Palestine Polytechnic University

College of Information Technology and Computer Engineering

Developing Data Collection Tools for Measuring the User Behavior in Facebook

Submitted by:

Meqdad Ahmad Darweesh

Abdulhameed Wael Abuzanunah

Hamzah Sameer Bahar

Supervisor:

Dr. Hani Salah

This report was submitted in partial fulfillment of the requirements for
the degree of Bachelor of Engineering in Computer Systems Engineering

2018

Table of Contents

Table of Figures

List of Tables

Abstract

Facebook is the most popular Online Social Network (OSN) worldwide. It has grown from one billion users in the third quarter of 2012 to two billions in the second quarter of 2017. The usage of Facebook has developed from contacting friends to share personal information, and it also became away for people to share, recommend, and link together all kinds of information.

Facebook provides a huge set of behavioral data that if analyzed will produce different kinds of useful information that could be benefited from in understanding human behavior in depth, and impact design choices of social networks and applications. In addition to that, analyzing these information help users to discover what they use Facebook for, and how much time they spend using it. However, user behavior is not understood enough yet, because accessing this type of data is limited.

By behavioral data we mean how the user interacts with Facebook functionalities. For example, how many likes, comments, shares and other similar functions, in addition to the content these functions applied to. In other words, we need to collect user interactions with which content together with respective timing information. In particular, we focus on four types of behavioral data: user's actions, user's friend list, activity log, and basic demographical information. In this project we aim to update and redesign an outdated software tools that were used in some previous similar studies [1]and [2].

Towards that end, we develop two software tools: First, Facebook Privacy Watcher and Analyzer (FPWA) which is a Browser extension (plugin) -specifically for chrome-. Second, Facebook Activity Log Collector (FALC) which is Android application for smartphones' users. The plugin collects the data of participating users and stores it temporarily in the Chrome browser storage. This enables us to make these data viewable by the user who can then decide whether to send it or not to our server. In the same manner, the Android application also collects user actions, but only the actions which stored in the activity log of Facebook account.

We have a small number of users who participate in our project, according to that, the analysis results won't be comprehensive and representative. In spite of that, we develop a Python program that analyzes any size of samples regardless of the amount of our collected samples.

Ch1: Introduction

1.1: Preface

In recent years, online social networks (OSNs) became a vital part of people's daily life. The most popular OSN is Facebook. Facebook provides different communication and content sharing functionalities. It also helps its users to communicate and socialize with others, share their thoughts, advertise their products to the interested parties in an easy and simple way. So, gathering what people like and what attract their attention will generate a huge set of behavioral data. To serve that purpose we upgraded the previously used concept (Plugin), and we also added another contribution by developing Android application(FALC), since the number of Facebook users who use smart devices cannot be ignored.

1.2: Problem Statement

Nowadays, people publish and share lots of personal details. So, the privacy is important issue in this context. Sometimes users misconfigure the sharing level of the contents they publish. As a result of that, it is important to increase the awareness about privacy and privacy levels of their shared contents.

Many researches study the ability of building peer to peer(P2P) social networks as an alternative to centralized ones like Facebook and Twitter. Collecting behavior data about users on these centralized networks helps in building a robust P2P social network. But accessing that kind of data is limited.

Different old studies concern with non-popular OSN's and many of their results are out of date. In [1] and [2], they concentrate about Facebook in particular and they collect more valuable data about user behavior by overcoming the limitation of accessing data using browser extension (plugin). But many changes happen to Facebook since then, according to that; their software tools are considered out of date and not operable by now.

1.3: Contribution and Solution

Accessing behavioral data is limited. To overcome this limitation, we develop Facebook privacy Watcher and Analyzer (FPWA) chrome extension plugin. By this we are able to collect the needed behavioral data from personal computers (PC's) side. We also add another contribution by developing smart devices application designed using Android for Android smart devices users. Results are obtained after analyzing the collected data using Python scripts written specifically for that purpose.

In order to help Facebook user better control their privacy settings, we implement Colors for Privacy Settings (C4PS) mechanism that was implemented in [1] as Facebook Privacy Watcher (FPW) plugin as the first part of our FPWA plugin. In this, every attribute in the user's Facebook page is given a color, this color indicates who can see that attribute. Color code scheme uses the well-known traffic light colors in addition to blue, which represent custom option. The meanings of colors are: Red: Visible to nobody, Blue: Visible to selected friends, Yellow: Visible to all friends, Green: Visible to everyone.

Monitoring user behavior in Facebook was achieved through Facebook Privacy Analyzer (FPA) in [2]. FPA is implemented as the second part of our FPWA plugin. Which is a browser plugin working at client side, to measure user's activity such as scrolling, sharing, commenting, and other activities. It also has access to user's profile details according to what user has chosen to expose, and it has access to activity log. The data will be stored in the user's device temporarily; this enables us to make these data viewable by the user who can then decide whether to send it or not to our server. User's privacy in this study is granted. i.e. user identity and any other information that may expose its identity will be anonymized. To serve that purpose, we will store user's id encrypted using hash algorithm.

We also develop hybrid Android application which combines native development with web technology. The application launches a WebView to load Facebook page in it, Facebook links are the only links to be loaded into that WebView. The application collects the actions that happened three days earlier, which means that the collected data of one day was generated three days before. The privacy of the collected data is preserved by hashing any number appears and removing any user name. Despite the lack of data samples that we collect, we develop a Python script used to analyze these collected data samples and plots some representative charts and graphs.

1.4: Structure

The rest of the report is structured as follows: In chapter 2, we examine published scientific papers and some data collection tools related to our work. We will illustrate the design options and our chosen design in chapter 3 in addition to implementation part. After that in chapter 4, we will consider analysis part in terms of collected data, analysis details and presenting some preliminary results. Finally, we conclude our report in chapter 5 by summarizing our work and suggesting some directions for future work.

Ch2: Theoretical Background and Related Work

A lot of researches has been done before now about OSNs, they focused on different points, mainly in the fields of computer science and psychology. As previously mentioned above, basically this project depends on two previous papers [1] and [2].

2.1: Literature Review

User behavior in OSN has been studied in the literature by many authors. These studies can be organized by several pillars. Which can be research field like computer science and psychology, the way of gathering data, and the social network that has been worked on, like Facebook, Twitter, LinkedIn.

In this chapter, we focus on related work in the field of computer science about Facebook social Network that evaluate functionality. In order to capture functionality usage of Facebook users, we must have a way for gathering behavioral data of the user, so we focused on related work based on this principle in the remainder of this section.

Some studies collect datasets based on detailed click stream data collection in different OSN's. For example, Benvenuto et al. [3] collect data from four popular OSN's (Orkut, Myspace, Hi5, and LinkedIn), collecting data was through network aggregator website that enables the user to sign into the four networks with single authentication. Likewise, Schneider et al. [4] study how users actually interact with OSNs by extracting clickstreams from passively monitored network traffic. Their experiment was made on Facebook, LinkedIn, Hi5, and StudiVZ. Whereas in our study we rely on data collected from Facebook only that happened at client side using chrome plugin and Android application.

Pasquale et al [5]. analyze in depth the user behavior on three different social sharing systems (Flickr, Delicious, and StumbleUpon). Whereas in our study we focus on single social platform (Facebook). The aim of this study is to evaluate how the features and goals of a social system reflect on the behavior of its users. In contrast our study aims at collecting and analyzing the behavior of Facebook users in order to (1) update the results of similar previous studies that have been done years ago. (2) study the possibility of building DOSN. They study behavior on three platforms because people often subscribe to different platforms, and people can perform of these platforms in similar or different manner depending of the context and the platform itself. However, our study concerns with Facebook only, because our aim is to update the results of previous study in first place.

Backstrom et al. [6] provide a way of analyzing individual's personal networks depending on the way users balance their attention across their social contacts. This study concentrates only on friends. Although it analyzes friends from different perspective than its previous studies (how users balance their attention to contacts), friends still its main purpose. In our study we analyze friends in addition to other attributes.

Ferwerda et al. [7] extract 40 items of different Facebook profile sections that users can disclose or not disclose and ask the participants to indicate to which extent they disclose these items in an online survey. Their results (relationship between nondisclosure behavior of profile sections and personality traits) could be used by applications to infer personality when content data is missing, since personality traits are used by applications designers to provide personalized service to their users. However, in our study we extend that to focus on behavioral data – how user interacts with Facebook functionalities and what user like the most of them- In addition to that, we examine many aspects regarding a specific functionality, for example, games apps, in this we checked how frequently the user uses these apps, and how much time do they spent using them.

The closest works to ours are [1] and [2]. In [1] the authors deal with privacy settings, data oversharing, and access authorization in Facebook, they introduce C4PS (Color for Privacy Settings), which was first developed as mockup for privacy settings in Facebook, this mockup was evaluated in a lab study, and depending on the good results, a Firefox plugin implementing C4PS was developed.

In [2] Thomas et al. study how Facebook users coordinate Facebook functions to gain benefits for themselves. They also consider the changes in user behavior from 2009 till 2014 to conclude user behavior development in that period, they can understand content sharing among users by concentrating on newsfeed as the most important functionality of Facebook. By this they are able to understand content sharing and consumption, they contribute in making literature assumptions about Facebook user behavioral better, which can lead to increase the performance of alternative social networks, they provide evidences for Facebook to be an aging network.

2.2: Related Data collection tools:

There exist many browser's plugins like (Facebook Social Toolkit[1], Social Fixer for Facebook[2], Facebook Unseen[3] and many others) that work to enhance Facebook, by either making the performance or the graphical user interface (GUI) of Facebook page better or adding some functions that are not normally in Facebook. In the same time there is a browser plugin called Data selfie which we can consider as the closest plugin to ours. This plugin tracks Facebook user activity. It tracks everything from what you clicked to what you typed. These data are stored locally and it does not share it. The user can generate a report after a while of usage, this report may contain: (1) information about what someone could learn from your activity (2) the like pages and people who spend most time with (3) It can also show complex thoughts like: how you lean politically, whether you're more relaxed or emotional, your religion.

To the best of our knowledge existing Android applications related to Facebook are only alternatives to the official Facebook application and there is no application that work alongside with Facebook application and collect behavioral data about the user. Examples of such applications include: Fast[4], Friendly for Facebook[5], and Metal for Facebook[6].

[1] Check the link: https://www.plugexsoftware.com [Access time: 19/3/2018 at 07:07 PM]

[2] Check the link: http://socialfixer.com/ [Access time: 22/3/2018 at 10:44 AM]

[3] Check the link: https://chrome.google.com/webstore/detail/unseen-for-facebook/jiomcgpfgkeefipihnplhadgdoollmap [Access time: 22/3/2018 at 11:20 AM]

[4] Check the link: https://play.google.com/store/apps/details?id=app.fastfacebook.com [Access time: 25/3/2018 at 12:14 PM]

[5] Check the link: https://play.google.com/store/apps/details?id=appinventor.ai_Sebachahla.Friendly_alpha8 [Access time: 25/3/2018 at 12:30 PM]

[6] Check the link: https://play.google.com/store/apps/details?id=com.nam.fbwrapper [Access time: 25/3/2018 at 1:10 PM]

Ch3: Analysis and Design

3.1: Requirements

Our project mainly deals with behavioral data of Facebook whose access is restricted and limited to Facebook itself. So, to gain that kind of data we are required to design and implement collection tools to help us reaching that goal. Knowing that the process of developing these tools is not easy as it appears.

Our required behavioral data can be classified into four main types: (1) user actions that may include viewing images in the newsfeed, typing activities (commenting or chatting), and opening and closing Facebook tab, number of clicks on the toolbar at the top of Facebook page, scrolling duration with respect to the session, and much more collected actions. (2) user's friends list and groups with some statistics for gender of friends and users. (3) Permissions that third-party apps may ask to access from the user account. (4) Demographical data, e.g. the gender of the user, in addition to the language of Facebook account and location.

Analysis of previously mentioned collected data is also required, according to that we developed analysis program using Python to handle analysis part despite the lack of number of participants. As a consequence of that, the collected data was not comprehensive nor representative.

3.2: Plugin

3.2.1: Design options

To collect behavioral data from Facebook users, different options are available. First, collecting these data using Facebook API (Graph API) encapsulated into web application. Second, using a software tool (Plugin) that works inside the browser to collect these data while user is using Facebook. Third, developing a desktop application that runs internally as a browser to enable the user to access his account on Facebook.

Table 1: Available design options for data collection (PC's)

Design options	Number of users	Accessing required data	Sharing and publishing	Prerequisites / constraints
Facebook API (Web application)	Limited	Not achievable	Easy	Lots of constraints
Desktop application	Unlimited	Achievable	Not easy	Limited
Plugin	Unlimited	Achievable	Easy	Limited

The Facebook API (sometimes called Graph API) gives us a sustainable method to collect some behavioral data, i.e. it is not affected by Facebook programming changes. While Desktop application and plugin could be affected and stopped in some cases. It is hard to work with Facebook API, Facebook asks you to give them a lot of justifications to allow you to get some data about very limited number of users. Although, we contact them and asked to use their API but the data they provided did not meet our requirements, i.e. some requirements can be extracted from the API but most of our requirements cannot.

3.2.2: Our design

Because of these vulnerabilities for other options, we decided to go forward with plugin completely. Trustiness and publishing are easier in the case of plugin than desktop application, and the plugin do not affect the traditional pattern of usage for Facebook, i.e. developing our special browser won't be comfortable for a lot of users while they are using their Facebook accounts.

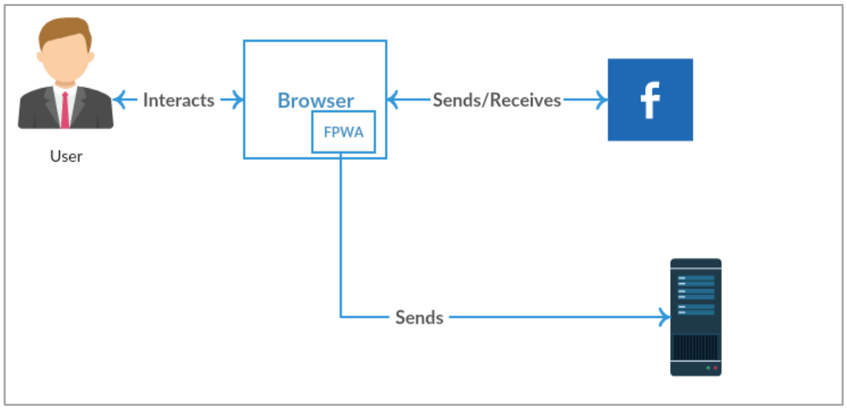


Figure 1: Plugin working principle and required connections

As shown in figure (1) above our plugin (FPWA) is part of the browser, which means that it collects data from browser and not intercept requests travelling from Facebook website to the server of Facebook or vice versa. To accomplish this, we used JavaScript and jQuery with help of chrome browser's API's to get that job done. We were able to detect Facebook page elements from which the requests are to be issued using HTML and Ajax links located in the browser itself.

The collected data is then stored into browser storage to enable us make it viewable to the user as JSON Viewer page. By this, the user will be able to look at what we collected about him and decide to send this data to our server or not depending on three buttons located at the same JSON Viewer page, that determine how frequently the data is to be sent to our server.

We also tried to add another feature to our plugin which is the ability of detecting fake news. In this, we intended to implement this feature as a progress bar such that for each news post -posts that come from news pages or groups- the plugin determines the degree of trustiness of it and reflects it on the progress bar. But we were not able to add that features because of the lake of databases about trusted and not trusted news pages.

3.2.3: Implementation

To work with client side of Facebook, you need to handle the used programming languages in websites that we want to collect our data from. A notification messages library (Toast library) used to notify the user about the progress of gathering process. It's divided to ten stages as follows:

1: It collects the apps of user accounts on Facebook and the used devices for login into his account.

2: It collects the about section in his account, this section contains some information about the user (like: work, education, his favorite quotes …. etc.)

3: It collects the liked pages in Facebook.

4: It collects the joined groups by the user (id of each group to know the number of joined groups, and the id here is hashed).

5: It collects the gender of user.

6: It collects his location by gathering (Longitude & Latitude).

7: It collects the language of Facebook that user customized.

8: It collects the date of birth for the user.

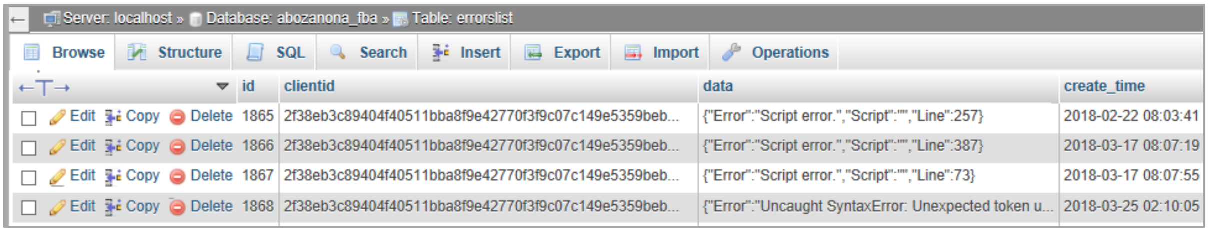
9: It collects his friend's IDs (We hashed the ID's wherever they occur).

10: In this stage, the data arranged to be viewed in JSON format for the user to see the collected data about him.

The security level in our project is considered well, the used hashing algorithm is SHA-256, its implemented using SJCL (The Stanford JavaScript Crypto Library) as published in [8], the security level in this algorithm is acceptable since it produces 64 digits with different combinations (numbers, letters) for each item that we hash[7]. The hashing operation is irreversible. At the same time, it is secure until beyond 2030 according to Stanford university developers[8].

In addition to that, our server is safe. The server is working on Secure Sockets Layer (SSL) protocol, which is the most widely used protocol for implementing cryptography on the web. SSL uses different forms and combinations of cryptographic techniques to provide secure communication over the network.

As mentioned previously about the rapid Facebook changes, we implement a new method to trace the errors and inactive features in the plugin that may be defected after some changes in Facebook website. An error reporting function was added to our plugin source code, it sends a message to our server about the error happened in the plugin (it sends the line number in the code), this message will be stored in a separated database as in the figure (2) below.


Server: localhost » Database: abozanona_fba » Table: errorslist
Browse
Structure
SQL
Search
Insert
Export
Import
Operations

	id	clientid	data	create_time
□ Edit Copy Delete	1865	2f38eb3c89404f40511bba8f9e42770f3f9c07c149e5359beb...	{"Error":"Script error.","Script":"","Line":257}	2018-02-22 08:03:41
□ Edit Copy Delete	1866	2f38eb3c89404f40511bba8f9e42770f3f9c07c149e5359beb...	{"Error":"Script error.","Script":"","Line":387}	2018-03-17 08:07:19
□ Edit Copy Delete	1867	2f38eb3c89404f40511bba8f9e42770f3f9c07c149e5359beb...	{"Error":"Script error.","Script":"","Line":73}	2018-03-17 08:07:55
□ Edit Copy Delete	1868	2f38eb3c89404f40511bba8f9e42770f3f9c07c149e5359beb...	{"Error":"Uncaught SyntaxError: Unexpected token u...	2018-03-25 02:10:05

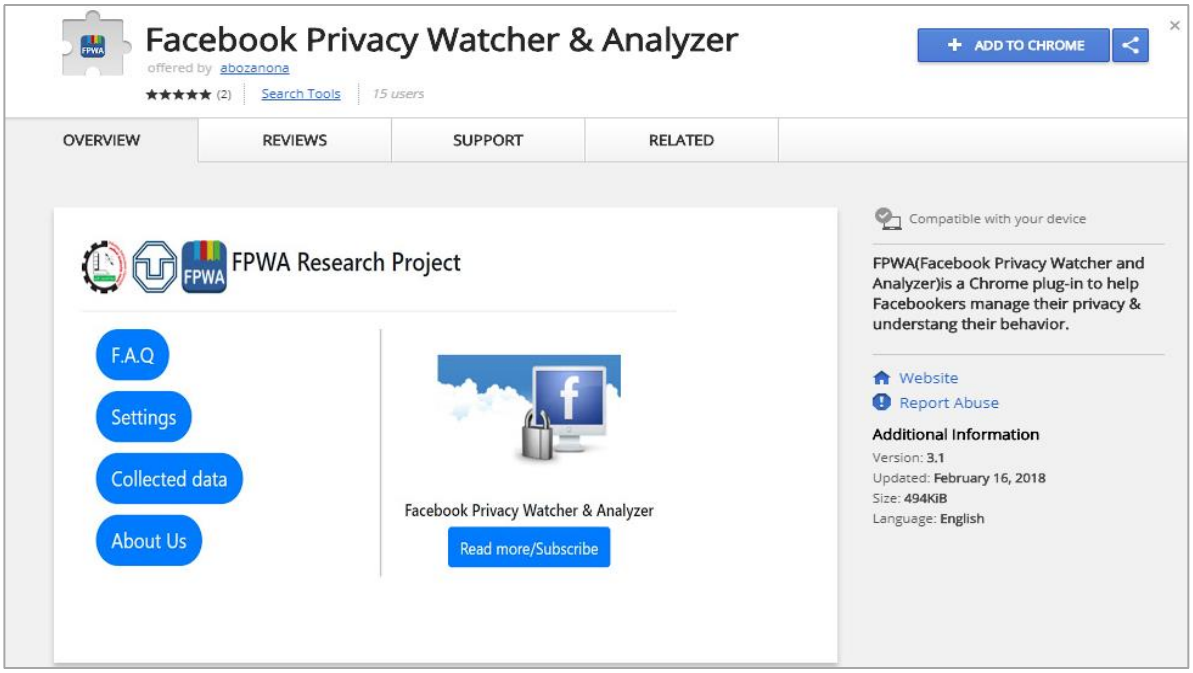
Figure 2: Error list database view on the server

3.2.4: Usage

The interface is simple and understood by different technical levels of plugin's users. Different options in front of the user to choose. First, the user should go to visit the link of downloading plugin from the Chrome Web store – the link of download is mentioned in the website – and installing the plugin in the user Chrome browser as below in figure (3), the user should click on the blue button "ADD TO CHROME" on the top right corner.

[7] Try to hash using SHA-256 at: http://www.xorbin.com/tools/sha256-hash-calculator

[8] Check the link: https://github.com/bitwiseshiftleft/sjcl/wiki/Asymmetric-Crypto [Access time: 13/3/2018 at 04:18 PM]



Facebook Privacy Watcher & Analyzer

offered by abozanona

FPWA(Facebook Privacy Watcher and Analyzer)is a Chrome plug-in to help Facebookers manage their privacy & understang their behavior.

Additional Information

Version: 3.1

Updated: February 16, 2018

Size: 494KiB

Language: English

Figure 3: Chrome Web store installation interface

After installing the plugin by clicking on the “ADD TO CHROME” button, the main interface will appear to the user as in figure (4) below. Many options offered for the user to choose from this interface.

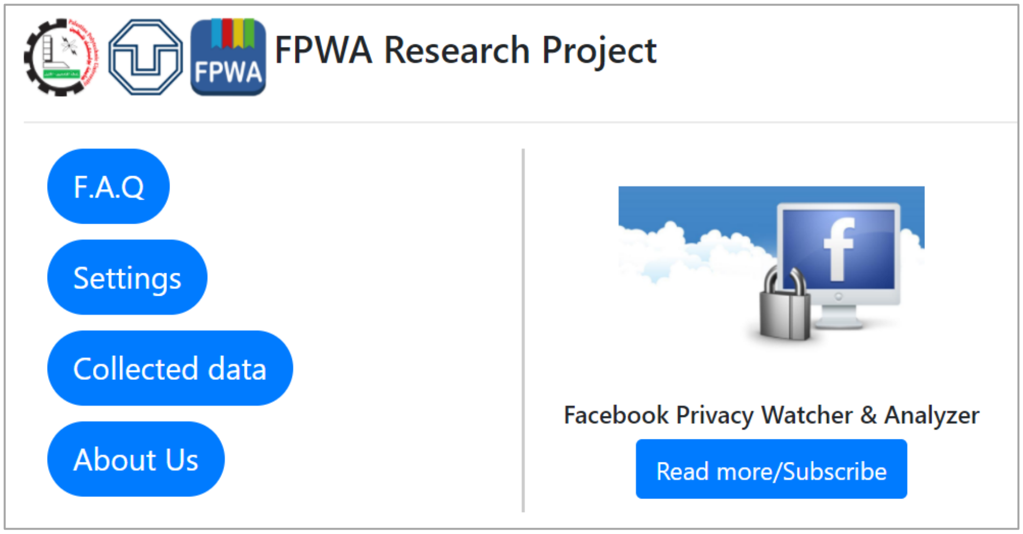
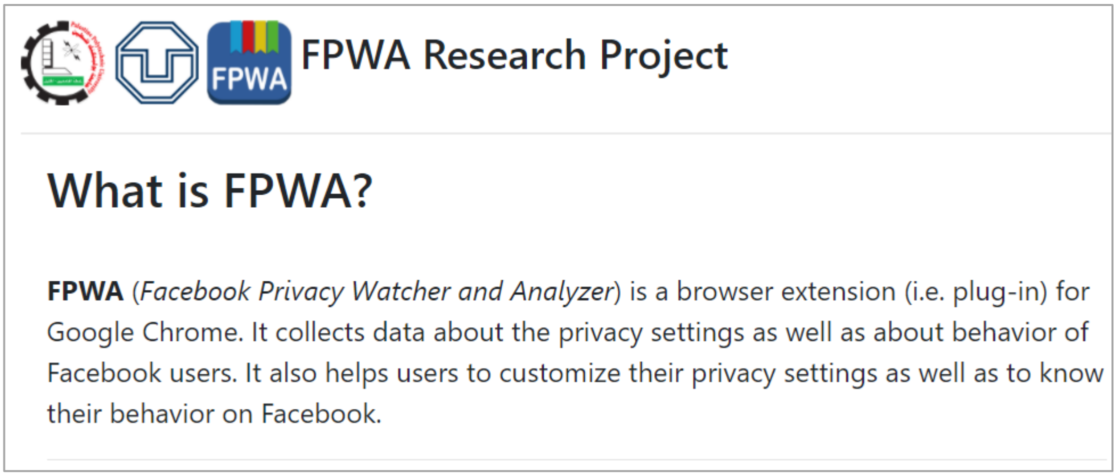


Figure 4: Main interface of the plugin after installation.

- **FAQ**: The user can read some FAQs that may jumps in his mind about different things in plugin/project – all FAQs are in appendix A – here is an example below in the figure (5).



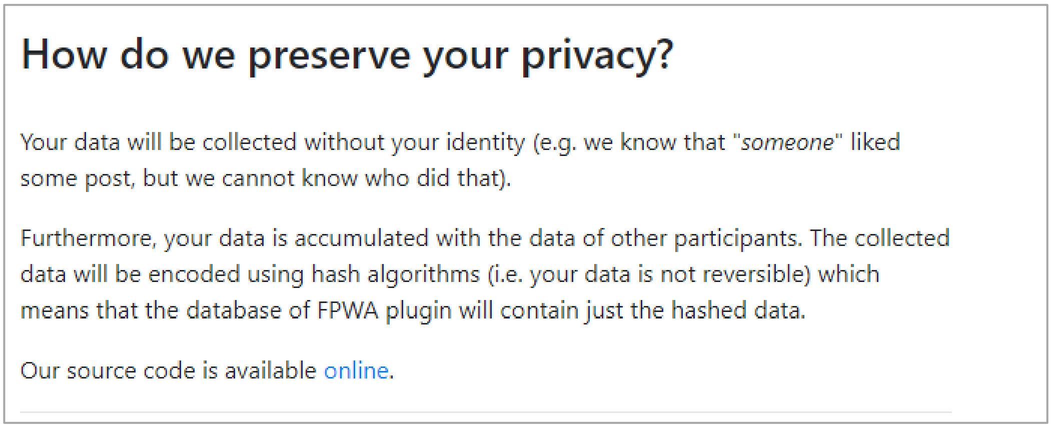
FPWA Research Project

What is FPWA?

FPWA (*Facebook Privacy Watcher and Analyzer*) is a browser extension (i.e. plug-in) for Google Chrome. It collects data about the privacy settings as well as about behavior of Facebook users. It also helps users to customize their privacy settings as well as to know their behavior on Facebook.

Figure 5: FAQ1 – Simple definition for the plugin and the work principle

Another FAQ interface is "How do we preserve your privacy?" with a short answer below the question as in figure (6).



How do we preserve your privacy?

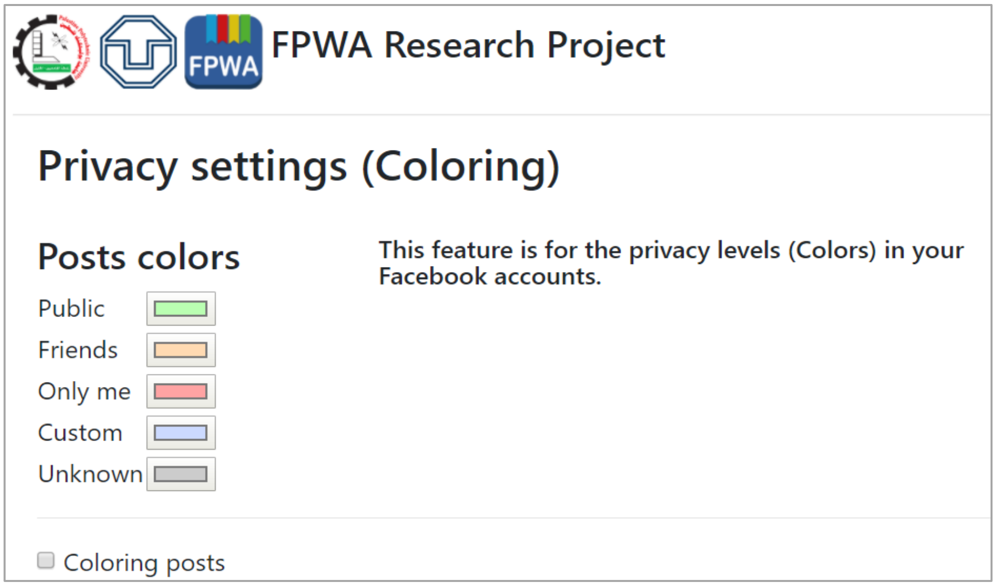
Your data will be collected without your identity (e.g. we know that "*someone*" liked some post, but we cannot know who did that).

Furthermore, your data is accumulated with the data of other participants. The collected data will be encoded using hash algorithms (i.e. your data is not reversible) which means that the database of FPWA plugin will contain just the hashed data.

Our source code is available online.

Figure 6: FAQ2 - How do we preserve your privacy

- **Settings**: The user can enable or disable coloring posts feature, enabling this feature makes the posts of the user newsfeed colored depend on the shown system as in figure (7) below.



FPWA Research Project

Privacy settings (Coloring)

Posts colors

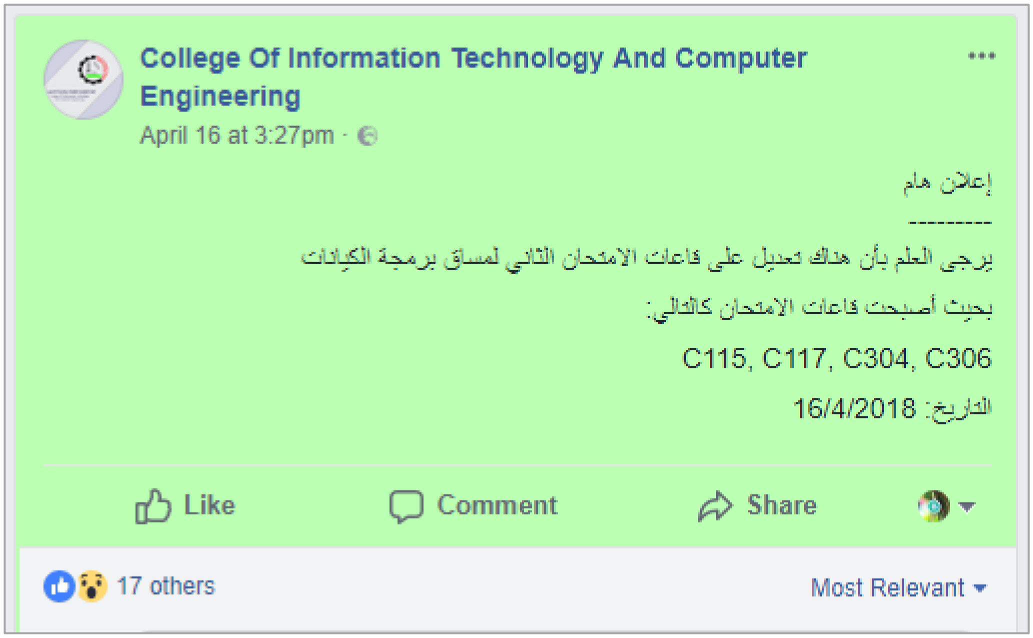
Public	🟩
Friends	🟧
Only me	🟥
Custom	🟦
Unknown	⬜

This feature is for the privacy levels (Colors) in your Facebook accounts.

☐ Coloring posts

Figure 7: Settings – Privacy coloring settings interface to enable or disable the coloring posts feature.

"Public" posts will be colored with Green as in figure (8), and the "Only me" posts will be colored to Red as in figure (9).



College Of Information Technology And Computer Engineering

April 16 at 3:27pm ·

إعلان هام

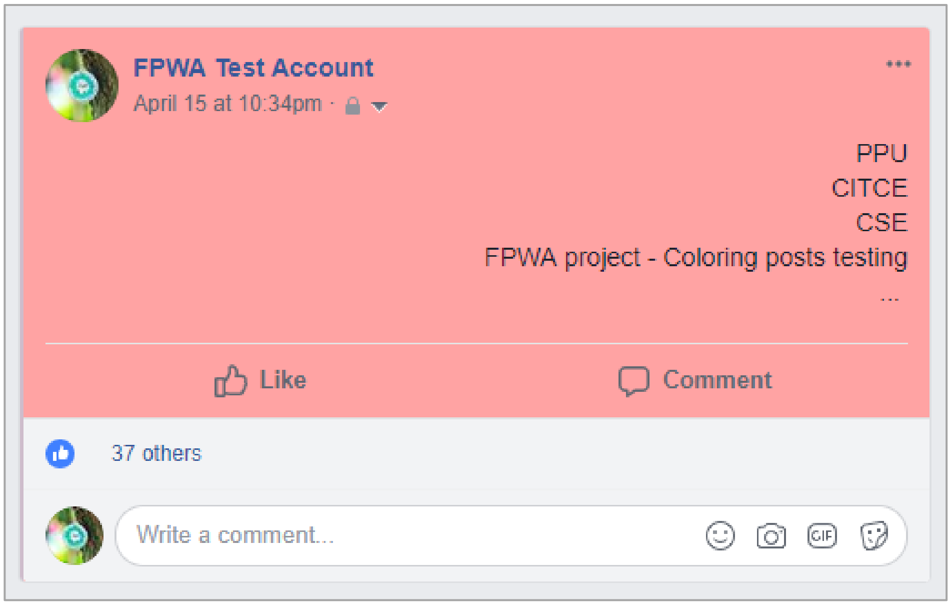
يرجى العلم بأن هناك تعديل على قاعات الامتحان الثاني لمساق برمجة الكيانات

بحيث أصبحت قاعات الامتحان كالتالي:

C115, C117, C304, C306

التاريخ: 16/4/2018

Figure 8: Active coloring posts feature - Public post colored Green



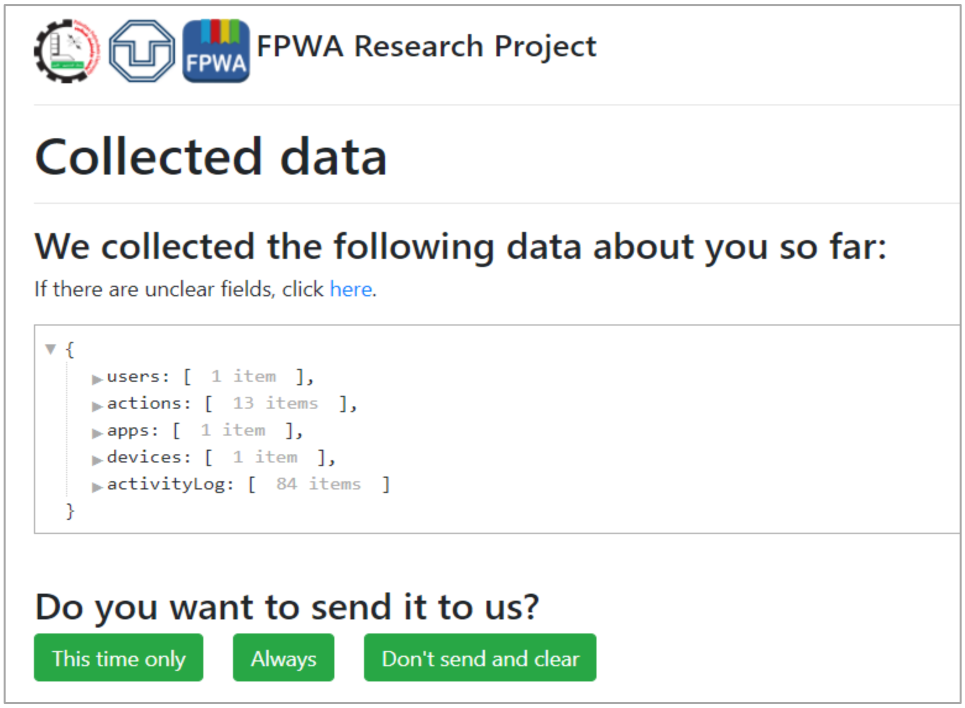
FPWA Test Account

April 15 at 10:34pm ·

PPU
CITCE
CSE
FPWA project - Coloring posts testing
...

Figure 9: Active coloring posts feature - Only me post colored Red

- **Collected data**: The user can view the collected data about his account as in figure (10). Multiple choices for sending data to the server are offered, the user can cancel the sending operation by clicking on “Don’t send and clear” or the user can click on “Always” to make the sending operation works automatically (sending the data every day without user interaction) and finally, the user can decide to send or not for each time by clicking on “This time only”. If the user selects this time only, the collected data screen as in figure (10) will appear again after 24 hours from last sending operation to select his new choice.



FPWA Research Project

Collected data

We collected the following data about you so far:

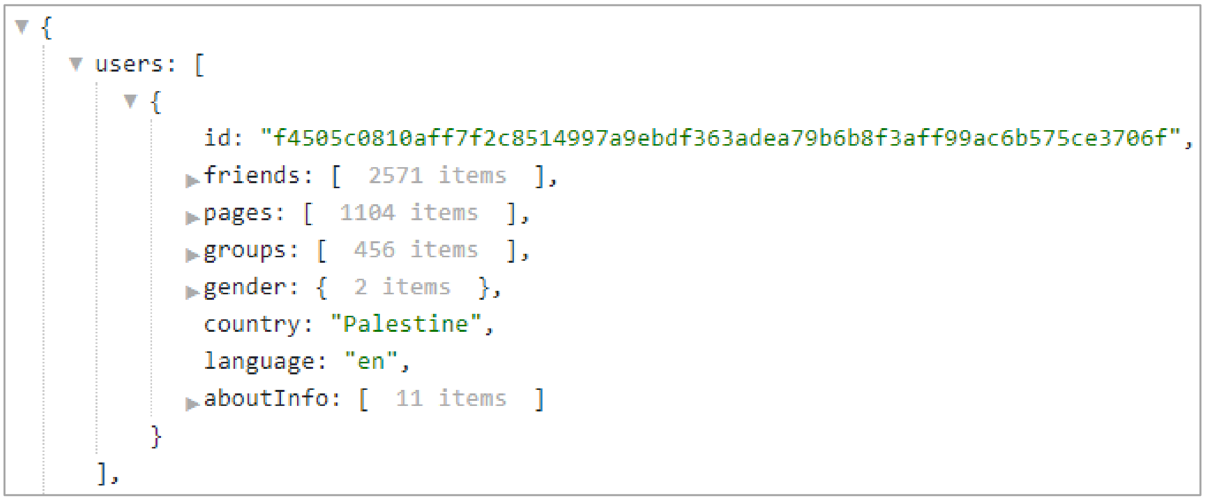
If there are unclear fields, click here.

```
▼ {
    ▶ users: [ 1 item ],
    ▶ actions: [ 13 items ],
    ▶ apps: [ 1 item ],
    ▶ devices: [ 1 item ],
    ▶ activityLog: [ 84 items ]
  }
```

Do you want to send it to us?

Figure 10: Collected data – General view on the collected data from the user Facebook account

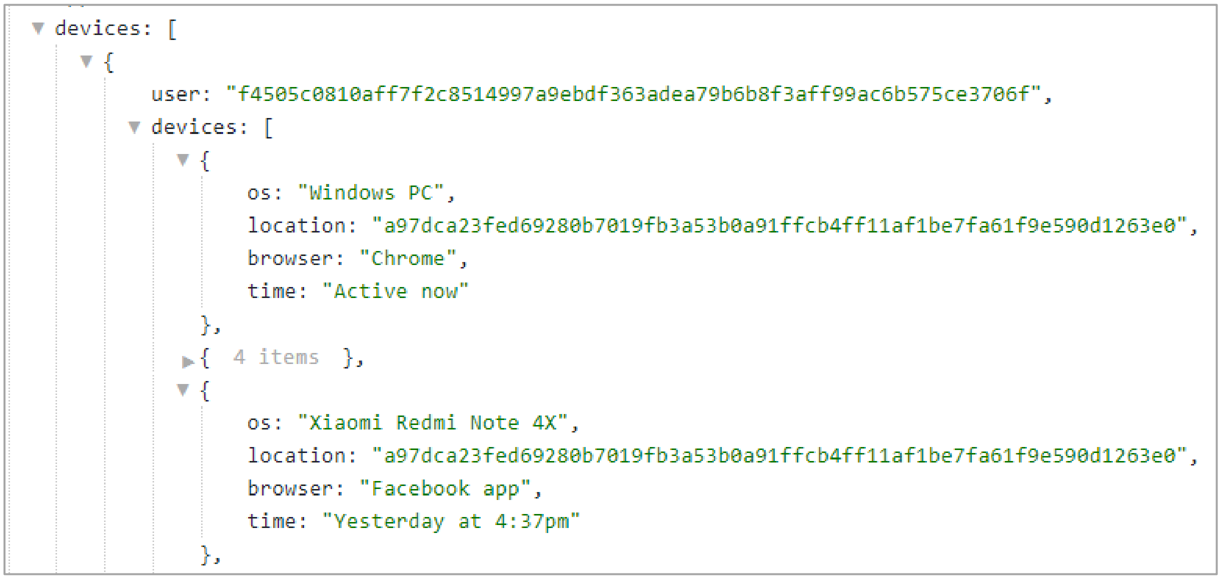
Different types of data will be collected about the user; the collected data could be viewed in JSON Viewer page. In figure (11), the user can view the collected data about his account like the hashed ID for the user, friend's IDs, pages, groups, gender, Date of Birth, country and language.



```
▼ {
    ▼ users: [
        ▼ {
              id: "f4505c0810aff7f2c8514997a9ebdf363adea79b6b8f3aff99ac6b575ce3706f",
            ▶ friends: [  2571 items  ],
            ▶ pages: [  1104 items  ],
            ▶ groups: [  456 items  ],
            ▶ gender: {  2 items  },
              country: "Palestine",
              language: "en",
            ▶ aboutInfo: [  11 items  ]
          }
      ],
```

Figure 11: Some collected data about the user account (friends, pages …etc.)

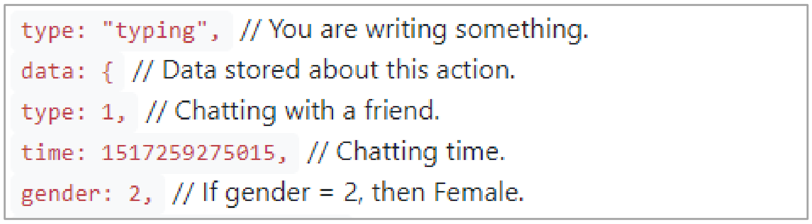
Another collected data types the user can check, like the logged-in devices as in figure (12) that the user was used it to log-in his account.



```
 ▼ devices: [
     ▼ {
           user: "f4505c0810aff7f2c8514997a9ebdf363adea79b6b8f3aff99ac6b575ce3706f",
         ▼ devices: [
             ▼ {
                   os: "Windows PC",
                   location: "a97dca23fed69280b7019fb3a53b0a91ffcb4ff11af1be7fa61f9e590d1263e0",
                   browser: "Chrome",
                   time: "Active now"
               },
             ▶ {  4 items  },
             ▼ {
                   os: "Xiaomi Redmi Note 4X",
                   location: "a97dca23fed69280b7019fb3a53b0a91ffcb4ff11af1be7fa61f9e590d1263e0",
                   browser: "Facebook app",
                   time: "Yesterday at 4:37pm"
               },
```

Figure 12: Some collected data about the user account (logged-in devices)

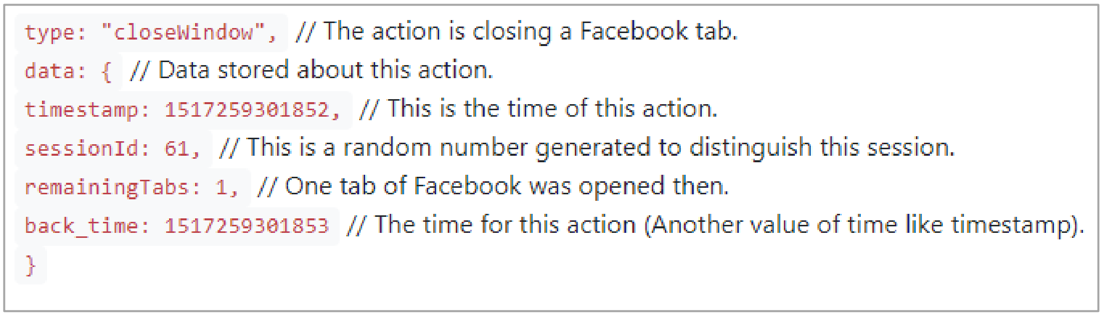
If some JSON fields are unclear for the user, he can read the explanation of these fields below the sending data options. In figure (13) below, this is an example of fields explanation, the user was chatting with a friend in his Facebook account, she was female as shown below. The chatting time representing the number of milliseconds since midnight January 1, 1970 as in many programming languages too.



```
type: "typing",  // You are writing something.
data: {  // Data stored about this action.
type: 1,  // Chatting with a friend.
time: 1517259275015,  // Chatting time.
gender: 2,  // If gender = 2, then Female.
```

Figure 13: Fields explanation example - Chatting with a friend action

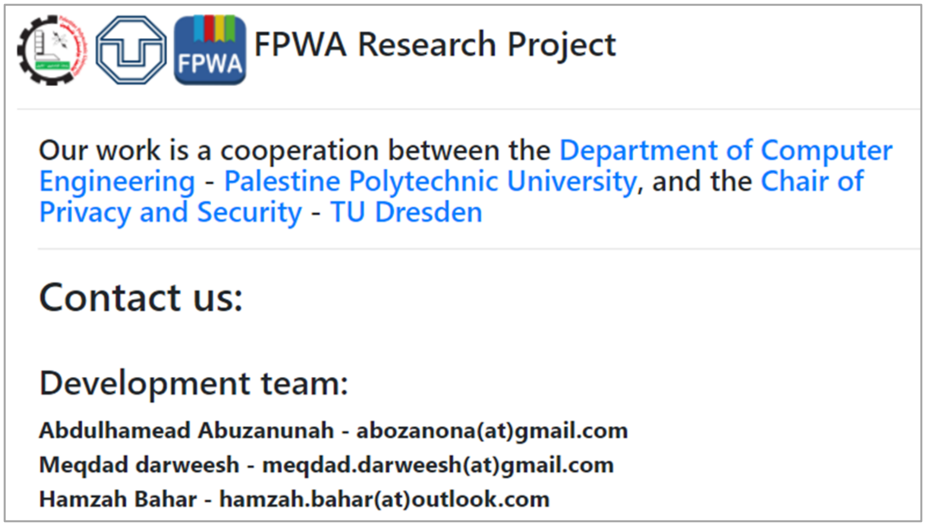
Another fields explanation example, it's when the user closing a Facebook tab, see figure (14). Plugin collects the time of action and the session ID which is generated randomly to distinguish the current session, and it collects how many Facebook tabs remains after this closed tab.



```
type: "closeWindow",  // The action is closing a Facebook tab.
data: {  // Data stored about this action.
timestamp: 1517259301852,  // This is the time of this action.
sessionId: 61,  // This is a random number generated to distinguish this session.
remainingTabs: 1,  // One tab of Facebook was opened then.
back_time: 1517259301853  // The time for this action (Another value of time like timestamp).
}
```

Figure 14: Fields explanation example - Closing a Facebook tab action

- **About us**: In this interface as figure (15), the user can read information about the universities and the participated team in this work. Also, all of these information is mentioned in the website.



FPWA Research Project

Our work is a cooperation between the Department of Computer Engineering - Palestine Polytechnic University, and the Chair of Privacy and Security - TU Dresden

Contact us:

Development team:

Abdulhamead Abuzanunah - abozanona(at)gmail.com
Meqdad darweesh - meqdad.darweesh(at)gmail.com
Hamzah Bahar - hamzah.bahar(at)outlook.com

Figure 15: About us interface – The development team and their supervisors

3.3: Application

Browser plugin was the only tool used to collect data in the previous two old studies. In our new work we developed in addition to the plugin a mobile application Facebook Activity Log Collector (FALC) to collect the data from smartphones users. We choose to build Android application rather than iOS or Windows Phone applications, because Android is more popular than other mentioned platforms and because it is open source platform, so it is relatively easier to build Android application.

3.3.1: Design options

We were searching on how to build Android application serving our purpose, many options are available that might help us reach our goal. Facebook SDK for Android and TCP Sniffer are of these available options.

Our first possibility to build application capable of collecting data was Facebook SDK for Android, which is a Java library for Android provided by Facebook that let you read and write to the Facebook social graph using Graph API. By using Graph API, one can make requests and get responses in JSON format, after asking for permissions, which are required by Facebook to enable you collect the data provided by these permissions. In order to obtain these permissions, you must provide Facebook with evidence that prove you are not going to use these permissions in a wrong way. There is a set of permissions that we have checked to make sure they meet our needs. Indeed, some of these permissions meet what we need on one hand, there are other requirements we did not find permissions for it on the other hand. So, it was clear to us at that time that we cannot rely on Facebook SDK for Android to build the app, but at the same time we can use the provided permissions that serve our requirements, as a result, we started working on the other path (TCPSniffer).

After Facebook SDK for Android, we turned to the idea of TCP Sniffer. In this the application try to capture the packets going out from the Facebook application to Facebook server and vice versa. Packet Sniffers works by intercepting network traffic that packet sniffing software has access to on its host device. Once the raw packet data is captured, it will be analyzed by packet sniffing software.

We tried to sniff data by building Android app using Virtual Personal Network (VPN) as in figure (16) below, but when we run the app all network traffic between Facebook app and its server has stopped, in other words, actions done via Facebook app does not reflect, and the app stops receiving from the server. After searching about that problem, we found that we need certificate in order for everything to work fine but installing the certificate does not work since other functions of the mobile phone have been corrupted.

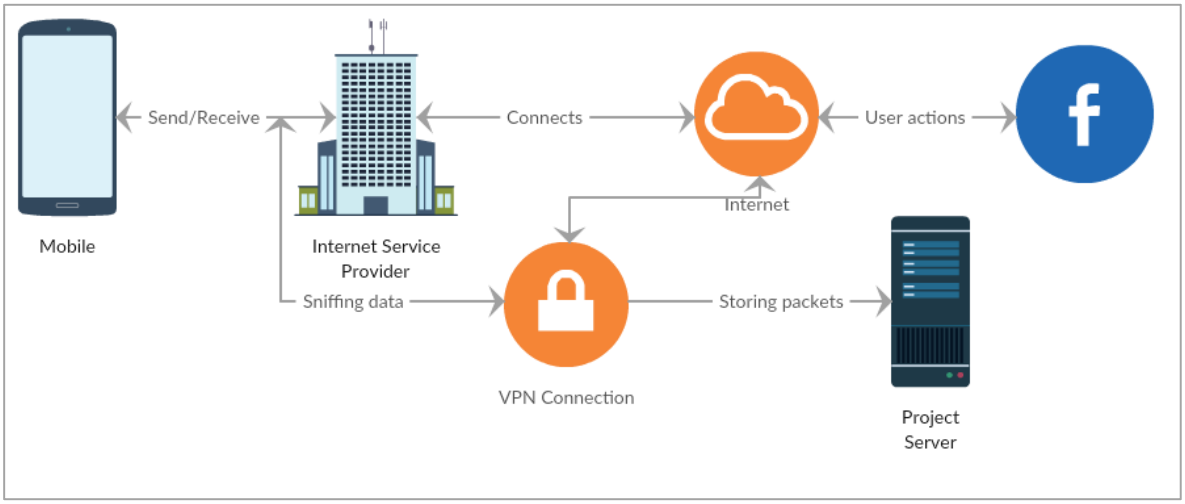


Figure 16: First implemented approach of Android Application

We kept searching until we find Android code that utilizes TCP Sniffer and works fine, but it only works with rooted mobile devices. We reject this solution, because it does not make sense to make all users root their mobile devices.

Another approach is to use NDK to build our own tunnel, but the construction of that tunnel was difficult, so we try to look for open source applications that utilize that feature instead of building it. We found that application but it stores the packets in PCAP file. Analyzing this file was not easy since Facebook app use FP/Zero protocol, this means that data is encrypted and the routs cannot be read.

Table 2: Available design options for data collection (Smart Devices)

Design options	Number of users	Accessing required data	Sharing and publishing	Prerequisites / constraints
Facebook API (Android application)	Limited	Not achievable	Easy	Lots of constraints
TCP Sniffer	Unlimited	Achievable	Easy	Lots of constraints
Hybrid (FALC)	Unlimited	Acceptable	Easy	Limited

3.3.2: Our design

In Android application we couldn't continue with our previous approach, we tried very hard to decrypt the generated keys from FP/Zero protocol, but eventually we couldn't do it. Here we decided to turn into totally different approach which is to develop our code to collect data from the Facebook application on the smartphone. After all difficulties we face in building Android application either by using Facebook SDK for Android or TCP Sniffer. We decide to make use of what we did in the plugin and build a hybrid application which combines native development with web technology.

New Android application is developed using Java (Android Studio) to collect the data stored in Activity log of the user account on Facebook and it will send the collected data to our server according to specific schedule that is previously configured, data sending process will only complete when the acceptance is being given by the user, see figure (17).

The user needs to login into his account when he opens our application, then the user can continue his traditional activities on the official Facebook application, and he will not feel that something weird or strange is happening, our application will work in the background.

Our new approach of building the application depends on the idea that the application work in a wrapper or container which means that we run the application into the browser, this enable us to use functionalities offered by web-developing languages like JavaScript. By this we could use the same approach we used in our plugin for analysis process. Instead of trying to pick the Facebook packets while they are travelling from Facebook app to the server or vice versa. Also, security and anonymization are considered as in plugin, we are hashing each item that could leads to the user, i.e. the user is anonymized in our database and we know the type of actions just, not the user identity.

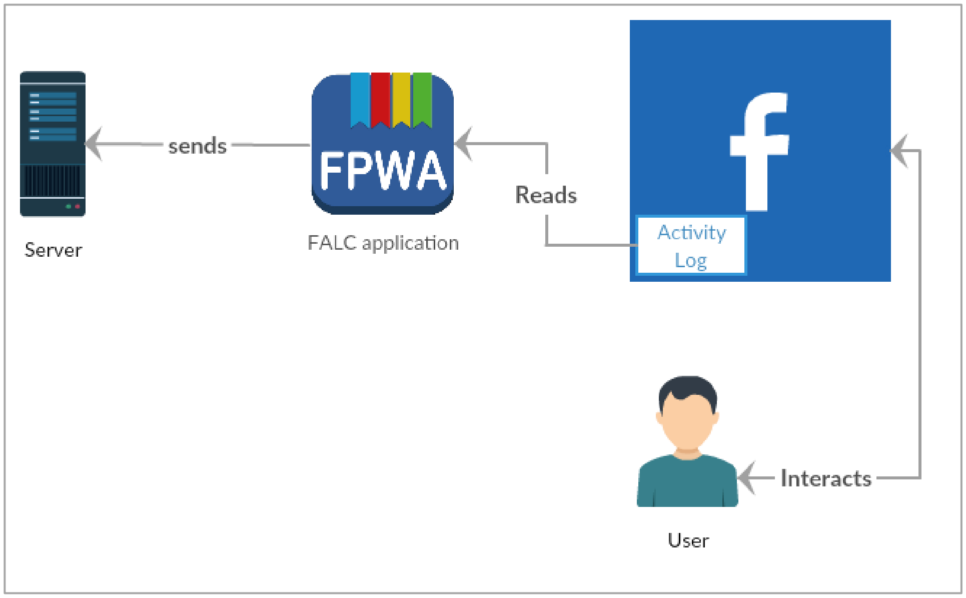


Figure 17: FALC application - Block diagram for work principle

3.3.3: Implementation

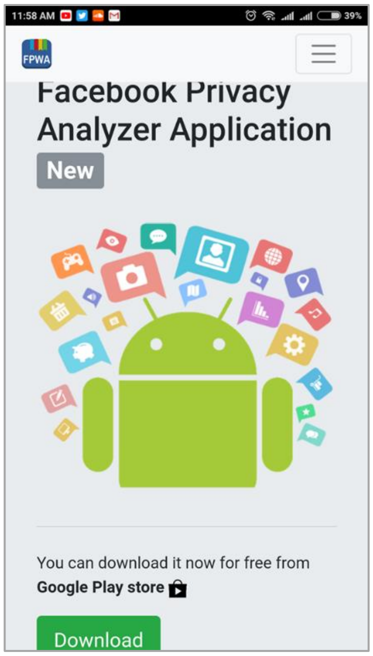
The application launches a WebView to load Facebook page in it, Facebook links are the only links to be loaded into that WebView. Two Facebook pages are to be loaded into that WebView. First page is the login page, this page appears when the user first starts the application, to ask him to log into his Facebook account, after log in a new cookie will be stored into the WebView called c_user, this cookie is used to know if the user is logged in or not, if the user is logged in, the app move him to the second page, which has the history of what we collected about him. User ID can be found in the c_user cookie as well; this ID is to be stored in the storage of the application.

The application opens the activity log of the Facebook user account, the activity log consists of group of filters, for example there is filters for friends, likes, pages, and many other. Some filters may exist in one Facebook account and not appear in another one, examples of these filters include: filters of the apps used by Facebook user, such that each app has its own filter. These filters appear according to what apps the user use.

Data is collected from these filters every day; such that every half-hour data of one filter is to be collected by opening the link of that filter associated with User ID. The collected data is then stored into a local database and sent immediately to our server. The time of the action and the content in which the action happened to are stored also, for example, if the user liked something, the time of the like is stored, and the content that has been liked is also stored along with hash value of the user ID. The application collects the actions that happened three days earlier, which means that the collected data of one day was generated three days before. The privacy of the collected data is preserved by hashing any number appears and removing any user name.

3.3.4: Usage

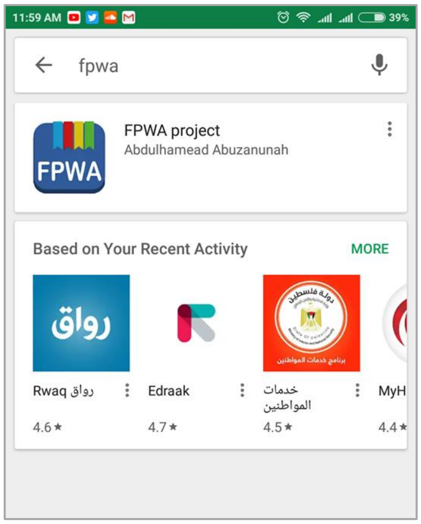
Using and interacting with the application is very easy and friendly, the user can use the mentioned link on the project website as in figure (18) or he can search for the application on Google Play Store as in figure (19).



Facebook Privacy Analyzer Application

You can download it now for free from **Google Play store**

Figure 18: Download link for application on project website



fpwa

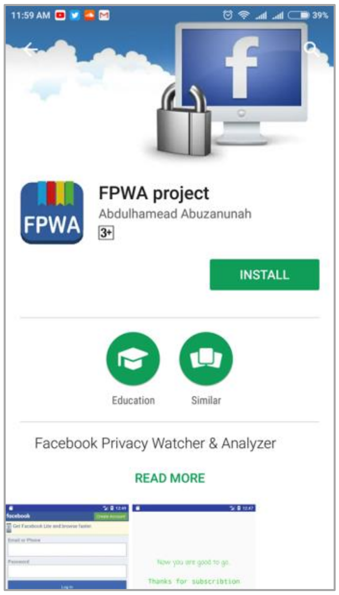
FPWA project
Abdulhamead Abuzanunah

Based on Your Recent Activity

Rwaq رواق	Edraak	خدمات المواطنين	MyH
4.6★	4.7★	4.5★	4.4★

Figure 19: Download link for searching on application in Google play store

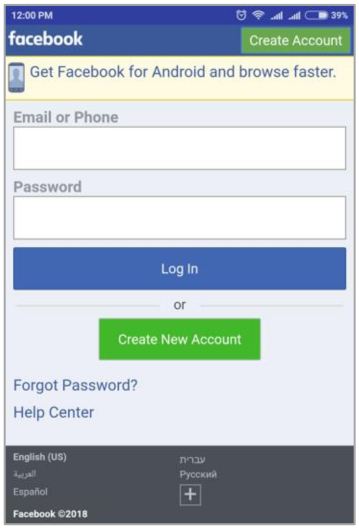
After finding the download screen for the application as in figure (20), the user needs to click on "Install" to download the application.



Facebook Privacy Watcher & Analyzer

Figure 20: Installation screen on Google Play Store

Now, the user can open the application and log into his account in Facebook through our application as in figure (21) below.



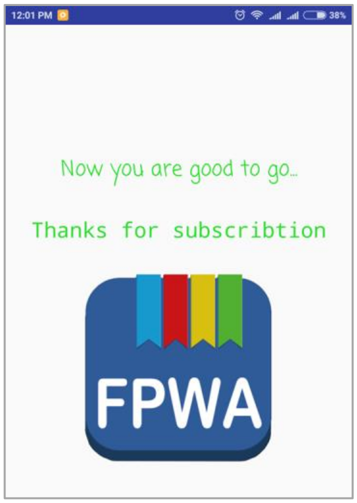
Get Facebook for Android and browse faster.

Email or Phone

Password

Figure 21: Log in screen for user account in application

After finish logging in process, the successful screen will appear for the user as in figure (22) below.

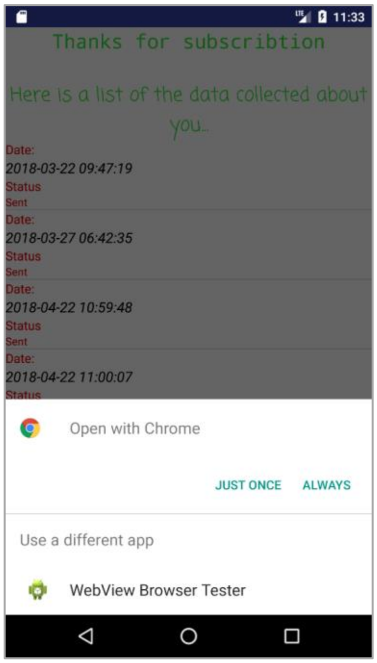


Now you are good to go..

Thanks for subscribtion

Figure 22: Successful log in screen in the application

Now, the user does not need to do another action, he can continue using the official Facebook application on his Android device. To view the collected data about his account, he can check that as in figure (23) below.



Thanks for subscribtion

Here is a list of the data collected about you..

Date:
2018-03-22 09:47:19
Status
Sent
Date:
2018-03-27 06:42:35
Status
Sent
Date:
2018-04-22 10:59:48
Status
Sent
Date:
2018-04-22 11:00:07
Status

Open with Chrome

Use a different app

WebView Browser Tester

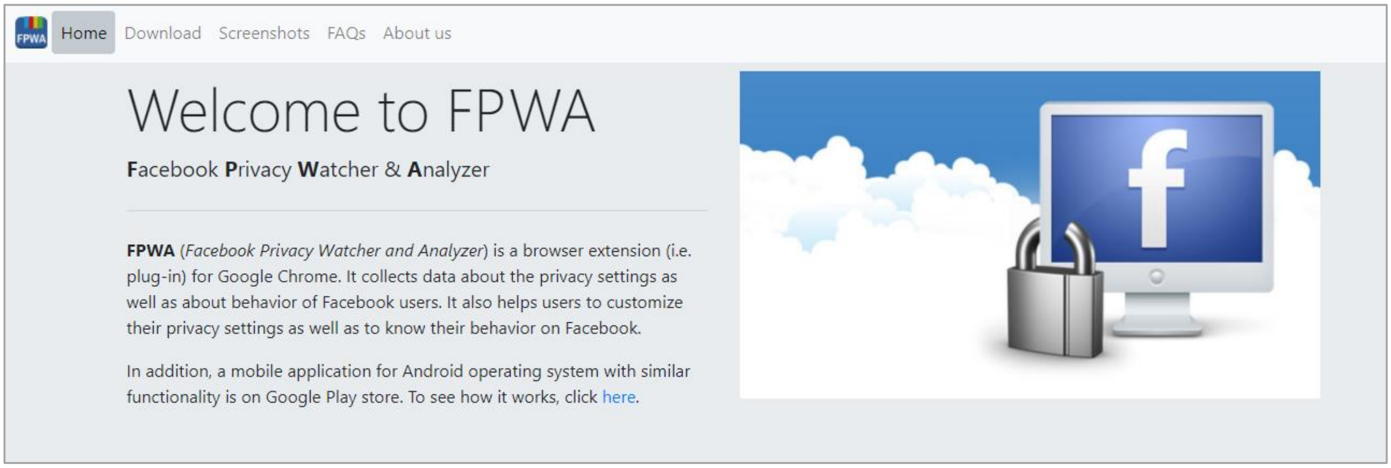
Figure 23: Viewing the collected data in application

3.4: Project Website:

The project website (https://fba.ppu.edu) was developed using Bootstrap 4, it was divided to different sections to be friendly for the visitors. It contains the required explanation and links:

- **Home section:**

This is the first section in website, at the top of main page. Small two paragraphs explain the main purposes and the definition of the software tools in the project, as in figure (24).



Welcome to FPWA

Facebook **P**rivacy **W**atcher & **A**nalyzer

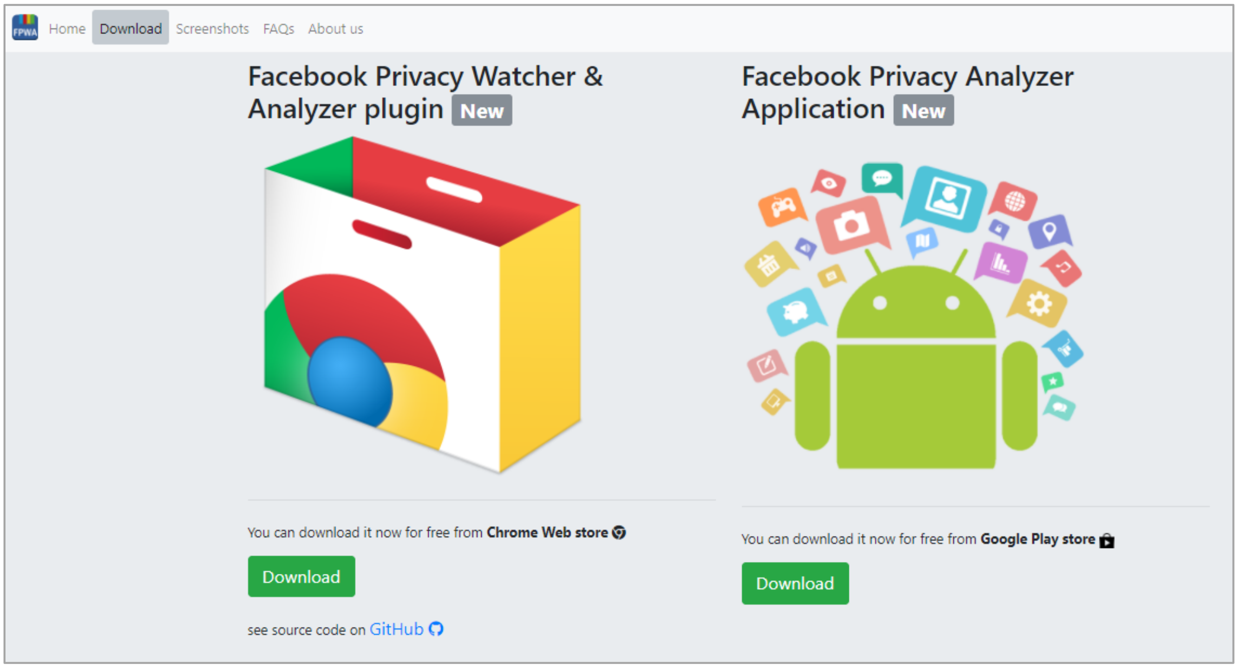
FPWA (*Facebook Privacy Watcher and Analyzer*) is a browser extension (i.e. plug-in) for Google Chrome. It collects data about the privacy settings as well as about behavior of Facebook users. It also helps users to customize their privacy settings as well as to know their behavior on Facebook.

In addition, a mobile application for Android operating system with similar functionality is on Google Play store. To see how it works, click here.

Figure 24: Website - Home section in the project website

- **Download section:**

In this section, the downloading links attached, the Chrome plugin and Android application, see figure (25).



Facebook Privacy Watcher & Analyzer plugin New

You can download it now for free from **Chrome Web store**

see source code on GitHub

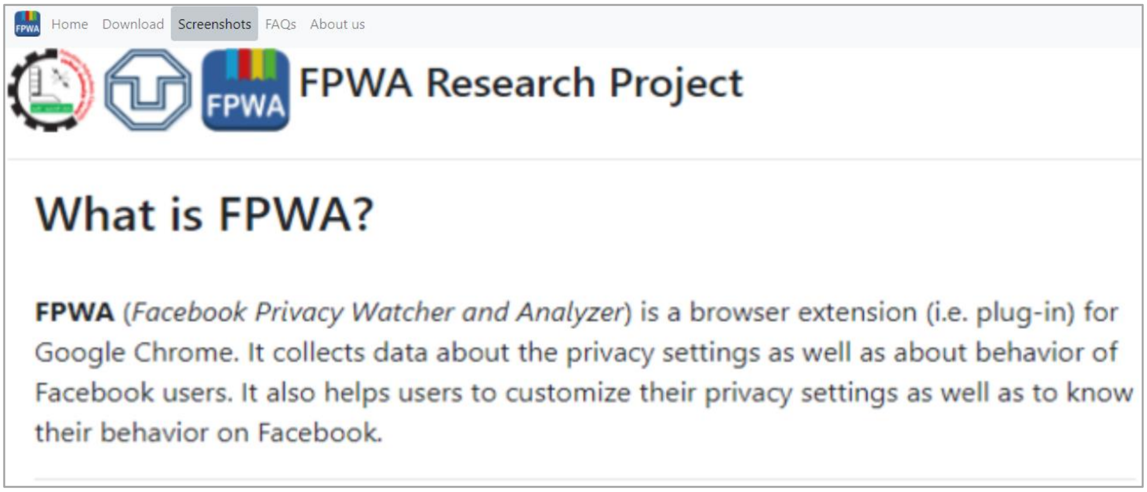
Facebook Privacy Analyzer Application New

You can download it now for free from **Google Play store**

Figure 25: Download section in the project website

- **<u>Screenshots section</u>**

This section contains some screenshots for the plugin, the user can swap between them to explore the main interfaces before downloading, see figure (26).



FPWA Research Project

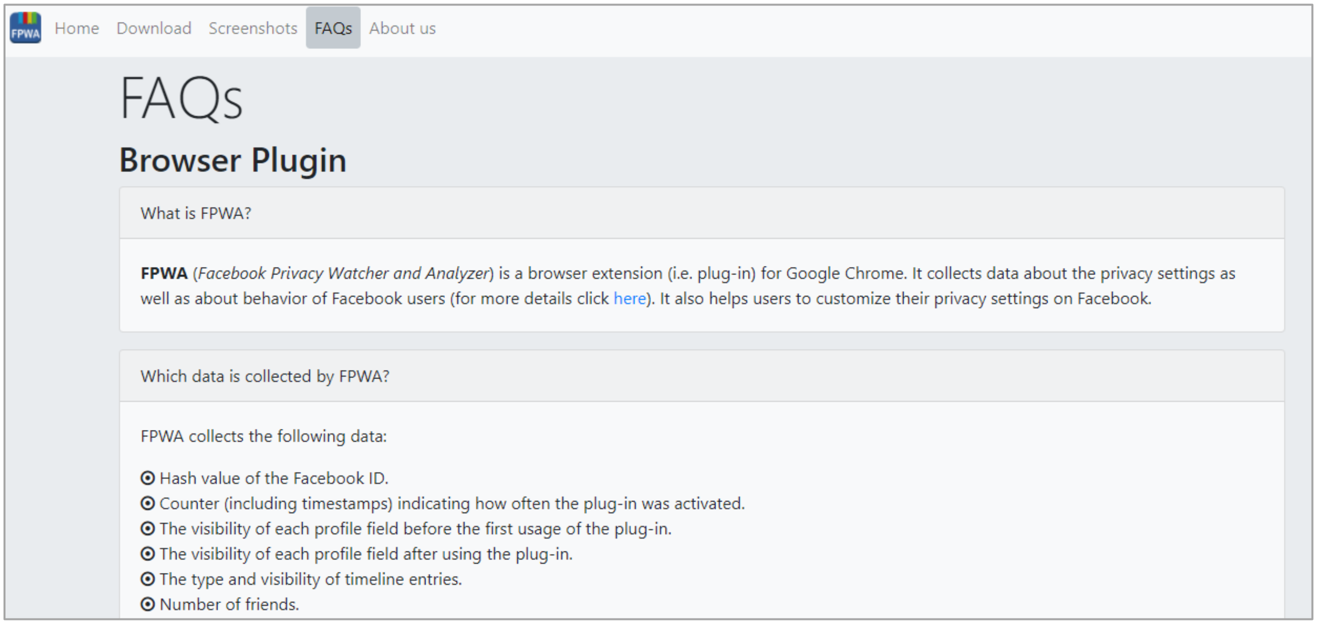
What is FPWA?

FPWA (*Facebook Privacy Watcher and Analyzer*) is a browser extension (i.e. plug-in) for Google Chrome. It collects data about the privacy settings as well as about behavior of Facebook users. It also helps users to customize their privacy settings as well as to know their behavior on Facebook.

Figure 26: Screenshots section in the project website

- **<u>FAQs section:</u>**

It contains the frequently asked questions, like what is FPWA and what the plugin and application collecting from the participants, see figure (27), all questions are attached in Appendix A.



FAQs

Browser Plugin

What is FPWA?

FPWA (*Facebook Privacy Watcher and Analyzer*) is a browser extension (i.e. plug-in) for Google Chrome. It collects data about the privacy settings as well as about behavior of Facebook users (for more details click here). It also helps users to customize their privacy settings on Facebook.

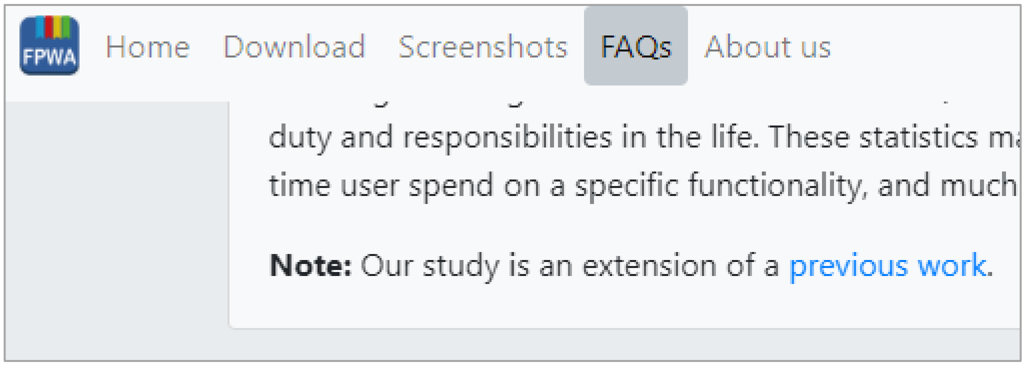
Which data is collected by FPWA?

FPWA collects the following data:

⦿ Hash value of the Facebook ID.
⦿ Counter (including timestamps) indicating how often the plug-in was activated.
⦿ The visibility of each profile field before the first usage of the plug-in.
⦿ The visibility of each profile field after using the plug-in.
⦿ The type and visibility of timeline entries.
⦿ Number of friends.

Figure 27: FAQs section in the project website

In FAQs, we mentioned a link for a new page that summarize the two previous studies that we want to update, when the user clicks on the link as in figure (28) below, he will see the page as in figure (29).

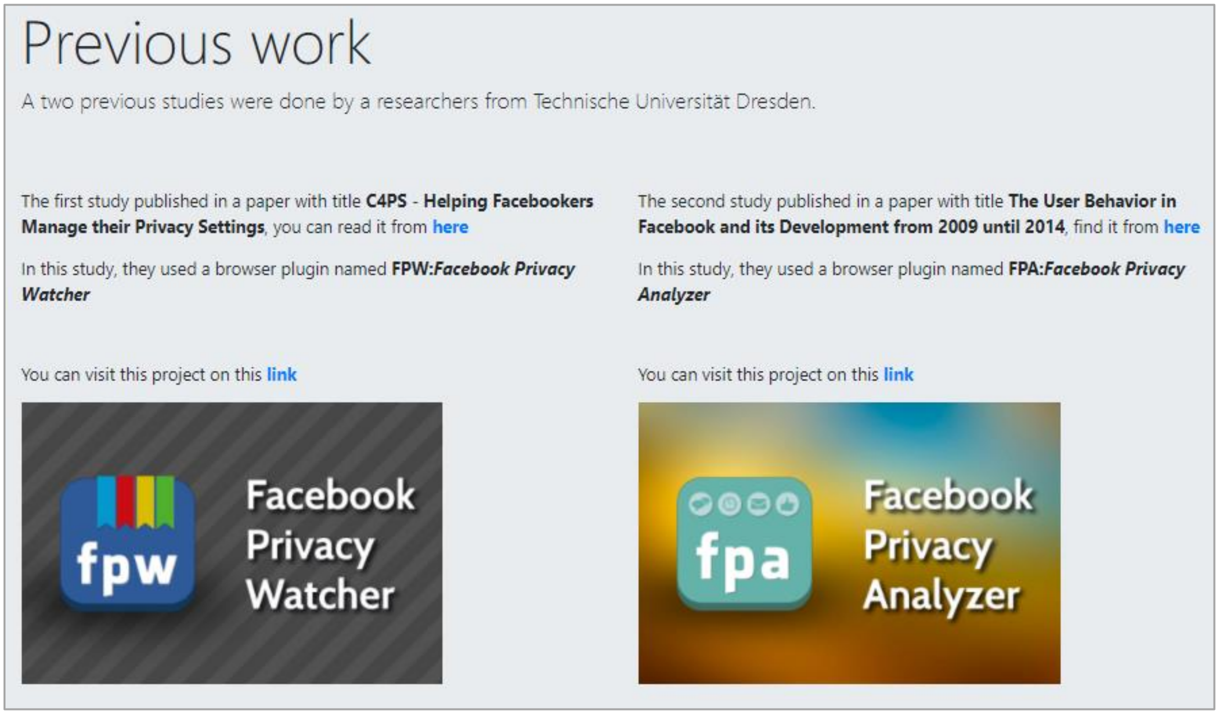


duty and responsibilities in the life. These statistics m
time user spend on a specific functionality, and much

Note: Our study is an extension of a previous work.

Figure 28: Previous work link in FAQs

The previous work page contains the required links for the data collection tools websites and the published papers with their links to read.



Previous work

A two previous studies were done by a researchers from Technische Universität Dresden.

The first study published in a paper with title **C4PS - Helping Facebookers Manage their Privacy Settings**, you can read it from **here**

In this study, they used a browser plugin named **FPW:*Facebook Privacy Watcher***

You can visit this project on this **link**

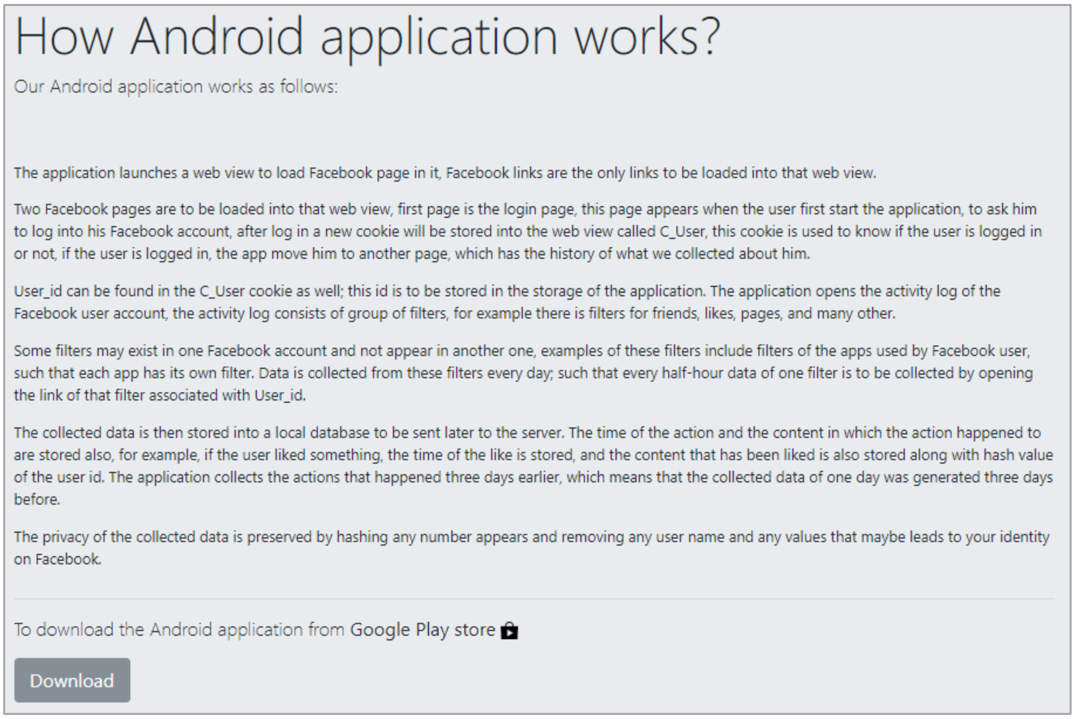
The second study published in a paper with title **The User Behavior in Facebook and its Development from 2009 until 2014**, find it from **here**

In this study, they used a browser plugin named **FPA:*Facebook Privacy Analyzer***

You can visit this project on this **link**

Figure 29: Previous work screen on the project website

For Android application, we mentioned a link in FAQ that leads the website visitor to a new page contains a detailed explanation for the application work principle and the link of application on Google Play Store as in figure (30).



How Android application works?

Our Android application works as follows:

The application launches a web view to load Facebook page in it, Facebook links are the only links to be loaded into that web view.

Two Facebook pages are to be loaded into that web view, first page is the login page, this page appears when the user first start the application, to ask him to log into his Facebook account, after log in a new cookie will be stored into the web view called C_User, this cookie is used to know if the user is logged in or not, if the user is logged in, the app move him to another page, which has the history of what we collected about him.

User_id can be found in the C_User cookie as well; this id is to be stored in the storage of the application. The application opens the activity log of the Facebook user account, the activity log consists of group of filters, for example there is filters for friends, likes, pages, and many other.

Some filters may exist in one Facebook account and not appear in another one, examples of these filters include filters of the apps used by Facebook user, such that each app has its own filter. Data is collected from these filters every day; such that every half-hour data of one filter is to be collected by opening the link of that filter associated with User_id.

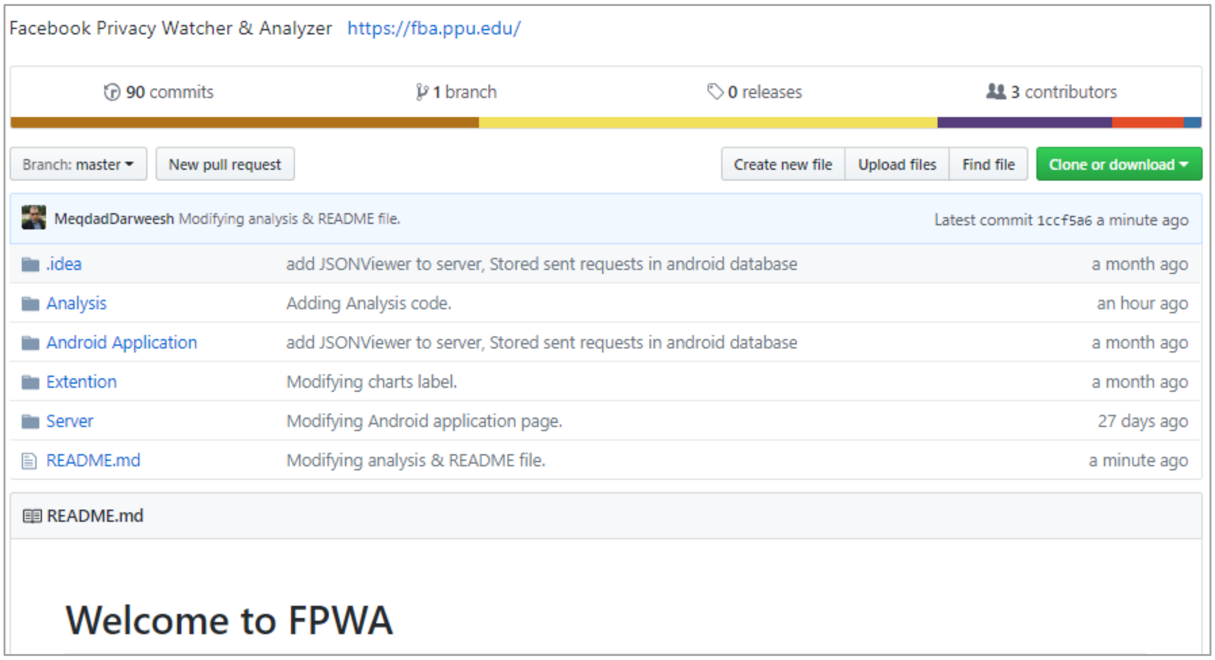
The collected data is then stored into a local database to be sent later to the server. The time of the action and the content in which the action happened to are stored also, for example, if the user liked something, the time of the like is stored, and the content that has been liked is also stored along with hash value of the user id. The application collects the actions that happened three days earlier, which means that the collected data of one day was generated three days before.

The privacy of the collected data is preserved by hashing any number appears and removing any user name and any values that maybe leads to your identity on Facebook.

To download the Android application from Google Play store

Figure 30: Application work principle explanation page on website

For developers and those who needs to explore the source code of project, we mention a link for the project on GitHub in the website, see figure (31). Readme file on GitHub is used to clarify the main idea and the details about the involved team. The GitHub repository contains the classes of JS for plugin, the Android studio files for the application, the website files and the analysis code.



Facebook Privacy Watcher & Analyzer https://fba.ppu.edu/

MeqdadDarweesh Modifying analysis & README file. Latest commit 1ccf5a6 a minute ago

.idea	add JSONViewer to server, Stored sent requests in android database	a month ago
Analysis	Adding Analysis code.	an hour ago
Android Application	add JSONViewer to server, Stored sent requests in android database	a month ago
Extention	Modifying charts label.	a month ago
Server	Modifying Android application page.	27 days ago
README.md	Modifying analysis & README file.	a minute ago

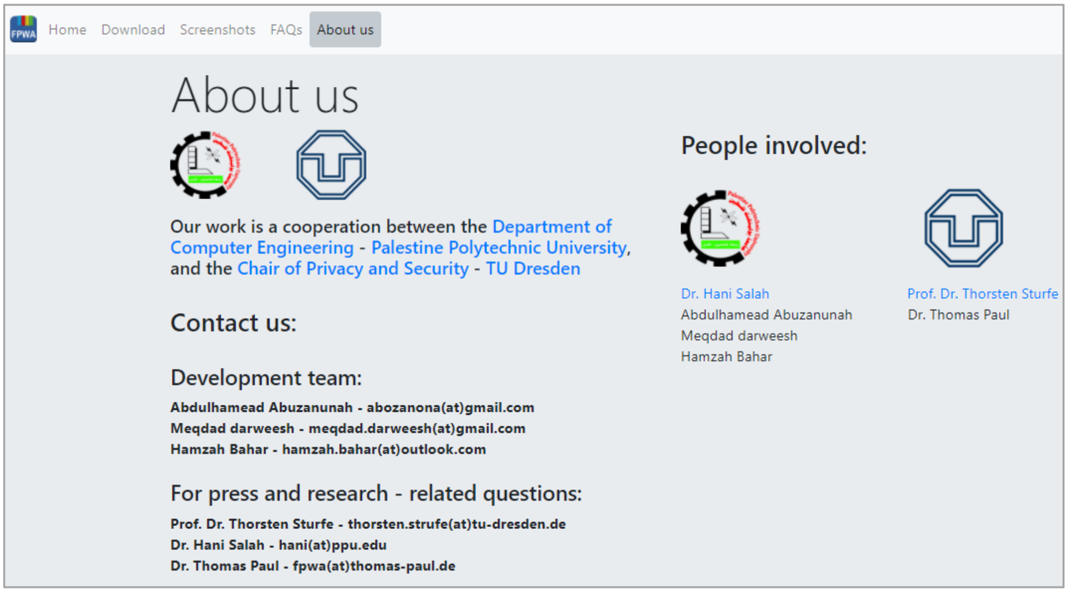
README.md

Welcome to FPWA

Figure 31: The project repository on GitHub

About us section:

Last section contains the data about the project developers, supervisors and their universities with the required links, see figure (32).



About us

Our work is a cooperation between the Department of Computer Engineering - Palestine Polytechnic University, and the Chair of Privacy and Security - TU Dresden

Contact us:

Development team:

Abdulhamead Abuzanunah - abozanona(at)gmail.com
Meqdad darweesh - meqdad.darweesh(at)gmail.com
Hamzah Bahar - hamzah.bahar(at)outlook.com

For press and research - related questions:

Prof. Dr. Thorsten Sturfe - thorsten.strufe(at)tu-dresden.de
Dr. Hani Salah - hani(at)ppu.edu
Dr. Thomas Paul - fpwa(at)thomas-paul.de

People involved:

Dr. Hani Salah Abdulhamead Abuzanunah Meqdad darweesh Hamzah Bahar	Prof. Dr. Thorsten Sturfe Dr. Thomas Paul

Figure 32: About us section in the project website

Ch4: Preliminary Results

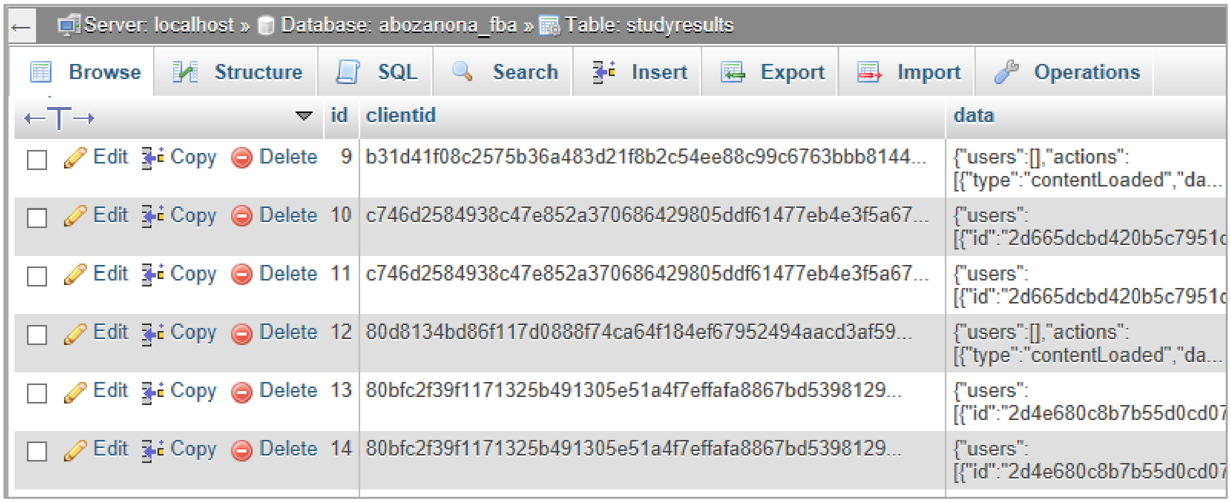
In this chapter, the analysis of the collected data set was introduced. The result was not comprehensive, but the analysis program can analyze any sample amount. The analysis program is developed using Python with some libraries that make the analysis easier. Also the analysis code is uploaded to the project repository on the GitHub website.

4.1: Data Sets

4.1.1: Plugin

The plugin's number of downloads are small, and the collected data is not comprehensive. Our plugin users are 19 users with different data submissions, some of the users choose to send data "Always" as in the sending data options, others choose to send data "This time only" and some users just download it and didn't send any data.

The sent data stored as JSON format as in the previous study and it is easy to deal with in analysis process. In the analysis, we use Python and the related libraries like Pandas, NumPy, MatPlotLib and other libraries. In the previous work, R language was used. We prefer Python than R because we have some previous knowledge and we used it before, also Python is rich and have a lot of third-party software libraries for data analysis in different fields. As in figure (33) below, you can see the collected data on our server, each user has a different "clientid" when he submits the collected data, if the user submits many times, the "clientid" will remain the same, but the collected data will be changed with different values like "session id" in JSON format.


←
Server: localhost » Database: abozanona_fba » Table: studyresults
Browse
Structure
SQL
Search
Insert
Export
Import
Operations
←T→

←T→	id	clientid	data
□ Edit Copy Delete	9	b31d41f08c2575b36a483d21f8b2c54ee88c99c6763bbb8144...	{"users":[],"actions": [{"type":"contentLoaded","da...
□ Edit Copy Delete	10	c746d2584938c47e852a370686429805ddf61477eb4e3f5a67...	{"users": [{"id":"2d665dcbd420b5c7951c
□ Edit Copy Delete	11	c746d2584938c47e852a370686429805ddf61477eb4e3f5a67...	{"users": [{"id":"2d665dcbd420b5c7951c
□ Edit Copy Delete	12	80d8134bd86f117d0888f74ca64f184ef67952494aacd3af59...	{"users":[],"actions": [{"type":"contentLoaded","da...
□ Edit Copy Delete	13	80bfc2f39f1171325b491305e51a4f7effafa8867bd5398129...	{"users": [{"id":"2d4e680c8b7b55d0cd07
□ Edit Copy Delete	14	80bfc2f39f1171325b491305e51a4f7effafa8867bd5398129...	{"users": [{"id":"2d4e680c8b7b55d0cd07

Figure 33: A view for collected data on the server

4.2: Data Analysis

4.2.1: Plugin

The server database received data from 16 users (12 Males / 4 Female) with different data submissions (around 54). Figure (34) illustrate the percentage of male to female participants. English was the language of most users' accounts, in figure (35) we can see that the number of users who choose English as an account language surpassed those who choose Arabic. One can also notice from figure (35) that English and Arabic were the only languages used by the users, but this result was consistent with the society our users came from.

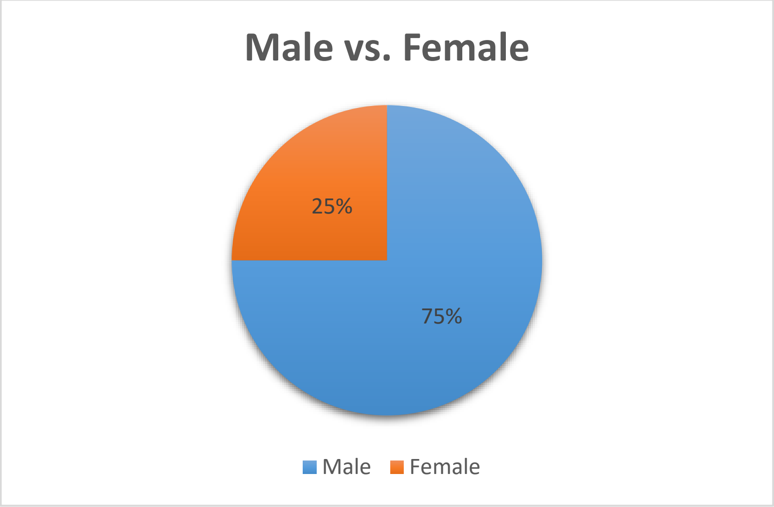


Figure 34: Percentage of users' gender - Males vs. Females

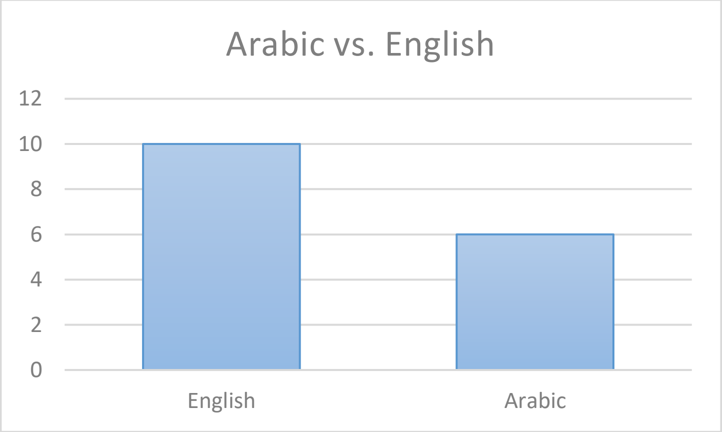


Figure 35: Languages of Facebook account

Our results also showed that the percentage of males in the accounts related to male users is far greater than the percentage of females as in figure (36). Similarly, the percentage of females in the accounts related to female users is higher than the males as shown in figure (37).

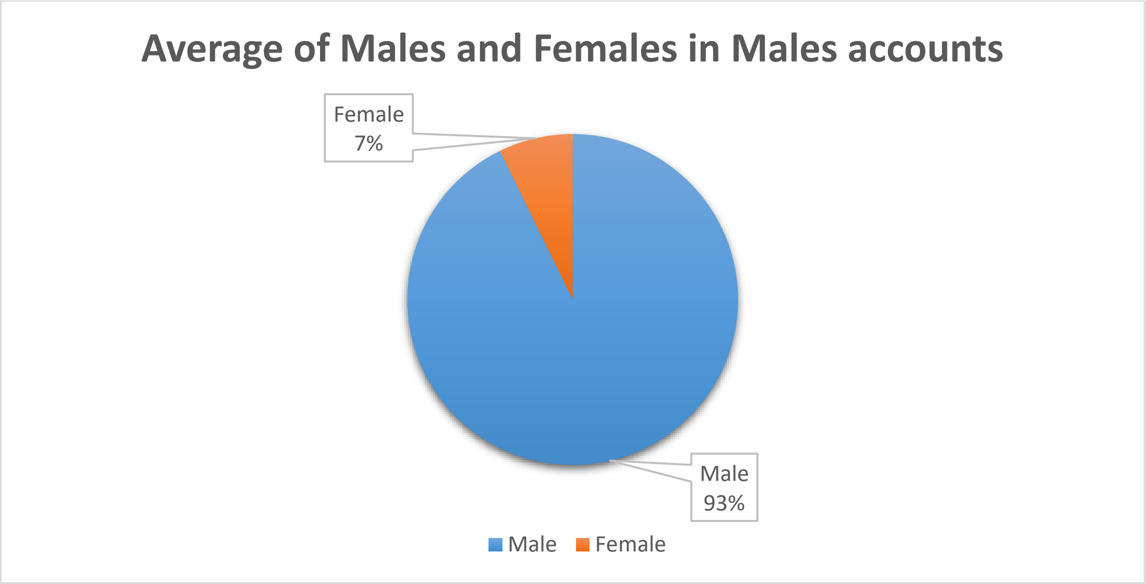


Figure 36: Average of Males and Females in Males accounts

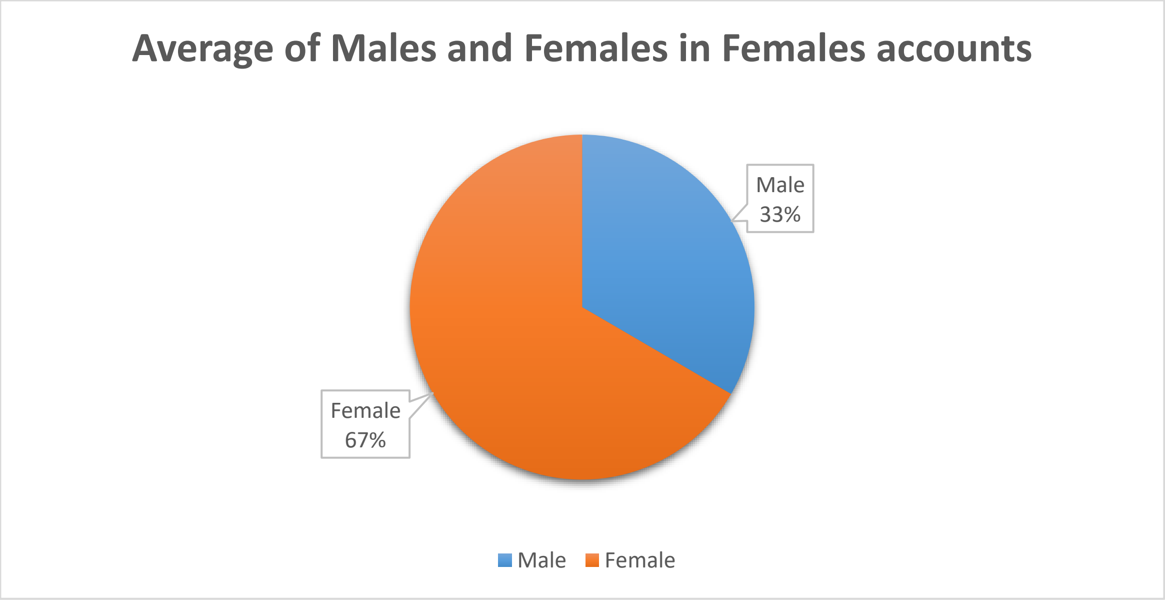


Figure 37: Percentage of Males and Females in Females accounts

As figure (38) illustrates, the public groups are the most of joined groups, while the second is closed groups and the last one is secret groups. Public groups mean that the group is searchable and the user can view the posts and members of the group without joining the group. Closed group mean that the group is searchable but the user can not view the posts, just members and the admins accounts. Secret group is unsearchable and the user can't know anything about it, like the members or admins or anything like that. The only way to join the secret group is by adding the use to the group from the admin of the group or a member, it depends on the group's configurations.

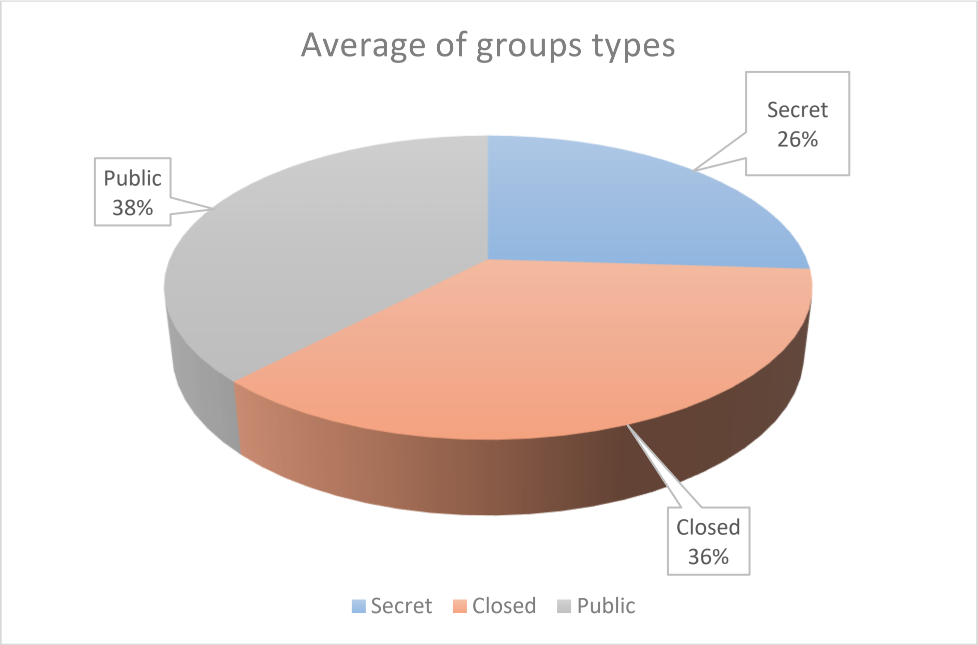


Figure 38: Average of groups types

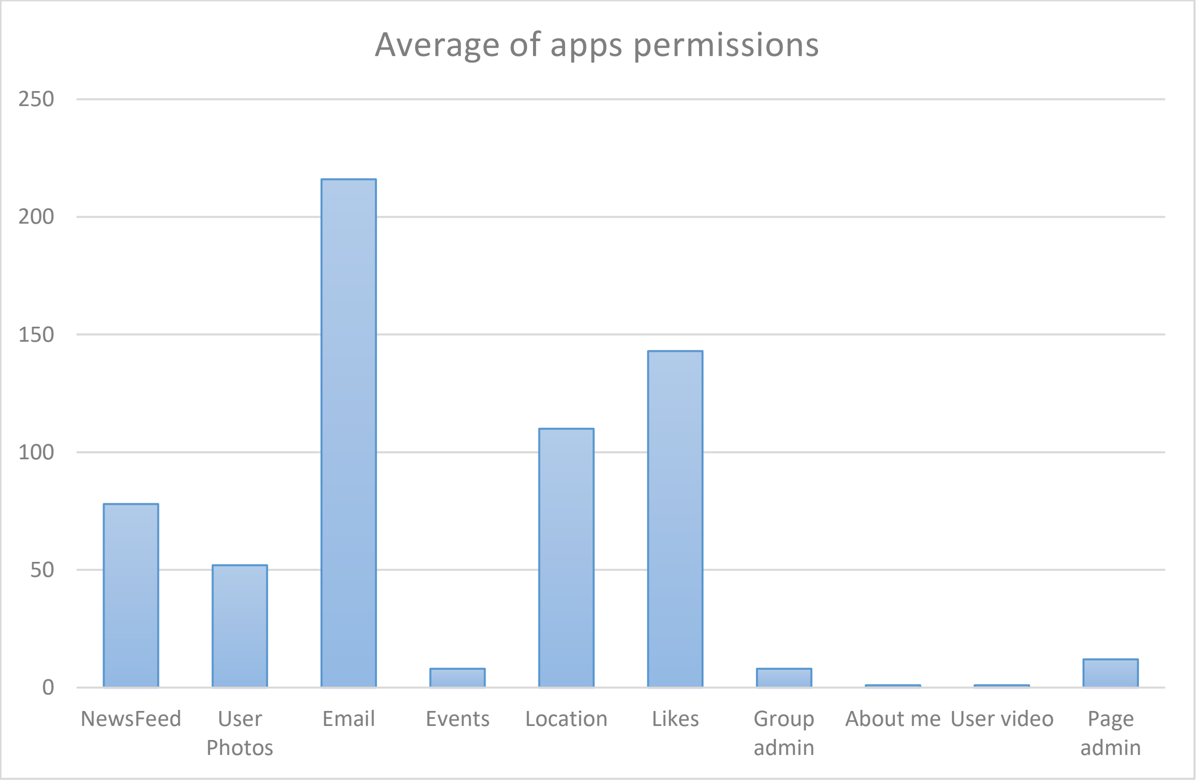


Figure 39: Average of apps permissions

Figure (39) shows the average permissions that Facebook apps ask for. One can notice from figure (39) that the highest percentage of permissions asked by the apps are for Email and likes comes second.

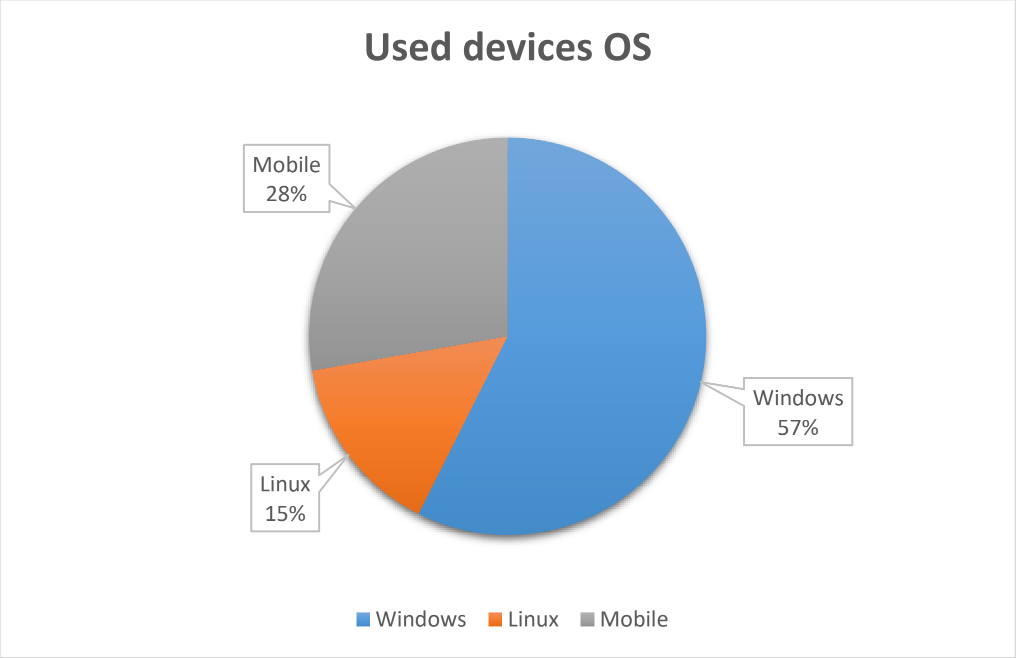


Figure 40: Percentage of used devices OS

Figure (40) shows that the higher majority of our participants use windows as an operating system to their PC's, this result is consistent with our society users since most of them use windows machines.

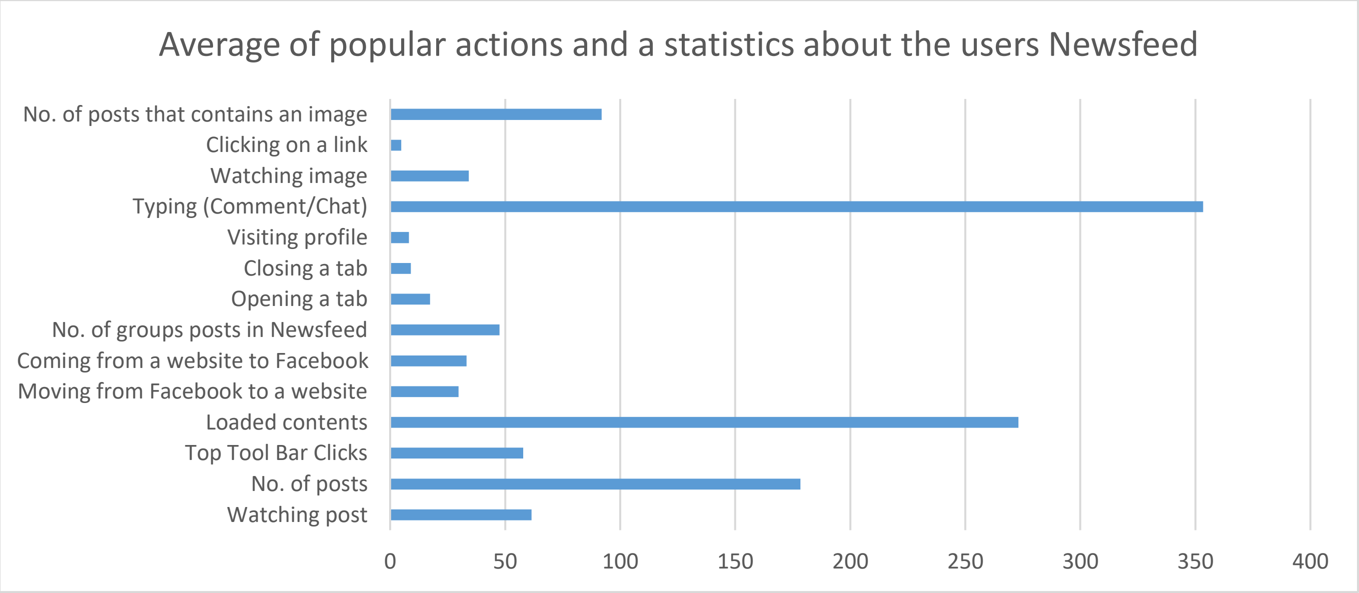


Figure 41: Average of popular actions with statistics about the users Newsfeed

Figure (41) shows number of actions that used to be done by the users. The results show that typing; either by commenting or chatting was the most popular action, other actions along with its popularity can be inferred from the figure.

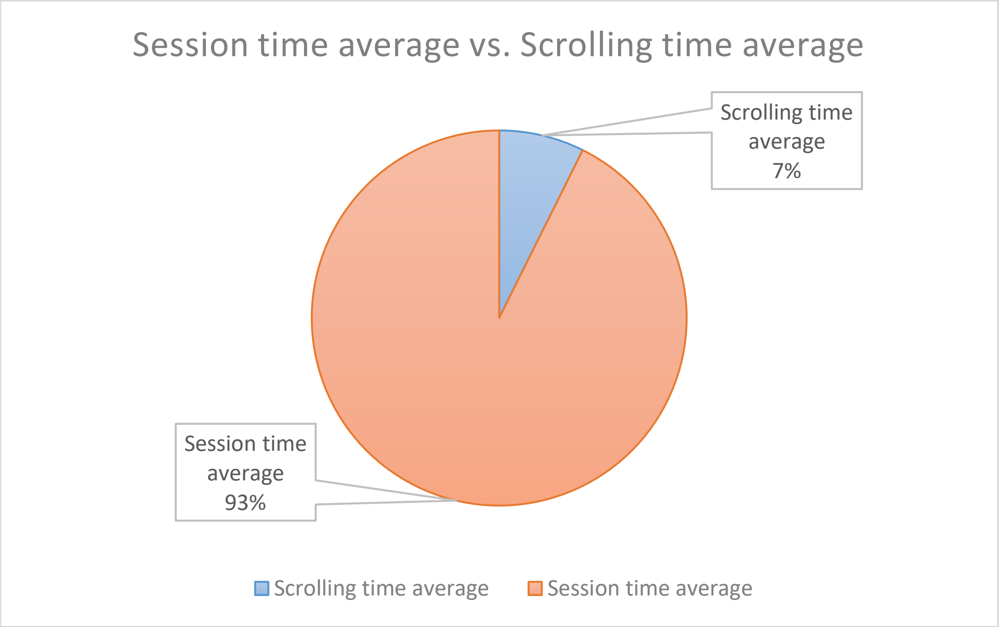


Figure 42: Percentage of session time average vs. Scrolling time average

Figure (42) shows the average of session time compared to the average of scrolling time of that session time. From it, it is clear that the average scrolling time is small compared to the whole session time in average. This result may seem inconvenient, but this is because of the lake of data.

4.3: Results

Generally, our results were not representative due to three main reasons: (1) the lake of participance from the users (2) some of the participated users just downloaded the plugin and they did not send any data (3) some other users submitted the data just before the plugin completed the collection process, this resulted in lots of empty fields in some data samples.

The proportion of male participants in our study was 75% of the total users. In the same manner, English was the preferred language for most users. We also found that Facebook API's request the most permissions for Email, and the average scrolling time represents only 7% of the whole average session time. It is also clear from the results that typing either by commenting or chatting with friends was the most popular action to our participants. In [2] the average session time was 31.40 minutes, this time starts when log in to Facebook happened and ends when either log out or closing the last open Facebook tab in the browser, whereas in our results, the average session time was 7.24 minutes, which is much more less than [2]. This could have given us an indication about Facebook to be an aging network if our data was enough and complete.

Trying to find the best developing technique for the android application costed us plenty of time as mentioned in section 3.3.1 (Design options for application). We managed to build the application (FALC), but there was not enough time to publish it and give the users the opportunity to download it. So, we did not have any data samples from the application. As a result of that, analysis part regarding the application was missed from our documentation.

Ch5: Summary and Future Work

5.1: Summary

In this project we developed and redesigned behavioral data collection tools for Facebook. We also developed Python scripts in order to analyze the collected data. We mainly programmed two data collection tools. First, Facebook Privacy Watcher and Analyzer (FPWA): a browser plugin developed mainly using jQuery and JavaScript Programming languages to collect data from chrome browser at client side. Second, Android application; Facebook Activity Log Collector (FALC) that was developed as a hybrid application which combines the native development with web technology to benefit from what we did in the plugin after a long and very hard journey in trying to build a native application capable of achieving our purpose, but all our attempts have not worked.

The Behavioral data we collected by FPWA plugin can be classified into four main types: (1) some user's actions, (2) user's - friend list, groups and pages -, (3) apps permissions, and (4) basic demographical information. Whereas the collected data by FALC application was from activity log of Facebook account only, the app collected the actions that were stored in activity log three days earlier. In this project our aim was to update and redesign an outdated software tools that were used in some previous similar studies [1]and [2].

After sharing FPWA and the application in both Chrome store and Google play store, we got a small number of downloads, so our collected data sample was not enough to update the results of the previous studies [1] and [2]. Despite that, we developed an analysis program using Python to analyze the collected data sample knowing that our analysis program is able to analyze any amount of collected data that is in JSON format.

5.2: Future work

Other researchers may work on maintaining both the plugin and the application to keep them updated and working despite all changes that may happen to Facebook website. They may also work harder on advertising process, in order to obtain a larger sample than the one we got. By that the analysis results would be more comprehensive and representative than ours. So, these results would be valid to update the results of the previous studies.

It would be also valuable for the other researchers to develop a new version of smart devices application to work with iOS, and to upgrade the plugin to work with other browsers like Mozilla Firefox, Opera, or Microsoft edge.

References

[1] M. S. D. P. M. V. T. S. Thomas Paul, "C4PS - Helping Facebookers Manage their," Springer-Verlag, Berlin, 2012.

[2] D. P. T. S. Thomas Paul, "The User Behavior in Facebook and its Development from 2009 until 2014," arXiv, Berlin, 2015.

[3] T. R. M. C. V. A. F. Benvenuto, " Characterizing user behavior in online social network.," IMC, Chicago, 2009.

[4] A. F. B. K. W. W. Fabian Schneider, "Understanding online social network usage from a network perspective," *IMC,* pp. 35-48, 2009.

[5] E. F. F. A. L. A. G.-J. H. Pasquale de Meo, "Analyzing user behavior across social sharing environments," *TIST,* vol. 5, 2013.

[6] E. B. J. M. K. T. M. L. I. R. Lars Backstrom, " Center of attention: How facebook users allocate attention across friends.," *ICWSM,* 2011.

[7] M. S. M. T. Bruce Ferwerda, "Personality Traits and the Relationship with (Non-) Disclosure Behavior on Facebook," *WWW '16 Companion,* pp. 565-568, 2016.

[8] M. H. D. B. Emily Stark, "Symmetric Cryptography in Javascript," in *Computer Security Applications Conference*, 2009.

Appendices

- **Appendix A: FAQ (Frequently Asked Questions)**

Quoted from the official website: https://fba.ppu.edu

- Browser Plugin

○ What is FPWA?

FPWA (*Facebook Privacy Watcher and Analyzer*) is a browser extension (i.e. plugin) for Google Chrome. It collects data about the privacy settings as well as about behavior of Facebook users. It also helps users to customize their privacy settings on Facebook.

○ Which data is collected by FPWA?

FPWA collects the following data:

- Hash value of the Facebook ID.
- Counter indicating how the user spent watching the post.
- The type and visibility of timeline entries.
- Number of friends.
- Number of photos and labels.
- Commenting and chatting time
- Countries from which we receive the feedbacks.
- Demographical data, e.g. the gender and age of the user.
- The usage pattern w.r.t. actions.
- The use of third party apps.
- Permissions for the third party apps.
- The content shared by users.
- Measuring durations and frequencies of sessions.
- The popularity of different Facebook functionalities and apps permissions.
- Dynamics in user behavior.
- How long different types of newsfeed entries stay in the viewport of the browser before being clicked?
- How many newsfeed posts have been viewed on average per session?
- What are the types of accessed content (posts of images)?

○ How do we preserve your privacy?

Your data will be collected without your identity (e.g. we know that "someone" liked some post, but we cannot know who did that).

Furthermore, your data is accumulated with the data of other participants. The collected data will be encoded using hash algorithms (i.e. your data is not reversible) which means that the database of FPWA plugin will contain just the hashed data. Also, our source code is available online.

- **Why do we collect the aforementioned data about you and other volunteers?**

We need (and use) your data only for academic/research purposes. In particular, the objective of our study is twofold:

First, evaluating data exposure in Facebook.

The huge set of functionalities Facebook offering to their users attracts lots of people from different age rages to join Facebook. This will make them vulnerable to different kinds of risks. Protecting personal data means a lot to most Facebook users. Therefore, we provide a simple (color-based) mechanism that will help our volunteers to set their privacy settings in a way that limit private data exposure.

Second, understanding user behavior in Facebook.

User behavior in Facebook is a critical issue for different parties. For example, authors of alternative online social networks architectures to make well-founded design choices, and Facebook itself, because understanding user behavior impacts their design choices in Facebook, and help them provide the most appropriate functions and features to that user. In addition to that, user behavior in Facebook may concern the user itself, for example, the user might be eager to know statistics about his/her usage of Facebook because he could spend a lot of his time on Facebook at the expense of his duty and responsibilities in the life. These statistics may include: the most Facebook functionalities used, the time user spend on Facebook, or even the time user spend on a specific functionality, and much more of statistics that can be measured.

- **How can you benefit from FPWA?**

First of all, FPWA helps you to better customize your privacy settings in Facebook. It provides an innovative interface to overlook and change privacy on Facebook easily. The basic idea is that people can gather information more easily when it is associated with colors. In the profile page of Facebook and were linked with colors. The idea is inspired from FPW.

Second, once we collect sufficient amount of data about you and other volunteers, you will be able to see information and statistics about your behavior in Facebook. Furthermore, you can compare your behavior to the behavior of other volunteers.

- **How do we store the collected data?**

We use JSON file format to store the collected data.

- Android Application

- **Why this application?**

Recently, most people tend to use smart devices to reach their Facebook account. From that, the need for smart devices application was emerged.

- **How to use it?**

All you need is to login your Facebook account from our application, and then continue your usual behavior on the original Facebook application.

- **How it works?**

The application is collecting the Activity log in your Facebook account. it collects your actions that happened before 3 days, you can view the collected data before sending to the server.

- **How do we preserve your privacy?**

Your privacy data is completely anonymized, every data element (your id, group id, page id ...etc.) is hashed to anonymize your identity as well as in the Chrome browser plugin.