Git. When Git is installed, few things should be done to customize the Git environment. These things are done only once on any given computer. First thing that needed to be done was to set the username and email address. It is important because these information are used as an identity which is used in every commits. The other thing that was supposed to be done was to set the default text editor. I did not do that because it was not so important. After setting username and email a new Git repository

was created inside the app's directory by using `git init` command. This command creates a new subdirectory called `.git` which contains all the necessary repository files. In order to start tracking files, all those files and subdirectories were first added to the staging area and then committed using Git command like this:

```
git add .  
git commit -m "created first git repository"
```

The next step was to create a bare repository of the app and set the remote origin url of the app to the bare repository and finally clone the bare repository in the local machine.

```
git clone --bare DemoWebsite DemoWebsite.git  
git remote set-url origin file:///home/rabi/DemoWebsite.git  
git clone rabi@78.47.150.59:/home/rabi/DemoWebsite.git
```

The last Git command was executed from the local computer which cloned the app's git repository in local machine. That was all needed for setting up Git. I could push all the changes I made in local repository to the remote server and from server I could pull the changed version of files and folders.

7.2.5 Setting up database

To set up database, first I logged in as a user named postgres in terminal. Postgres user and a database named postgres are automatically created by PostgreSQL on installation. Logging in as postgres means logging into to postgres database from where I could manipulate the database. I created new role or user named "demouser" with super user privilege and set its password to "demouserpass". Then a new database named "demodb" was created.

Viope had previously integrated its service with another web service called moodle.org. So the database structure was already modified to store the user information from external sources. This project also used the same database structure with small changes. What was needed for this project were two tables each of which stores user's internal information and external source related information respectively. First table called person, was used to store basic user information such as user id, username, password, first name, last name, address, postal code, city, country code, phone number, email address, gender and few other columns. Another table called external_source_info, was used to store name of external service provider, username in external account, user id in external account and some other columns. The database table structure is attached in the appendixes section of this report.

7.2.6 Creating test website

After creating a simple Dancer app and configuring all the required tools, a test website was created. This test website consisted a very basic authentication system which was required to test Facebook login integration.

First, all contents of the default template files were replaced with simple HTML contents. Template engine is responsible for displaying template files in the browser. Template files are basically HTML files with .tt extension which are stored in the views directory of a Dancer app. By default, Dancer use simple template engine, Dancer::Template::Simple which was upgraded to more advanced engine, Dancer::Template::TemplateToolkit. To upgrade the template engine, Debian package for Template Toolkit was first installed and following changes were done in the config.yml file.

```
# template engine
# simple: default and very basic template engine
# template_toolkit: TT

#template: "simple"

template: "template_toolkit"
engines:
  template_toolkit:
    encoding: 'utf8'
    start_tag: '<%'
    end_tag: '%>'
```

To handle what happens when a browser send request for webpage, routes are declared in appname.pm file (DemoWebsite.pm file). A route declaration indicates which HTTP method/s it is valid for, path it matches and the code which respond that request. An example of route declaration is shown below:

```
get '/login' => sub{
  if(!session->{logged_in_user}){
    template 'login.tt';
  }
  else{
    redirect "/dashboard";
  }
};
```

In the above example, the script matches HTTP GET request for login page and handles that request with the code block, which is a Perl subroutine/function. The code block send login.tt template file to the browser if the user is not logged in otherwise redirects the browser to dashboard route.

To handle session, `Dancer::Session::YAML` session engine was used. It comes installed with the Dancer framework. To use this session engine, it was enabled by adding the following lines in the `config.yml` file.

```
#session config
session: "YAML"
```

`Crypt::SaltedHash` module was used to encrypt the password before saving it to the database. Debian package for this module called `libcrypt-saltedhash-perl` was installed.

Dancer cannot directly interact with database server. It requires few Perl modules as shown in the figure below (Perl Database Access 2015):

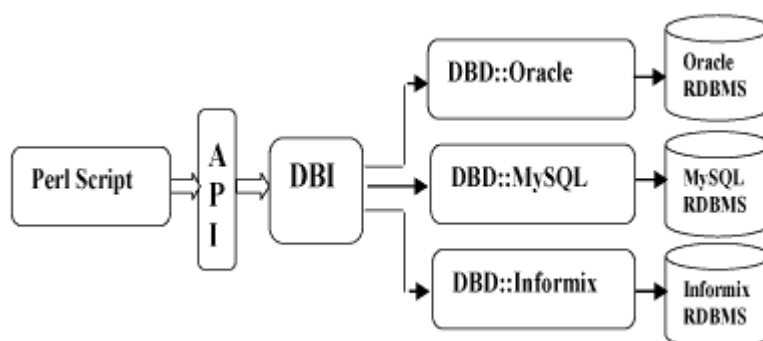


Figure 7: Architecture of a DBI application

Any Perl script need two modules: DBI and database driver module (in this project's case `DBD::Pg`) in order to interact with the database. DBI is a database access module for Perl programming language which provides database interface independent of the actual database being used. DBI is responsible of taking SQL queries and dispatching it to the appropriate driver for actual execution. It is also responsible for taking the results from driver and giving it back to the calling script.

DBI was already installed with Dancer framework. Debian package for `DBD::Pg` module called `libpq-dev` was installed. In addition to these two modules, another module called `Dancer::Plugin::Database` was also installed because it provides easy and convenient methods to manipulate database. `Dancer::Plugin::Database` was installed using Debian package called `libdancer-plugin-database-perl`. `config.yml` file was then updated to include database information as follows:


```
#Database connection details
plugins:
  Database:
    connections:
      main:
        driver: 'Pg'
        database: 'demodb'
        host: 'localhost'
        port: 5432,
        username: 'demouser'
        password: 'demouserpass'
```

Dancer::Plugin::FlashMessage module was installed to display temporary messages, also known as flash message. It can be used to inform the user that a certain operation has just happened. Debian package called libdancer-plugin-flashmessage-perl contains this module.

All of the above mentioned modules were imported in a script (DemoWebsite.pm file) where they were being used as follows:

```
package DemoWebsite;
use Dancer;
use Dancer::Plugin::Database;
use Crypt::SaltedHash;
use Dancer::Plugin::FlashMessage;
use Template;
```

After installing and configuring all those modules, different route handlers and necessary template files were created and a test website was built. Following screenshots show how the test website looked like. More screenshots of the test site are attached to the appendixes section.

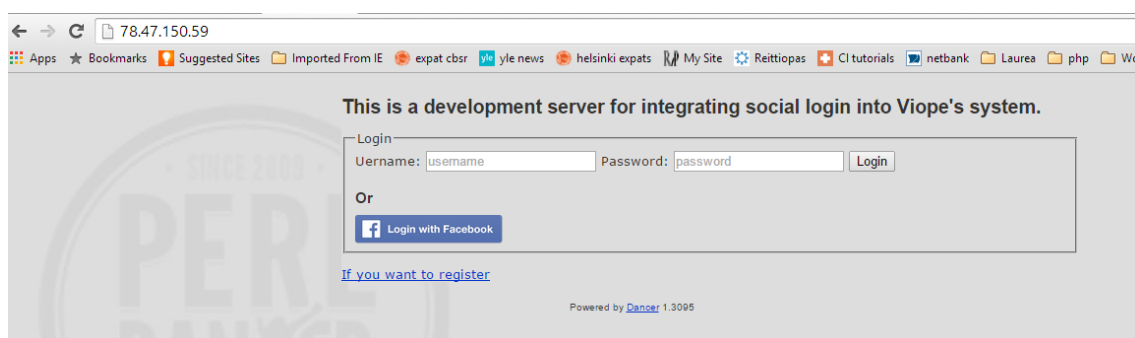


Figure 8: Homepage of test site

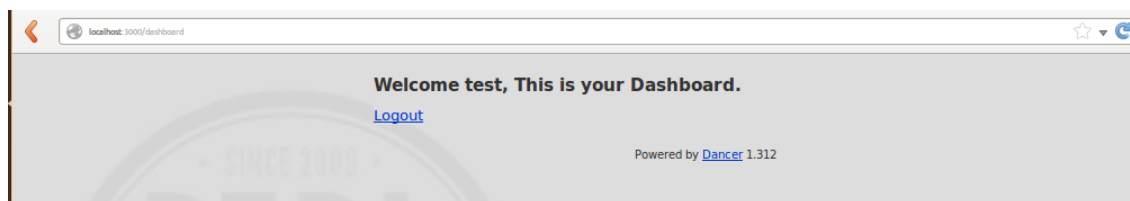


Figure 9: Dashboard page of test site

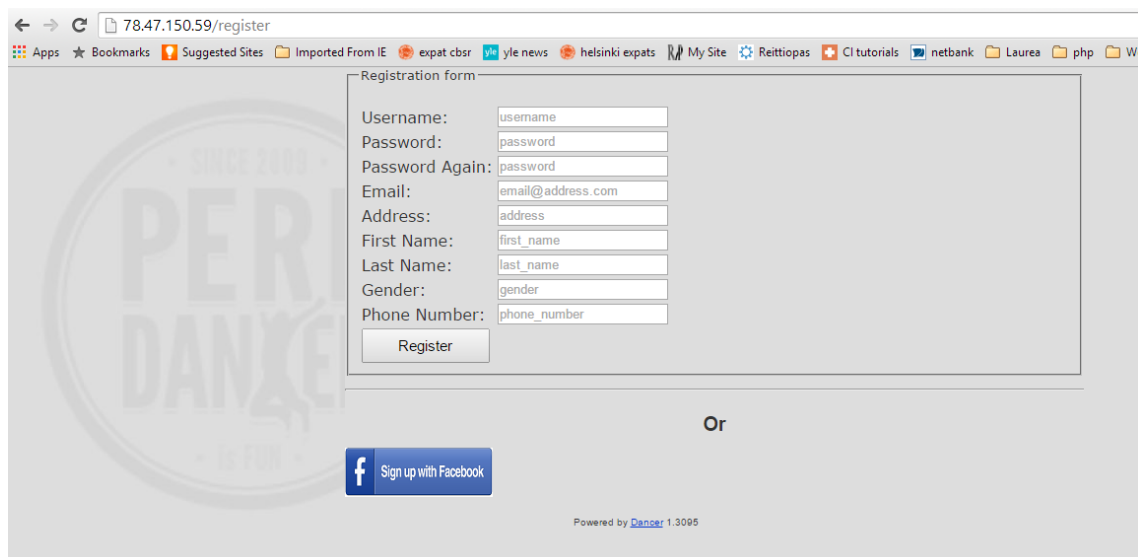


Figure 10: Registration page of test website

Once a simple test website was created in the local machine, the app was pushed to the server using Git's push command and pulled from the server using pull command. All the modules mentioned in this sub section were installed both in the server and the local computer.

7.3 Integrating Facebook Login

Facebook offers Javascript and PHP SDKs for implementing Facebook login in third party websites. In the case of this project, none of the SDKs were suitable because it requires server to server communication in Perl language. As a result, a manual Facebook login flow was built. To build login flow, Facebook app was required. So a test Facebook app called testFbApp was created following these steps:

- Log in to Facebook
- Under "Developer" section, click Manage Apps
- Under "MyApps" tab, select Add a New App
- Choose Website from prompted dialog box

- Type name of new app and click Create New Facebook App ID
- Choose one of the categories and then click Create APP ID
- Quick Start for Website section can be skipped

The user must be registered as a developer in Facebook in order to complete above mentioned steps. The most important pieces of information needed from testFbApp are app id and the app secret. These information were stored in config.yml file along with post back url which is used by Facebook to redirect the user after completing login flow.

```
#Facebook config
facebook:
  postback: "http://localhost/facebook/postback/"
  app_id: "1594762367411421"
  secret: "sdRs123087f9077e1b51e0e2bfaetee35"
```

All the Facebook related information were assigned to separate variables in DemoWebsite.pm file in order to use them easily.

```
#Facebook related info

my $fb_app_id = config->{facebook}->{app_id};
my $fb_postback = config->{facebook}->{postback};
my $fb_app_secret = config->{facebook}->{secret};
```

Facebook login dialog is needed to invoke as a first step of Facebook login flow. Facebook login dialog was used for three different purposes in this project: for logging in existing users, for registering new users and for linking the existing user's Facebook account. In the test website, three links were created (first one in login page, second one in registration page and third one in the dashboard page) all of which send HTTP GET request for '/facebook/:fb_case' route. The route handler then invoke the Facebook login dialogue, meaning it redirects users to Facebook login page. In the route declaration, :fb_case is sent as a parameter which indicates where the request came from, log in page or registration page or account linking page. For example, a link in login page looked like this:

```
<a href="/facebook/fb_login"> </a>
```

'/fb_login' in the href attribute of the anchor tag was used as a signal that indicates the anchor tag is located in log in page. Link in registration page has href attribute with the value equals to '/facebook/fb_registration' while the link in dashboard page has href attribute with the value of '/facebook/facebook_link'. Following is the route handler script:

```

get '/facebook/:fb_case' => sub {
  #redirect user to FB login dialog page
  my $state = params->{fb_case};
  redirect "https://www.facebook.com/dialog/oauth?
  client_id=$fb_app_id&redirect_uri=$fb_postback&scope=email&state=$state";
};

```

The redirect url is consist of four parameters:

- client_id: This is a required parameter and it is the ID of testFbApp.
- redirect_uri: This is also required parameter and it is the post back url where users are redirected after logging in to Facebook.
- scope: This is an optional parameter and it is a comma separated list of permissions to request from a user. Only email address access permission was requested from user because it is a required user information which is used to handle different authentication scenarios. Facebook sends public profile information of a user by default.
- state: This optional parameter is supposed to be an arbitrary string to guard against Cross-site Request Forgery attack. But in this project, it was used to distinguish whether the user is logging in or signing up or just linking Facebook account.

Once the user is logged in in Facebook, for the first time, he/she is prompted a dialog box which gives him/her an option whether to allow the website to access his/her email address (and/or other information if they were passed in scope parameter) or not. Regardless of whether user grant access or not, Facebook redirects the browser to the post back url with the access code and state parameters attached to it. Then the post back url route handler retrieve the access code and the state information. This state information is used to handle different case scenarios which is covered later in this report. Access code is exchanged for access token which is required to fetch user information from Facebook. To request access token, the script needs to make HTTP Get request to the OAuth endpoint. In order to make HTTP requests from script, LWP::Simple module was used which is one of the modules that comes bundled with LWP library. The LWP library was installed from Debian package called libwww-perl.

```

#get post back from facebook
get '/facebook/postback/' => sub {
  #assign authorization code to the variable auth_code
  my $auth_code = params->{code};
  my $fb_case = params->{state};

  #access token request url
  my $access_token_url = "https://graph.facebook.com/oauth/access_token?
  client_id=$fb_app_id&redirect_uri=$fb_postback&client_secret=$fb_app_secret
  &code=$auth_code";

  #get access token response
  my $access_token_response = LWP::Simple::get($access_token_url);

```

In the code above, `access_token_url` is the OAuth end point where the script makes HTTP GET request for access token. It has following required parameters:

- `client_id`: It is the ID of the testFbApp.
- `redirect_uri`: This parameter must be the same as the original post back url that was used when starting the OAuth login flow.
- `client_secret`: It is the testFbApp secret which can be accessed from Facebook App Dashboard. This information must be kept secret as it is the core of the security of Facebook app and all people who use that app. So this should never be used in client side code.
- `code`: The access code received from the Login dialog redirect

The response received from this end point, if successful, is a string of access token and the time in seconds when it expires. Explanatory error message is received if the request was failed. If the successful response is returned access token is separated from the returned string and then it is used to access user information as follows:

```
if($access_token_response){
    my %access_token_hash = split /&|=/, $access_token_response;
    my $access_token = $access_token_hash{'access_token'};
    my $app_token = app_token();

    #get user information from Facebook Graph API call
    my $fb_user_info_url = "https://graph.facebook.com/me?
    access_token=$access_token";

    my $fb_user_info_response = LWP::Simple::get($fb_user_info_url);
    my $fb_user_info = decode_json($fb_user_info_response);
}
```

A HTTP GET request to Facebook's Graph API is made using access token. Graph API then response with the user information in JSON format which is decoded as a hashref (reference to a basic Perl data type called hash) and assigned to variable called `fb_user_info`. JSON module installed from Debian package `libjson-perl` was used to handle the objects in JSON format.

The access token can be inspected to verify that it is valid for the person and for the test-FbApp by using Facebook's Graph API endpoint.

```
#Inspect access tokens, if access token and app id both are valid then
#proceed otherwise return error message
my $inspection_request_url = "https://graph.facebook.com/debug_token?
input_token=$access_token&access_token=$app_token";
my $inspection_response = LWP::Simple::get($inspection_request_url);
my $inspection_response_decoded = decode_json($inspection_response);
```

The Graph API endpoint takes two required parameters:

- `input_token`: an access token which is to be inspected
- `access_token`: the app access token, which can be retrieved by using Graph API explorer tool or by making HTTP GET request as follows:

```
#create facebook app token in order to use in access token inspection process
sub app_token{
  my $app_token_url = "https://graph.facebook.com/oauth/access_token?client_id
=$fb_app_id&client_secret=$fb_app_secret&grant_type=client_credentials";
  my $app_token_response = LWP::Simple::get("https://graph.facebook.com/oauth/
access_token?client_id=1594762367411421&client_secret=
4f35123094f9077e1b51e0e2bfaea32b&grant_type=client_credentials");
  my @app_token_array = split /=/, $app_token_response;
  my $app_token = $app_token_array[1];

  return $app_token;
}
```

The access token is valid if the app id and user id returned from inspection response are equal to the id of testFbApp and the previously fetched user's id respectively. If the access token is validated, user with the same Facebook information is checked. And handling of three use cases mentioned earlier (logging in, signing up, linking) are carried out using the state parameter value. Each of these use cases are discussed in the following sub sections. If the access token is not validated, inspection failed message is returned.

```
if($fb_user_info->{id} == $inspection_response_decoded->{data}{'user_id'}
&& $inspection_response_decoded->{data}{'app_id'} == $fb_app_id ){

  my $user;
  $user = user('external_source_info', {ext_user_id => $fb_user_info->{id}});
  my $fb_user_person_id = $user->{person_id};
  my $user_info;
```

7.3.1 Log in with Facebook

If the value of state parameter is equal to 'fb_login', it is assumed that the user wants to log-in with Facebook and he/she already has an account in the test website. There can be two main scenarios. The first one is that the user has already used Facebook login with test website. If so his/her Facebook information is already stored in the database. In this case, the user is logged into the website and his/her session is started. As a session variables the user's username and user id fetched from database are stored.

```

#if login with fb was clicked
if($fb_case eq "fb_login"){
    #if retrieved fb account is found in database, the user should be logged in
    if(%{$user}){
        $user_info = user('person', {person_id => $fb_user_person_id});

        session logged_in_user => $user_info->{user_name};
        session logged_in_user_id => $user_info->{person_id};
        redirect "/dashboard";
    }
}

```

In the second scenario, the user's Facebook information is not found in the database. It means that the user is either a new user or an existing one who had not yet used Facebook login. To handle this situation, first it is checked if the user had granted permission to access his/her email address or not. If email access was not granted then the user is redirected to home page and appropriate message is shown like this:

```

#if email access permission is denied during fb login, ask the user
#to take appropriate action.
else{
    flash message => 'Account not found. Please log in using your
    username and password and link your facebook account or go to
    registration page and click sign up with facebook button first.';
    redirect '/';
}

```

If email address is retrieved from Facebook, existence of user with same email address in the person table is checked. If the user exists, he/she is signed in to the website and his/her Facebook related information are saved in the external_source_info table. Otherwise he/she is redirected to the index page with same message like the one used in case of email access permission deny.

```

#check user in person table with fb retrieved email address
if($fb_user_info->{email}){
    $user_info = user('person', {email_address => $fb_user_info->{email}});

    #if the user exists simply link his fb a/c and log the user in
    if($user_info){
        session logged_in_user => $user_info->{user_name};
        session logged_in_user_id => $user_info->{person_id};

        database('main')->quick_insert('external_source_info', {
            person_id => session->{logged_in_user_id},
            ext_username => $fb_user_info->{name},
            ext_source => "Facebook",
            context_label => "",
            ext_user_id => $fb_user_info->{id},
        });
        redirect "/dashboard";
    }
}

```

7.3.2 Sign up with Facebook

If the value of state parameter is equal to 'fb_registration', then it is assumed that the user wants to create an account in the website using Facebook login. Even though user is assumed to be the new one, it is necessary to check the existence of the Facebook account. Because it is possible that the user has already used Facebook login with the website but he/she forgot about that. In this kind of situation, to prevent from creating multiple accounts for the same user, its existence is checked. If the user with same Facebook information found, he/she is redirected to index page with the message saying that he/she already has an account and should use login with Facebook button instead.

```
if(%{$user}){
    flash message => "You already have linked your facebook account with Viope.
    Please click login with facebook button.";
    redirect '/';
}
```

If the Facebook account does not exist, it means either a new account has to be created or simply link the Facebook account to the existing account and log the user in. In order to link Facebook account, user must have granted email address access and he/she must have used same email address in both test site and Facebook.

```
#If a user with same fb id is not found
else{
    #check if email address is retrieved from fb
    if($fb_user_info->{email}){
        #check the user is in person table or not using email address retrieved
        #from fb
        $user_info = user('person', {email_address => $fb_user_info->{email}});

        #if a user with fb retrieved email exists, then link this fb a/c to
        #that user and log the user in
        if($user_info){
            session logged_in_user => $user_info->{user_name};
            session logged_in_user_id => $user_info->{person_id};

            database('main')->quick_insert('external_source_info', {
                person_id => session->{logged_in_user_id},
                ext_username => $fb_user_info->{name},
                ext_source => "Facebook",
                context_label => "",
                ext_user_id => $fb_user_info->{id},
            });
            redirect "/dashboard";
        }

        else{
            #show the pre-filled registration form and ask for username
            template 'register.tt', {ext_user_info => $fb_user_info,
                ext_provider => "Facebook"};
        }
    }
}
```


It is possible that the user had denied email access permission. And it is also possible that the email address access was granted but the account with that email does not exist in the database. In both cases a pre-filled registration form is displayed to the user and asked to enter username and email address (if access to it was denied during Facebook login). Once he/she pass the required information and submit the form, new account is created and the Facebook account is associated with the newly created account.

```
#user registration
post '/register' => sub {
  my $salted_passwd;

  #if password is provided, hash it other wise save empty string in
  #password column in database
  if(params->{password}){
    $salted_passwd = salted_hash(params->{password});
  }
  else{
    $salted_passwd = "";
  }

  #only add the user if it does not exists in database
  my $check_user = user('person', {user_name => params->{username}});

  if (!$check_user){
    database('main')->quick_insert('person', {
      user_name => params->{username},
      password => $salted_passwd,
      first_name => params->{first_name},
      last_name => params->{last_name},
      email_address => params->{email},
      street_address => params->{address},
      gender => params->{gender},
      phone_number => params->{phone_number},
    });

    #retrieve newly registered user's info to get person_id
    my $user = user('person', {user_name => params->{username}});

    #if registration process initiated from social login then save
    #social info
    if(params->{ ext_user_id }){
      database('main')->quick_insert('external_source_info', {
        person_id => $user->{person_id},
        ext_username => params->{ext_user_name},
        ext_source => params->{ext_provider},
        context_label => "",
        ext_user_id => params->{ext_user_id},
      });
    }

    #start a session for this newly created user
    session logged_in_user => params->{username};
    session logged_in_user_id => $user->{person_id};
    flash message => 'You are logged in.';
    redirect "/dashboard";
  }

  else{
    return "A user with ".params->{username}." already exists. Please
    use another username.";
  }
};
```

7.3.3 Link Facebook account

If the value of state parameter is equal to 'facebook_link', it is assumed that the user is already logged in and his/her Facebook account has not been associated yet. As in other scenarios, first the user with same Facebook id is checked in the database. This check might seem un-necessary because only those users are shown a Facebook link button who's account is not yet linked. But it is needed to handle some situations which might rarely happen. For instance, while I was testing Facebook link button, I was logged in as a user named test_user, who had not yet linked his Facebook account. And clicked the link Facebook button. test_user was automatically linked to Facebook account. He was not even redirected to Facebook login page. What actually happened was, before logging in as test_user, I just registered a new user, first_fb_user, using sign up with Facebook button and logged out from the test website. But first_fb_user was still logged in in Facebook. When test_user clicked link Facebook button he was automatically linked to the first_fb_user's Facebook account.

So to prevent from linking the same Facebook account to many users, existence of Facebook retrieved user is important to check. If it exists then user is informed that the Facebook account has already been linked and suggest the user what might have happened. Otherwise the user's Facebook account is linked to his/her account.

```
#If a user is found in database with same fb id
if(#{ $user }){
    flash message => "This facebook account has already been linked with other
    account. Check if someone else is logged into facebook in your browser";
    redirect '/dashboard';
}
else{
    #if that facebook id is not found, add that users fb info in
    #external_source_info table and redirect to dashboard
    database('main')->quick_insert('external_source_info', {
        person_id => session->{logged_in_user_id},
        ext_username => $fb_user_info->{name},
        ext_source => "Facebook",
        context_label => "",
        ext_user_id => $fb_user_info->{id},
    });
    redirect "/dashboard";
}
```

7.3.4 Unlink Facebook account

For some reason if a user wants to unlink his/her Facebook account, he/she should be able to do that easily. It can be done by providing the unlink button in the appropriate place. When user sends unlink request the script removes the user related Facebook data.

```

#unlink external authentication service
get '/unlink/:ext_provider' => sub{
  if(session->{logged_in_user}){
    #delete the user's external source information and redirect
    #to dashboard
    database('main')->quick_delete('external_source_info', {
      person_id => session->{logged_in_user_id}, ext_source
      => params->{ext_provider});

    redirect '/dashboard';
  }
};

```

8 Evaluation

The thesis work as a whole is successful. Throughout the thesis work all of the set objectives were achieved. Viopé's CTO told that this thesis work has been useful and he was satisfied with the end result. The thesis work delivered what the company had expected.

During the course of thesis work, few challenges were faced. Although there were some difficulties, the company has managed to provide the required resources and guidance which helped me achieve some of the objectives. Most of the technologies used for this project were completely new to me. I had no previous experience with programming in Perl, server setup and Debian distribution of Linux operating system.

With the completion of this thesis, I have gained a confidence in working with completely new technologies. I have learned several new and important skills which will definitely help me become a better professional. The main technical skills I have learnt are Web development in Perl, server setup, PostgreSQL database, Git, Linux and OAuth. Besides these technical skills, I have also improved my research and documentation skills from this thesis work.

References

- About Git. 2015. About Git. Accessed 18 February 2015.
<http://git-scm.com/about/branching-and-merging>
- About Twitter. 2015. Company. Accessed 23 February 2015.
<https://about.twitter.com/company>
- About Viope. 2014. About. Accessed 20 December 2014.
<https://www.viope.com/#/about>
- The Debian Linux FAQ. 2014. The Debian GNU/Linux FAQ Chapter 1- Definition and Overview. 16 October. Accessed 24 February 2015.
https://www.debian.org/doc/manuals/debian-faq/ch-basic_defs.en.html#s-what-is-debian
- Developers Facebook. 2015. Using Facebook Login with Existing Login Systems. Accessed 18 February 2015.
<https://developers.facebook.com/docs/facebook-login/multiple-providers>
- Drebes, L. 2012. How Social Login Is Changing Business—and Your Privacy. 28 February. Forbes. Accessed 25 December 2014.
<http://www.forbes.com/sites/forbesleadershipforum/2012/02/28/how-social-login-is-changing-business-and-your-privacy/>
- eMarketer. 2014. Facebook Flies High in Finland. 11 September. Accessed 11 January 2015.
<http://www.emarketer.com/Article/Facebook-Flies-High-Finland/1011193/2>
- Eriksson, P & Kovalainen, A. 2008. Qualitative Methods in Business Research. London: SAGE
- Facebook Statistics. 2014. Facebook Newsroom. Accessed 28 December 2014.
<http://newsroom.fb.com/company-info/>
- Finns favor Facebook. 2015. Finns favour Facebook. Taloustutkimus survey for Yle. 04 January. Accessed 11 January 2015.
http://yle.fi/uutiset/finns_favour_facebook/7719008
- Florencio, D & Herley, C. 2007. A Large-Scale Study of Web Password Habits. Microsoft Research. Accessed 23 December 2014.
<http://research.microsoft.com/pubs/74164/www2007.pdf>
- Gartner. 2013. Gartner Says Half of New Retail Customer Identities Will Be Based on Social Network Identities by 2015. February 5. Accessed 23 December 2014.
<http://www.gartner.com/newsroom/id/2326015>
- Survey Monkey. 2012, Social Privacy Survey Results. Survey Monkey. Accessed 25 December 2014.
<http://www.slideshare.net/Gigya/social-privacy-survey-results>
- Goings, K. & Abel, P. 2013. The value of social login: Solving the Engagement Gap. 9 October. Accessed 23 December 2014.
<http://www1.janrain.com/rs/janrain/images/industry-research-value-of-social-login-2013.pdf>
- LinkedIn Press. 2014. LinkedIn reaches 300 million members worldwide. 18 April. Accessed 29 December 2014.
<https://press.linkedin.com/site-resources/news-releases/2014/linkedin-reaches-300-million-members-worldwide>

Loginradius.2014. Q3 2014 Report: Social Login and Social Sharing Trends. Accessed 11 January 2015.

http://landing.loginradius.com/hs-fs/hub/436275/file-1991698711-pdf/PDF/statistical-analysis-on-global-social-login-and-sharing-trends-q3-2014.pdf?utm_campaign=Premium%20Contents&utm_medium=ebook%20-%20social-login-and-sharing-trends-q3-2014&utm_source=Ebooks

Marketing Terms. 2014. Conversion Rate. Accessed 24 December 2014.

http://www.marketingterms.com/dictionary/conversion_rate/

Nginx. 2015. nginx. Accessed 18 February 2015.

<http://nginx.org/en/>

Nummela, T. 2015. SUOMITWITTER: Calculating Finnish tweeps since February 7th 2013. Accessed 17 February 2015.

<http://www.toninumela.com/suomi-twitter/>

Oauth. 2007. Oauth Core 1.0. 4 December. Accessed 27 December 2014.

<http://oauth.net/core/1.0/>

OAuth 2.0. 2012. The OAuth 2.0 Authorization Framework. Internet Standard Track Document. Accessed 23 February 2015.

<http://tools.ietf.org/html/rfc6749#section-1.2>

Olson, M. 2014. Social Login Trends Across the Web for Q3 2014. Posted 8 October. Accessed 10 January 2015.

<http://janrain.com/blog/social-login-trends-q3-2014/>

Peles, O & Hay, R. 2014. SpoofedMe - Intruding Accounts using Social Login Providers: A Social Login Impersonation Attack. 3 December. Accessed 26 December 2014.

<http://www.slideshare.net/ibmsecurity/spoofed-me-socialloginattack>

Perl Database Access. 2015. Tutorialpoint.com. Accessed 20 February 2015.

http://www.tutorialspoint.com/perl/perl_database.htm

Perl Dancer. 2015. Homepage. Key features. Accessed 18 February 2015.

<http://perldancer.org/>

Pickard, A. 2008. Are you suffering from password pressure?. 17 January. Accessed 13 January 2015.

<http://www.theguardian.com/technology/2008/jan/17/security.banks>

Pönkä, H. 2014. Suomalaiset Facebookissa 2011-2014 - käyttäjämäärät ja kuvaajat. Posted 14 January. Accessed 17 February 2015.

<https://harto.wordpress.com/2014/01/14/suomalaiset-facebookissa-2011-2014-kayttajamaarat-ja-kuvaajat/>

PostgreSQL. 2015. Postgresql 9.4.1 Documentation. Accessed 19 February 2015.

<http://www.postgresql.org/docs/current/static/intro-what-is.html>

Salyer, P.2012. Social Login and Social Plugins Increase Page Views/Time Spent on Site. Posted 18 January. Accessed 24 December 2014

<http://blog.gigya.com/social-login-and-social-plugins-increase-page-views-time-spent-on-site/>

Sina Weibo. 2015. Business Overview. Accessed 30 December 2014.

<http://ir.weibo.com/phoenix.zhtml?c=253076&p=irel-homeProfile&t=&id=&>

Service now. 2015. External Authentication (Single Sign-On-SSO). Last modified 30 January. Accessed 22 February 2015.

[http://wiki.servicenow.com/?title=External_Authentication_\(Single_Sign-On_-_SSO\)](http://wiki.servicenow.com/?title=External_Authentication_(Single_Sign-On_-_SSO))

Tencent Weibo. 2014. Tencent Weibo Website. Page in Chinese translated into English using Google translate. Accessed 30 December 2014.

<http://dev.t.qq.com/websites/>

W3Techs. 2014. Usage statistics and market share of Wordpress for websites. Accessed 1 January 2015.

<http://w3techs.com/technologies/details/cm-wordpress/all/all>

Jing, W. 2014. Covert Redirect Vulnerability Related to OAuth2.0 and OpenId. 07 May. Accessed 02 January 2015.

http://tetrapp.com/covert_redirect/oauth2_openid_covert_redirect.html

Wang, R., Chen, S. & Wang, X. 2012. Signing Me onto Your Account through Facebook and Google: a Traffic-Guided Security Study of Commercially Deployed Single-Sign-On Web Services. Accessed 26 December 2014.

<http://research.microsoft.com/pubs/160659/websso-final.pdf>

Gigya Infographics. 2014. THE LANDSCAPE OF SOCIAL LOGIN Q3-2014 (July-September). Accessed 23 February 2015.

<http://info.gigya.com/rs/gigya/images/Q3%20Social%20Login%20Infographic.png>

Xing. 2014. XING is the social network for business professionals. Accessed 01 January 2015.

https://corporate.xing.com/no_cache/english/company/xing-ag/

Yle. 2013. Täällä somelaiset elävät - katso lista historiallisesta Facebookista juuri avattuun Pheediin. 5 March. Accessed 17 February 2015.

http://yle.fi/uutiset/taalla_somelaiset_elavat_-_katso_lista_historiallisesta_facebookista_juuri_avattuun_pheediin/6518189

Figures

Figure 1: Basic external authentication process.....	9
Figure 2: Perl sample code	16
Figure 3: Sample script written using Dancer framework	17
Figure 4: HTML code sample	19
Figure 5: Social login trend analysis by Janrain	24
Figure 6: Social login trend report by LoginRadius.....	25
Figure 7: Architecture of a DBI application	32
Figure 8: Homepage of test site	33
Figure 9: Dashboard page of test site	34
Figure 10: Registration page of test website.....	34

Tables

Table 1: List of social media EAS providers and basic user information they provide 21

Table 2: List of non-social media EAS providers and basic user information they provide

..... 23

Appendixes

Appendix 1: Social login trend Q3 2014 Gigya Infographics	50
Appendix 2: Screenshots of test website	52
Appendix 3: Tables used in database of test website	53

Appendix 1: Social login trend Q3 2014 Gigya Infographics

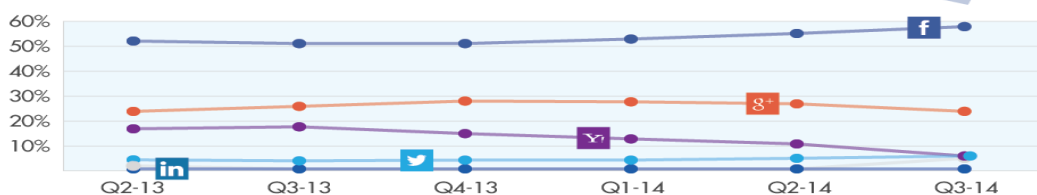
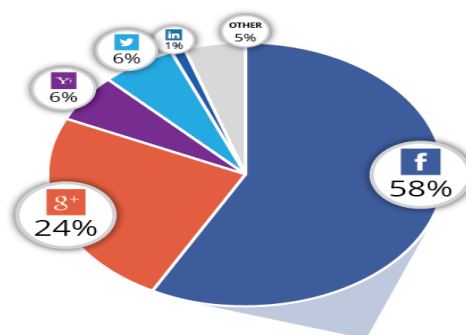
THE LANDSCAPE OF SOCIAL LOGIN

Q3 - 2014 (July - September)

In the below infographic, Gigya, the Connected Consumer Management Suite, breaks down how consumers used Social Login across the web and mobile devices by industry vertical and region in Q3 2014 (July - September).

SOCIAL LOGIN PREFERENCES

Facebook continued solidifying its position as the dominant identity provider on the web in Q3 2014. The social network's updates to Facebook Login earlier this year seem to be making a positive impact on Facebook identity across the web, with users now able to select the data they share with sites and apps with line-by-line controls. Meanwhile, Google/Google+ saw a slight dip in its usage in Q3, dipping below 25% for the first time in over a year. Yahoo, once a prominent identity provider, dipped well under 10% for the first time in nearly three years.

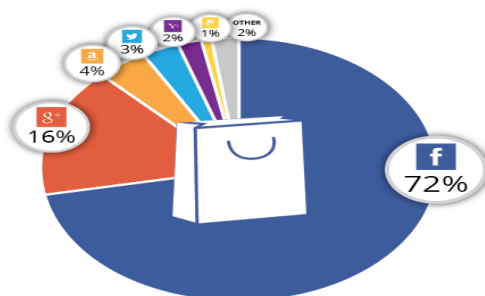


MOBILE DEVICES

Facebook saw a slight decline in mobile logins in Q3, losing 2% share of logins over the previous quarter. Google/Google+ made modest gains, gaining 3% over Q2, while Yahoo's slide was accentuated on mobile, with the company dipping to just 1% of logins.

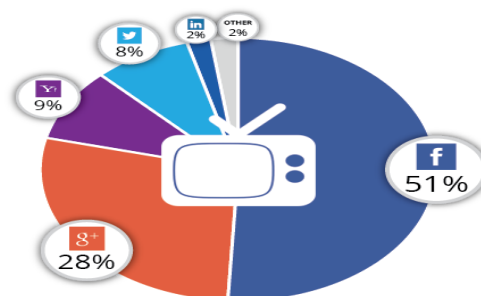
ECOMMERCE

While still maintaining its dominant position, Facebook saw a slight decline as an identity provider on Ecommerce sites and apps, while Google/Google+ stayed steady over Q3. Amazon, meanwhile, reached its highest share of logins since the release of Login with Amazon in May 2013.



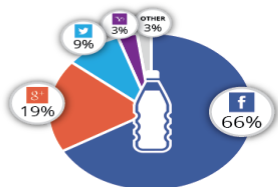
MEDIA/PUBLISHERS

Facebook cinched its majority position on Media & Publisher sites/apps in Q3 - the first time the world's largest social network has eclipsed 50% in more than a year. Google/Google+ and Twitter remained relatively steady over Q2. Yahoo dipped to under 10% - a 9 point decrease over Q3 2013.



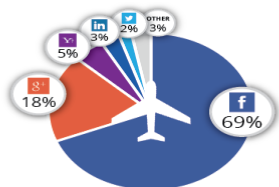
CONSUMER BRANDS

Facebook, Google/Google+ and Twitter all remained mostly steady in Q3 as identity providers on Consumer Brand properties.



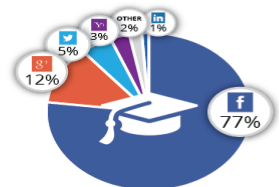
TRAVEL/HOSPITALITY

Facebook remained as the dominant identity provider on Travel & Hospitality web properties in Q3, accounting for more than two-thirds of all social logins. Google/Google+ also kept its percentage from Q2 while no other identity provider broached double digits.



EDUCATION/NON-PROFIT

Facebook's dominance was even more pronounced in the Education/Non-profit vertical, with the world's largest social network accounting for more than three-quarters of all social logins.



SOCIAL LOGIN PREFERENCES ACROSS THE WORLD



NORTH AMERICA

Facebook made incremental gains as an identity provider in North America in Q3, while Google saw a slight dip and Yahoo's share was cut in half from the previous quarter. Twitter, meanwhile, made substantial gains, adding 6% over the previous quarter.



EUROPE (INCLUDES RUSSIA)

Facebook and all other identity providers maintained their relative positions in Europe in Q3. While European users continue to use Facebook Login at a higher rate than North American consumers, the gap has narrowed to only 7%.



CENTRAL & SOUTH AMERICA

Facebook continued to be the overwhelmingly popular identity provider in Central and South America in the last quarter, with 8 in 10 consumers opting to log in via Facebook over other social login options.



AFRICA/MIDDLE EAST

Facebook lost a significant share of logins in Africa & the Middle East during Q3, dropping a surprising 11 points over Q2, while usage of Google/Google+ more than tripled to account for two-thirds of logins during the quarter.



ASIA/PACIFIC

Twitter supplanted regional networks Sina and QQ in Asia-Pacific during Q3, accounting for a higher percentage of logins than the two other networks combined. Facebook maintained its dominant position, but Google/Google+ also gained share of logins.

Source: Gigya Customer Implementations Q3- 2014



Gigya's Connected Consumer Management Suite enables the world's largest brands, including Pepsi, Verizon and ABC to understand and connect more closely with today's mobile and socially connected consumers. Our technology helps businesses access, consolidate and manage permission-based identity and behavior data, while providing deep customer insights that turn data into action.

Find out more at: www.gigya.com | 650.353.7230

Appendix 2: Screenshots of test website

There are 3 users in the database.

Name	Address	Email	Phone
demo user1	some Address	demo@user1.com	2352352
demo2 user2	some address 2	demo@user2.com	345343
test user	test address	test@user.com	52345346346

[Logout](#)

Powered by [Dancer](#) 1.3095

You are not authorize to see this page.

Welcome test, This is your Dashboard.

Available Social login providers:

External Login Providers	Status	Action
Facebook	Linked	Unlink Facebook
Moodle	Not Linked	Link Moodle

[Logout](#)

Powered by [Dancer](#) 1.312

Registration form

Username:

Email:

Address:

First Name:

Last Name:

Gender:

Phone Number:

Username is required to complete registration.

Powered by [Dancer](#) 1.312

Appendix 3: Tables used in database of test website

```
CREATE TABLE person
```

```
(  
  person_id integer NOT NULL DEFAULT nextval('seq_person_id'::regclass),  
  user_name character varying(50) NOT NULL,  
  password character varying(50),  
  first_name character varying(50),  
  last_name character varying(50),  
  street_address character varying(50),  
  postal_code character varying(20),  
  city character varying(50),  
  country_id character varying(2) NOT NULL DEFAULT 'aa'::character varying,  
  phone_number character varying(20),  
  email_address character varying(80),  
  gender character varying(10),  
  insert_time timestamp without time zone NOT NULL DEFAULT now(),
```

```
  CONSTRAINT pk_person PRIMARY KEY (person_id)  
);
```

```
CREATE TABLE external_source_info
```

```
(  
  person_id integer NOT NULL,  
  ext_username text NOT NULL,  
  ext_source text NOT NULL,  
  Ext_user_id bigint NOTNULL,  
  CONSTRAINT pk_external_source_info PRIMARY KEY (person_id, ext_username, ext_source),  
  CONSTRAINT fk_external_source_info_person FOREIGN KEY (person_id)  
    REFERENCES person (person_id)  
);
```