



D7.2

Web & Mobile Interface Templates - v1

v 1.0 2014-11-28

Apostolos Kousaridas (VELTI), Timotheos Kastrinogiannis (VELTI), Steven Strachan (CEA),
Noel Catterall (HWC)

The scope of this deliverable is the design of the first version of the Web & Mobile Interface for the overall platform regarding mobile and web clients, based on the requirements that are collected by WP2, WP4 and WP6. These requirements are used for user interaction and software engineering design. The focus of this deliverable is the on User Interface (UI) and User Experience (UX) design as well as on the state-of-the-art mobile and web client technologies. DataBait is the web-based application that provides to the end user the services of the USEMP platform, following the responsive design approach, while HTML5, CSS and JavaScript are the main technologies that are used.



Project acronym	USEMP
Full title	User Empowerment for Enhanced Online Presence Management
Grant agreement number	611596
Funding scheme	Specific Targeted Research Project (STREP)
Work program topic	Objective ICT-2013.1.7 Future Internet Research Experimentation
Project start date	2013-10-01
Project Duration	36 months

Workpackage	WP7
Deliverable lead org.	Velti
Deliverable type	Report
Authors	Apostolos Kousaridas (VELTI), Timotheos Kastrinogiannis (VELTI), Steven Strachan (CEA), Noel Catterall (HWC), George Mourikas (HWC)
Reviewers	Symeon Papadopoulos (CERTH), Marita Holst (LTU)
Version	1.0
Status	Final
Dissemination level	RE: Restricted Group
Due date	2014-09-30
Delivery date	2014-11-28

Table of Contents

- 1. Introduction** 2
- 2. USEMP User Interface Templates** 3
 - 2.1. Requirements of USEMP Tools and Design Principles..... 3
 - 2.1.1. DataBaitp-GUI..... 4
 - 2.1.2. DataBaitp -Browser 4
 - 2.1.3. DataBaitp -FB..... 5
 - 2.2. USEMP UX/UI..... 6
 - 2.2.1. Design Principles (UI perspective)..... 6
 - 2.2.2. Fluid and Responsive Design..... 6
 - 2.2.3. Assets and Media..... 9
 - 2.2.4. Application Map and UX Flows.....10
 - 2.2.5. Mock-up Analysis11
 - 2.2.6. Web and Mobile Device Technology23
 - 2.3. Client side communication with backend25
 - 2.3.1. Registration Interaction25
 - 2.3.2. Login Interaction.....25
 - 2.3.3. Facebook Login Interaction26
- 3. Personal Data Visualisation**.....27
 - 3.1. USEMP Personal Data and Privacy Profiling.....27
 - 3.2. SOTA Analysis of Data Visualisation Methods27
 - 3.3. Towards USEMP Visualisation Methods28
 - 3.3.1. Privacy28
 - 3.3.2. Notification Management.....31
 - 3.3.3. User Trackers.....32
- 4. Conclusions**.....34
- References**.....35

1. Introduction

The USEMP platform aims at providing tools that empower online users to control their personal data and to understand how they are used by third parties (i.e., online social networks, web sites etc). The USEMP platform is the result of the collaboration and the integration of different research areas such as multimedia information extraction, social network profile mining, monetary valuation of personal information, information visualization and interaction as well as legal and social studies.

DataBait¹ is the web-based application that provides to the end user the services of the USEMP platform in a consistent, easy to use and well integrated way. DataBait acts as the graphical user interface through which the USEMP system can convey to the end user results of the USEMP system processes and services. A non-exhaustive list of services includes: USEMP log-in and registration, USEMP Online Social Network (OSN) Log-in, user profile, user privacy inferences, user trackers, monetization insights, audience/influence, USEMP privacy notifications (pop-up), tracking services/activities control, etc. For the design of the DataBait, the responsive design approach has been selected. The main technologies that have been used are HTML5, CSS and JavaScript.

In D7.1 the architecture of the USEMP Platform has been defined, while in this document we are focusing on the design of the first version of the web and mobile interface of the DataBait application. In this deliverable, Section 2 initially gives an overview of the requirements and design principles of the USEMP system focusing on of the USEMP client side tools. Guidelines for the creation of User Interface (UI) and User Experience (UX) design are presented, while the UI/UX of the DataBait application is described, by presenting also the application map. Section 3 provides an analysis of the state of the art for the visualization of data and describes the current USEMP work for the visualization of information related to privacy, notification management, and user trackers. Finally, Section 4 concludes and summarises the most important issues of the Deliverable.

¹ The term LIO that has been introduced in D7.1 in order to refer to the main client side tools (LIOp-GUI, LIOp-BROWSER, LIOp-FB) has been replaced by the term DataBait (DATABAITp-GUI, DATABAITp-BROWSER, DATABAITp-FB)

2. USEMP User Interface Templates

2.1. Requirements of USEMP Tools and Design Principles

The requirements and design principles of the USEMP system and specifically of the USEMP client side tools (USEMP_TOOLS) that consist of the DATABAITp-GUI, the DATABAITp-BROWSER, and the DATABAITp-FB, which have been introduced in D7.1 and D2.2. These requirements are briefly presented below. The USEMP_TOOLS are purpose-made for the USEMP system.

- DATABAITp-GUI is an application that acts as the graphical user interface through which the USEMP_SYSTEM can convey to the OSN_USER results of the USEMP_SYSTEM processes and services (done by utilising the backend services and processes).
- DATABAITp-BROWSER is an application in the form of a plug-in made specific to collect the digital-trail of the end-consumer information generated while interacting with the OSN_USER browser where the plug-in is installed.
- DATABAITp-FB is a plug-in made specific for Facebook, configured specifically to be able to access (with the permission of the OSN_USER) the digital-trail generated by the OSN_USER while interacting with the OSN (in this case Facebook).

DATABAITp_GUI Requirement	Requirement Analysis	WP Requirement Definition
DATABAIT User Account Management	<ul style="list-style-type: none"> • USEMP Log-in (via USEMP_AA Server Communication) • USEMP OSN Log-in (OSN Auth.) • USEMP User Registration 	WP2 and WP7
DATABAIT User GUI Visualization	<ul style="list-style-type: none"> • USEMP User Profile • USEMP User Privacy • USEMP User Trackers • USEMP Monetization Insights • USEMP Audience / Influence • USEMP Privacy Notifications (pop-up) 	WP2, WP4 and WP6
Explicit Privacy Control	<ul style="list-style-type: none"> • Tracking Services/Activities Control • Audience Management Settings • Personal Data Protection Setting • Notifications Setting • Term of Use • USEMP Service Deactivation 	WP2, WP4 and WP7
Technical Requirements	<ul style="list-style-type: none"> • Cross Web Browser Compatibility • Cross Mobile Browser Compatibility • No Deployment / Installation Required 	WP2 and WP7

Table 1 DATABAITp_GUI Features and Goals

Table 1 summarises the DATABAITp_GUI goals and expected features, collected via USEMP WP2 (use cases and UST), WP4 (Legislations) and WP6 (Privacy and Monetisation

Design Interface) requirements and specifications. The following subsections provide a short overview of the main goals and requirements of each USEMP_TOOL.

2.1.1. DataBaitp-GUI

As it is also presented in Table 1, the DATABAITp-GUI will enable USEMP users to:

- *Manage their USEMP Account:*
 - Log-in
 - OSN Login (OSN Authentication)
 - User Registration
 - USEMP Privacy Setting Configuration
- *Visualisation of USEMP Services:*
 - User Profile Indicators
 - User Privacy Leaks
 - User Trackers Identification, filtering and Do Not Track Policies Creations
 - Personal Data Monetisation Insights
 - Audience / Influence Management
 - USEMP Privacy Notifications (pop-up)
 - Compare with OSN friends data

The main design requirements that were considered in the design of the DataBaitp-GUI are:

- The operation on all types of desktop and mobile devices.
- The creation of a layout that supports large audiences, with multi-browsers, multiple OS and supporting desktop, tablet and high-end mobile devices.
- A privacy-by-design application that is in line with the EU Legislations and Directives.
- The support of a set of UX (e.g., correct navigation, notifications configuration, simplicity, etc.) and UI design principles in order to optimise the end user Quality of Experience (QoE).
- The adoption of two levels of accessibility features for the support of USEMP user with a wide range of disabilities: a) W3C Web Content Accessibility Guidelines 2.0 Compliance, b) Web Browser Accessibility Optimisation Settings.

2.1.2. DataBaitp -Browser

As detailed in the previous section, the DATABAITp-BROWSER key goals are:

- To track USEMP users' volunteered and behavioural data generated via their web browser(s) (either explicitly or implicitly) and feed the latter into the USEMP back-end system via the USEMP API, denoted as USEMP_API.
- To track USEMP users' trackers, in terms of 3rd party online tracking and analytics services that monitor and collect end-users digital trail on their web browser(s).
- To enable USEMP users to identify their web browser tracker(s) and thus, define fine-grained do-not-track (DNT) rules (in a flexible and intuitive manner) that will be realised and reassured by the DATABAITp-BROWSER.
- To enable USEMP users to control the operation of the tool, in terms of making them aware of every action performed by the DATABAITp-BROWSER and how the latter affects their privacy.
- To enable USEMP users to authenticate/authorize via the DATABAITp-BROWSER in their USEMP account in a common and transparent way across browsers.

- To enable USEMP users to visually interact with the DATABAITp_BROWSER in an intuitive and QoE optimised way for each of the above goals.

DATABAITp_BROWSER's operation with USEMP end-users' browser(s). Thus, any potential technical solution should take into consideration this particular requirement.

2.1.3. DataBaitp -FB

The DATABAITp-FB is one of the key enablers of the USEMP System, since it serves as the interface of the latter with the OSN (Facebook) of the USEMP users. Specifically, the key goals of the DATABAITp-FB are detailed in the following:

- To track USEMP users' personal and behavioural data that are generated and/or disseminated by them via their OSN. Such data can be either historical (i.e., data generated by the user prior creating a USEMP user account, stored in the OSN system) or real-time (i.e., data generated by the user upon creating a USEMP account) and should be fed into the USEMP back-end system via the USEMP_API. The latter data will be collected via the available OSN APIs (Graph API²), denoted as OSN_API.
- To enable USEMP users to visually interact with the DATABAITp-FB in an intuitive and QoE optimised way.
- To support mechanisms that minimize the imposed network traffic overhead (due to DATABAITp operation) in order not to affect the performance of the related devices-browsers.
- To enable USEMP users to control the operation of the tool in terms of making them aware of every action performed by the DATABAITp-FB and how the latter affects their privacy.
- To enable USEMP users authenticate/authorise via the DATABAITp-FB in their USEMP account in a common and transparent way across browsers.
- To enable proper throttling mechanisms in order to reassure their efficient and continuous operation, given the OSN API call limitations.

² <https://developers.facebook.com/docs/graph-api>

2.2. USEMP UX/UI

2.2.1. Design Principles (UI perspective)

The objective of this chapter is to describe a number of guidelines for the creation of content and different approaches on delivering that content, which is intended to be used by the USEMP users using different means (i.e., smartphones, tablets, personal computers).

Smartphones & Tablets are becoming increasingly popular as part of consumers digital lifestyle in European countries. These devices are used for consuming entertainment and increasingly learning content. The most popular vendors of operating systems (and in some cases devices themselves) are Apple with its iOS operating system and Google with its Android operating system. Content and applications for these devices are primarily developed in three formats:

- **web applications:** these are rich media mobile sites that can also operate without network connectivity after they have been accessed. These are primarily developed in HTML5 and JavaScript taking advantage of the increasing support from tablet and smartphone vendors.
- **native applications:** these are native applications which are distributed by the device vendors. Each vendor has usually its own application development ecosystem. Examples are: Objective C and iOS SDK for Apple iOS devices and Java and Android SDK for Google Android devices.
- **native media formats:** these are media formats for content that can be published and controlled in vendors content stores. For example Apple has developed the iBook format for its Book storefront to iOS devices and Amazon has developed KF8 format for its own Kindle store devices.

The main development characteristics of web application are listed below, which are also some of the main challenges that we have to address in the design of the USEMP UI/UX:

- Multi-platform – the developed web app code can run in multiple platforms (operating systems, browsers),
- Instant iteration - no delays in the publication of new features and content (since there is no approval process with the application update),
- No approval process - no limitations placed on content or subject matter,
- Write-Once run everywhere does work only for a limited set of features.

2.2.2. Fluid and Responsive Design

There are two approaches in the design and implementation of rich media applications that are used to tackle a) rendering without defaults in various device screen sizes and b) taking advantage of the available real-estate of the device as much as possible:

- 1) Fluid design
- 2) Responsive design

They both allow websites to be viewed in mobile devices and various screen sizes, ultimately providing visitors with a better mobile user experience.

1) Fluid design

In a fluid website layout, also referred to as a liquid layout, the majority of the components have percentage widths, and thus adjust to the user's screen resolution.

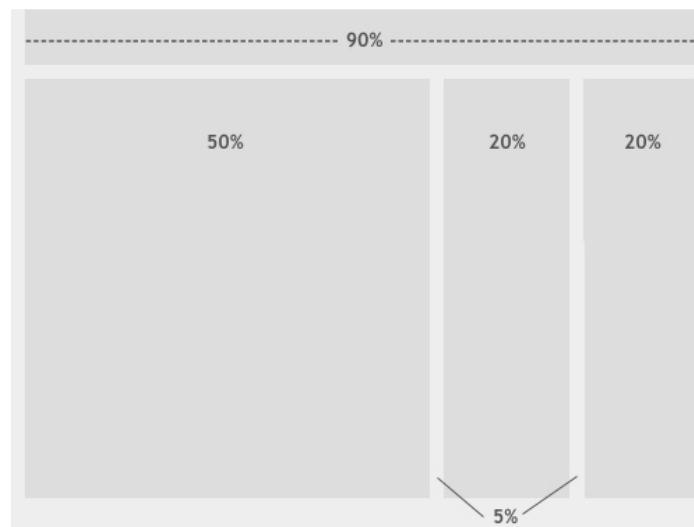


Figure 1. Example of fluid design

The Figure above shows a fluid (liquid) website layout. While some designers may give fixed widths to certain elements in fluid layouts, such as margins, the layout in general uses percentage widths so that the view is adjusted for each user. The fluid web app design better fits the cases which a mobile-only delivery targeted approach is being utilized.

Advantages of Fluid Design

- the fluid design can be easily implemented and understood by rich media authors since it has one design that fits all factors,
- the web content will render without defaults to both smartphones and tablets.

Disadvantages of Fluid Design

- it does not take into advantage the additional screen real estate which is available to tablets and larger devices (usually the design is based on the smallest screen to be supported), in most cases a different fluid design can be devised for larger factors devices (like tablets).

2) Responsive design

Responsive design takes the approach of progressively defining additional elements that can be added as the resolution/width of the screen device increases. The responsive design approach is based on a number of significant resolution screen sizes that are used commonly by devices. Figure 2 depicts a plot of the most important resolutions for screens from desktops (PC) to tablets and smartphones (resolutions below 240x320 are really reserved for mobile phones which are not smartphones).

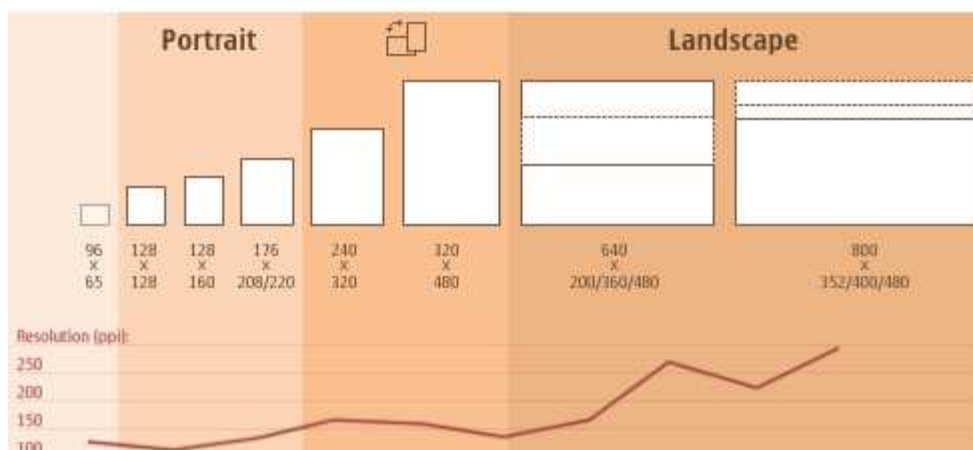


Figure 2. Popular devices screen resolutions and their popularity with new devices

The more common approach for responsive design is to define three (3) design variants for three large classes of devices (mobile/tablet and desktop). The designs are progressively adding more elements as the screen factor increases (see Figure 3 for an example of such a responsive design on which existing elements are re-positioned in the screen and new elements are added as more screen real-estate becomes more available).

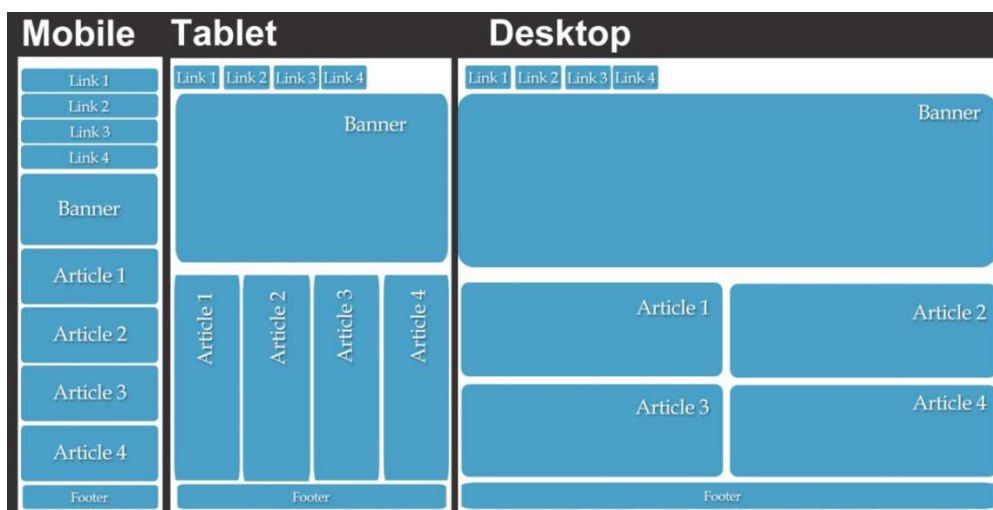


Figure 3. Example of responsive design

With the recent advances in web standards and responsive design techniques it is now possible to provide a progressive, gradually enhanced experience across a wide array of browsers, using one HTML5 markup document and a variety of different Cascading Style Sheets (CSS). This approach does not selectively deliver content to the user through browser sniffing or server-side device detection, but rather is requested by the mobile browser itself by only rendering the supported web content. Mobile devices that fall under the smartphone category currently have superior mobile browsers and support for HTML5 when compared to feature phones.

Advantages of Responsive Design

- responsive design provides additional content taking advantage of the additional screen size for larger devices like tablets and desktops (PCs)
- the web content will render without defaults to both smartphones and tablets

Disadvantages of Responsive Design

- responsive design requires additional effort and training to develop how the additional content should be placed by rich media applications designers for additional screen factors

The responsive web app design is aligned with the development approach in which one web app handles both desktop and mobile.

2.2.3. Assets and Media

Another challenge with developing (mobile) web applications is how to handle the assets and media that will be combined to produce the final rich media application. The most important categories to be supported are:

- Images
- Videos
- Documents

Images: images are generated in various formats and sizes, the most ubiquitous format being JPEG. The rich media content should have images in high enough resolution so that it can be adapted for all screen sizes (for the fluid design approach) or for a number of resolutions (for the responsive design approach). The following is a list of guidelines for image formats:

- for non-transparent images, use JPEG and lower the quality settings to achieve a low file size,
- for transparent images, use PNG,
- for transparent images with a small color palette, use PNG-8,
- merge transparent PNG animation sequences with background, to produce low-size JPEG equivalents.

Additionally the size of the images themselves should be optimized for the mobile rich media design so that the overall time to download and render all the images is small enough to make the final rich media content highly responsive to the end user.

Video/audio: video and audio support also varies for smartphones, tables and desktops. There are two type of services used for building rich media web apps:

- streaming that streams audio and video from a streaming server to the smartphone-tablet device,
- downloaded audio and video files that allow the user to download the audio/video file to his own device and reproduce it locally.

Video format and support in mobile and tablet devices varies significantly from desktop but also between different mobile devices, therefore it will be required to have video produced on a number of different formats, or leave that to be handled by a video server which is considered best practice. The complexity is obvious when using an example case of a generic video encoding spec (video: H.264 codec with video bitrate up to 160 Kbps & 48kHz & audio: AAC-LC codec with audio bitrate up to 160 Kbps & 48kHz). Even though this might seem compatible with most of the mid to high level mobile devices for iOS and Android, in fact support for it cannot be ensured. There might be requirements for different codec and

sizes for tablets and desktop, and native and web player compatibility might not be in line with the acceptable quality standards.

Documents: various document formats can be included to mobile web rich media apps provided there are native applications installed to the devices to handle the format they are created on. As an example PDF files, one of the most popular format for document sharing, can be included to rich media content and experienced in tablets and smartphones provided a PDF application is installed/supported by these devices. PDF applications are supported in Android and iOS devices.

2.2.4. Application Map and UX Flows

In this section, the main pages of the DataBait application users' main activities flow are described and mapped into the expected application's features (as detailed in Section 2.1).

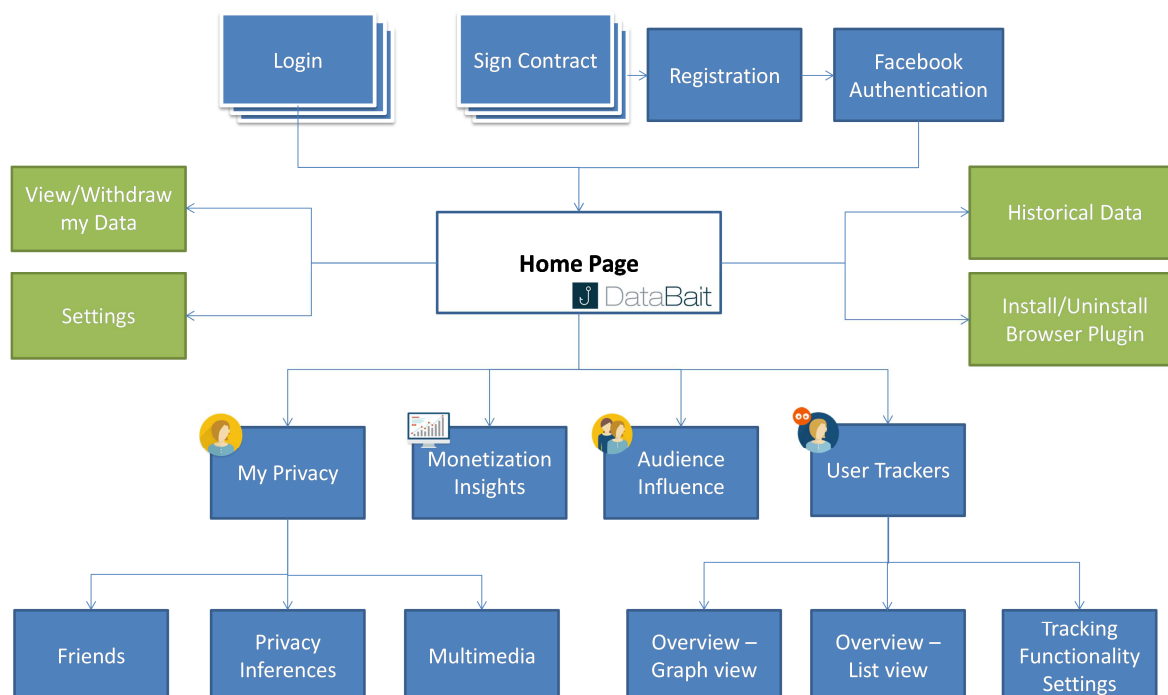


Figure 4 DataBait High-Level UX Flow

The first time that a user visits the DataBait app, she should register to the USEMP system and create the USEMP account. For this process, the initial step involves the signing of the USEMP Data License Agreement by the end user (DLA). Through this contract (i.e., DLA) the user becomes aware about the research scope of the USEMP project, how her personal data will be collected and will be processed by the USEMP consortium as well as about the permissions of all parties. With the signing of the contract, the user creates her account to the USEMP system. Thereinafter, the DataBait users are asked to give to the USEMP_SYSTEM their authorization to access their Facebook account. Through this declaration the USEMP System is allowed to track USEMP users' personal and behavioural data that are generated and/or disseminated by them via their OSN.

After the registration phase, the corresponding DataBait user is able to login to the DataBait web application using the DataBait credentials (username, password). After the login screen and the successful authentication, the user is redirected to the Home Page, which navigates the DataBait user to the USEMP services. The main four screens and corresponding functionalities that the DataBait application provides are:

- My Privacy
- User Trackers
- Monetisation Insights
- Audience Influence

The above four main functionalities have been identified to include the functional requirements that have been described in D2.2 and D7.1, taking into account the the research work in WP5 and WP6. In addition, the Home Page provides the link for the installation (and uninstallation) of the DataBaitp_Browser plugin by the DataBait user to the browser that she uses. As it is mentioned in Section 2.1.2 the main goals of the DataBaitp_Browser plugin are:

- To track USEMP users' volunteered and behavioural data generated via their web browser(s) and feed the latter into the USEMP back-end system.
- To track USEMP users' trackers, in terms of 3rd party online tracking and analytics services that monitor and collect end-users digital trails on their web browser(s).
- To enable USEMP users to identify their web browser tracker(s) and thus, define fine-grained do-not-track (DNT) rules.

In addition, the DataBait user is able from the Home Page to enable (or disable) the collection and storage of personal data collected from the Facebook account of the user ('Historical Data'). The user can also view the already collected data and proceed to the deletion of the latter from the USEMP database. After enabling the collection of Historical Data, the DataBait application can track the USEMP users' personal and behavioural data generated and/or disseminated via the Facebook account provider. Such data can be either historical (i.e., data generated by the user prior creating a USEMP user account, stored in the OSN system) or real-time (i.e., data generated by the user upon creating a USEMP account) and should be fed into the USEMP back-end system.

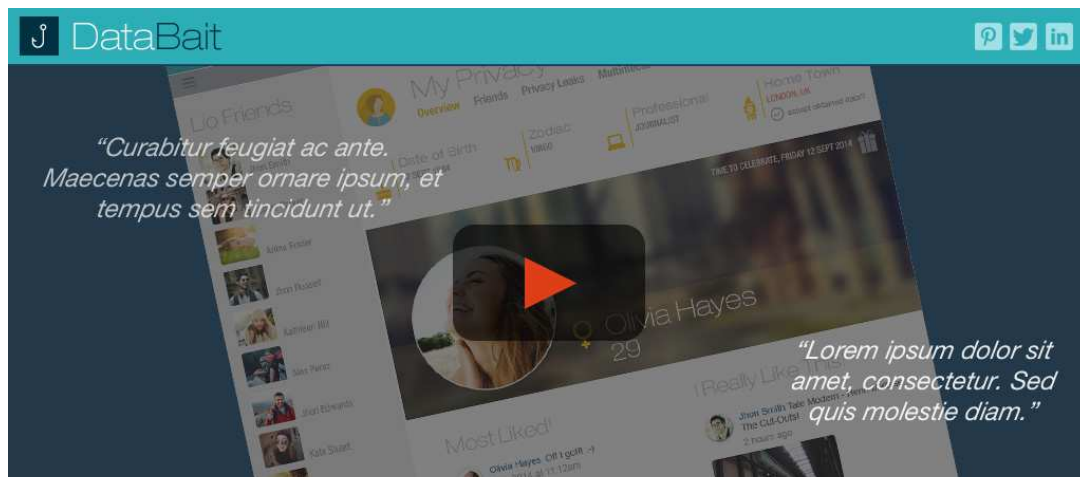
More details for the key functionalities of the DataBait application are provide below, where the screens of some of these functionalities are presented and analysed. In this deliverable, we are focusing on the "My Privacy" and "User Tracker" services.

The mock-ups for the Monetisation Insights and for the Audience Influence will be analysed and provided in D7.4. Monetisation Insights section will provide to the DataBait users an intuitive economic value of their digital personal data that are either directly shared in social networks (e.g., Likes of FB) or indirectly collected by various network actors that track their web browser activities. The Audience Influence will visualize how the friends affect the online behaviour of each DataBait user as well as how similar a DataBait profile is to different users.

2.2.5. Mock-up Analysis

The screens that provide the end user with the UI to consume the USEMP functionalities are presented and analysed in this section, taking into consideration the requirements that have been described for the USEMP tools as well as the design principles.

The first screen provides to the DataBait user with some information about the scope of the USEMP research (Figure 5). For that purpose a short video will be also available, which will visualise the background behind the DataBait application for both real users and for those that participate in the pilots (WP8) and contribute to the research. Through this screen the DataBait user will be able either to proceed to the login screen or to the USEMP web site in order to read more information about the USEMP research goals.



DataBait Research

Help us create a privacy enhancing app

Are you sensitive about having control over your online data?
Do you think science could help to achieve this?

Would you be willing to help us, so in the future a new online application can be launched where you, instead of companies, are in control of your online personal data?

Then please help the EU project USEMP to create the DataBait application by giving us access to your facebook and browser data and filling in a questionnaire.

It maybe sounds contradictory but we need personal data to develop an application that secures personal data and gives you control over your data. In a later stage you will be able to use the application that we created.

The data will be anonymized and only used for research purposes. Because we believe a consent is not transparent enough, we really want to make a contract with you about our relation.

By doing this, you help our research forward & if you want, we will keep you updated about the DataBait privacy enhancing application.

Yes, I want to participate in the research

Login and proceed

I want to know more about the research

Visit the project website

USEMP has received funding from the European Union's 7th Framework Programme



Figure 5. DataBait Informative screen

The next screen provides to the end user the login fields and the access to the registration screen for those users that use DATABAIT for first time (Figure 6).

In the case that the registration is selected, the next step includes the signing of the Data License Agreement (DLA) by the DataBait user. In this phase the user can read the contract (Figure 7-a, Figure 7-b) and then proceed to the sign by filling in her Full Name and clicking a checkbox which is an indication of her agreement with the DLA (Figure 7-c). In this screen the user has also the capability to:

- Print the contract,
- Download the PDF document of the contract,
- Send questions or comments about the contract to the USEMP members through e-mail.

The left side of all the screens of Figure 7 include a video, which provide to the DataBait users information about the role of the DLA.

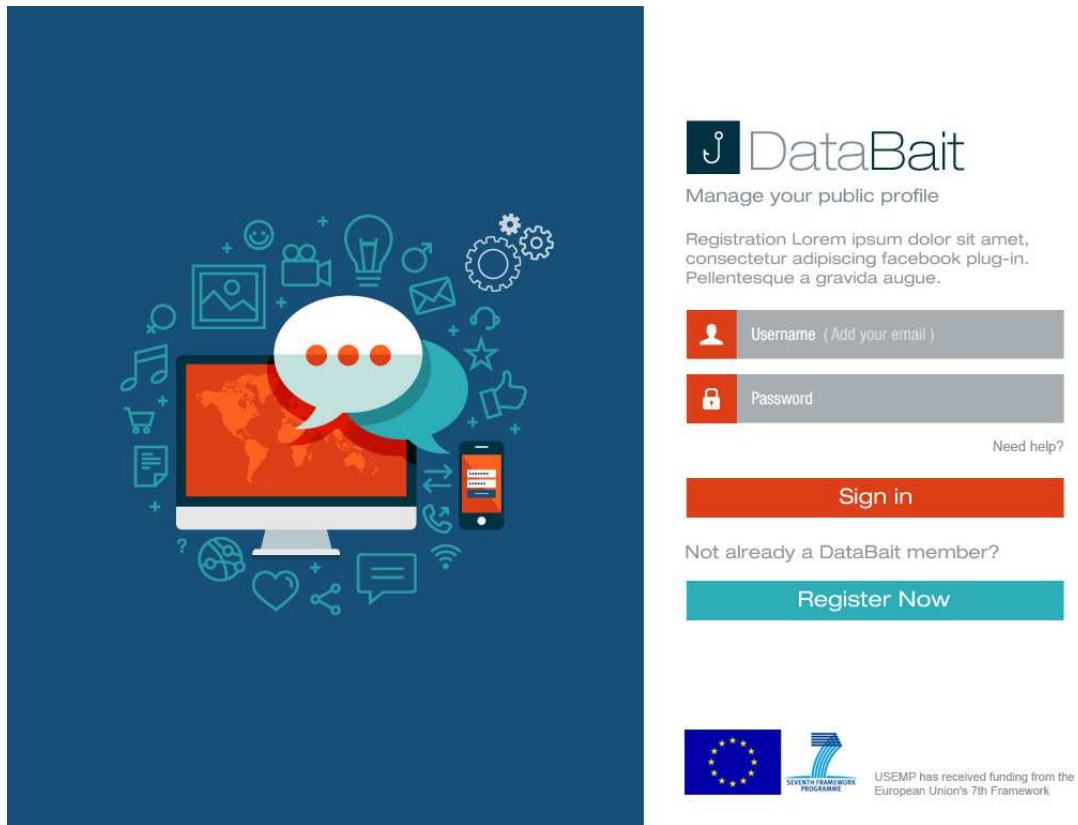
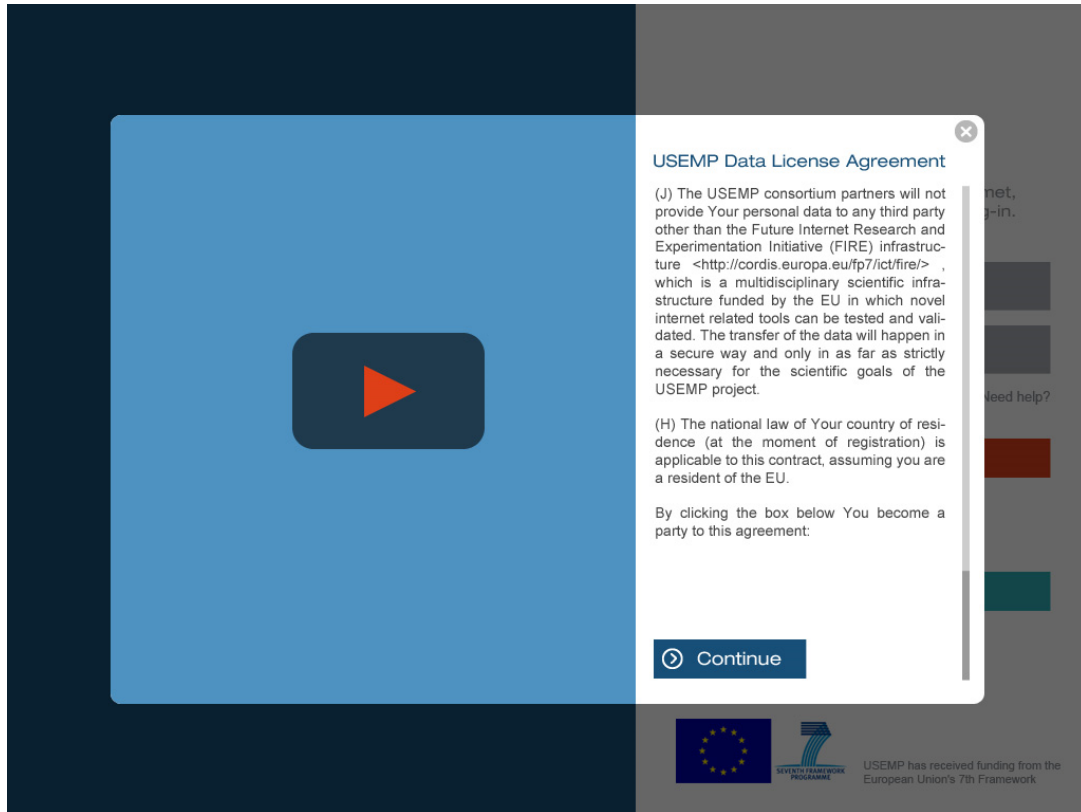


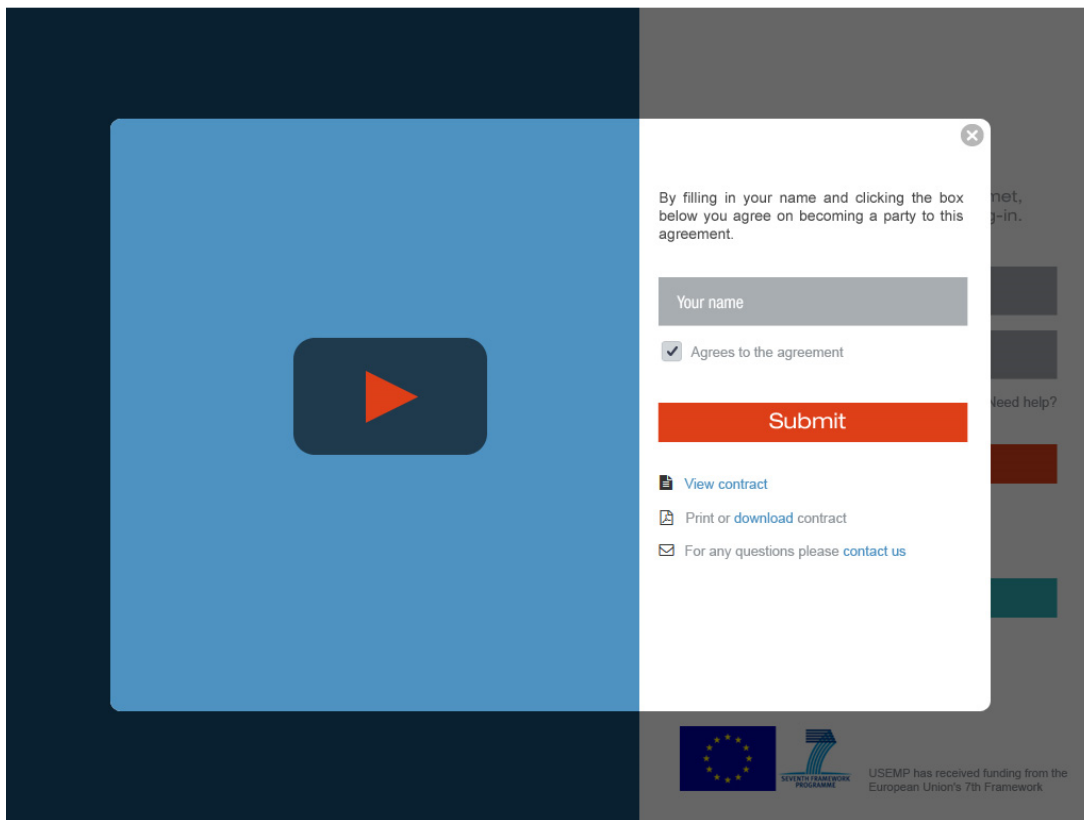
Figure 6. DataBait Sign In screen



(a)



(b)



(c)

Figure 7. DataBait Contract screens

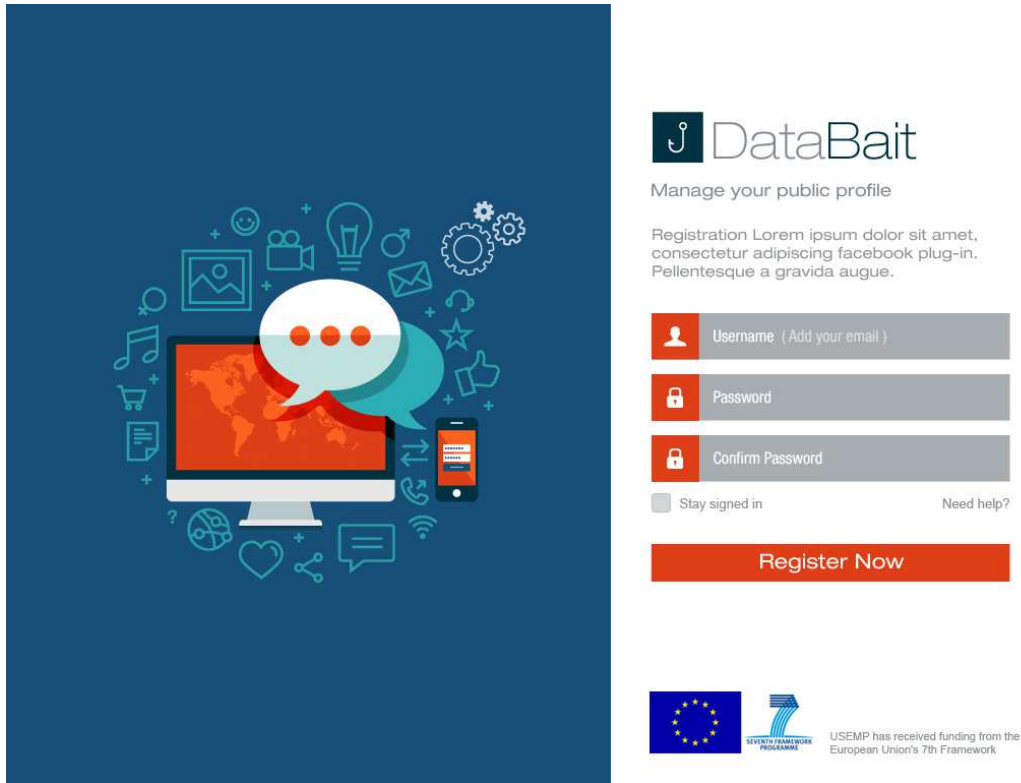
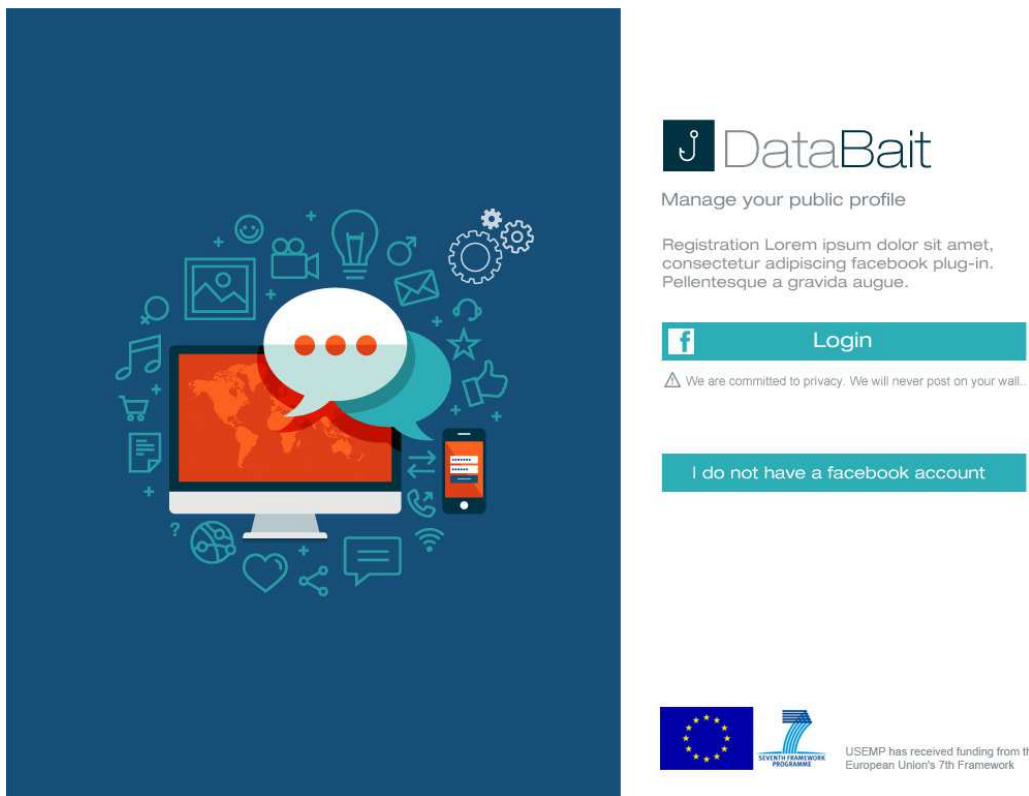
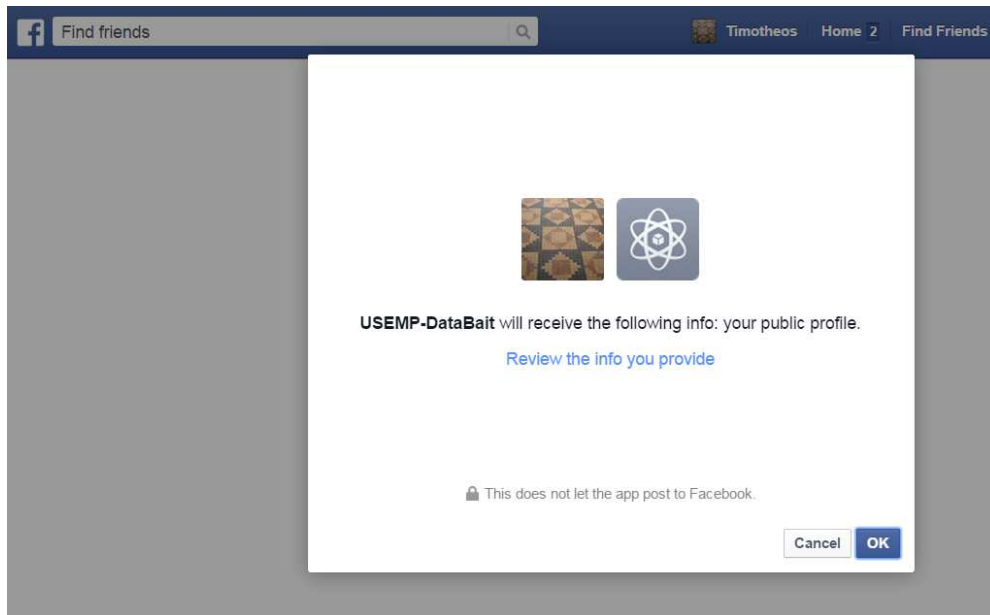


Figure 8. DataBait registration screen

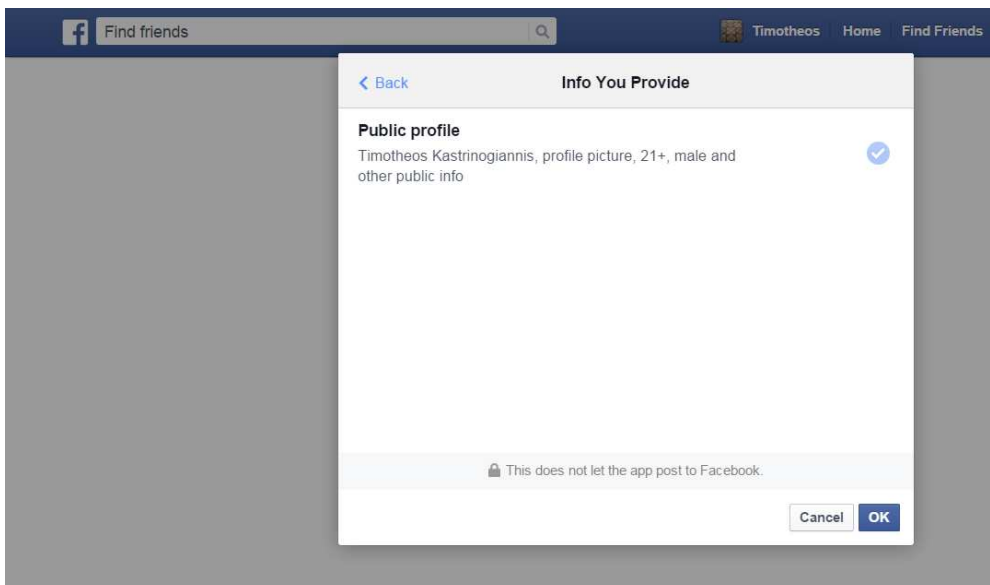
After the signing of the DLA, the next step includes the USEMP Registration (username, password) for creating a unique account, which will be used to access the DataBait services (Figure 8).



(a)



(b)



(c)

Figure 9. DataBait (a) Facebook Authentication screen, (b) Informative page, (c) Provided information

Then, the DataBait user indicates whether she holds a Facebook account and authorizes the USEMP System to be able to access the digital-trail generated by the user while interacting with the Facebook (additional OSNs could be used in future DataBait versions). Figure 9-a depicts the screen for:

- Facebook Registration,
- Facebook Permissions (& learn more section).

For the authentication of the DataBait user with the Facebook services the Graph API is used, which is an HTTP-based API to get data from Facebook's social graph. Figure 9-b presents the Facebook authentication page and the access rights of the DataBait application. Specifically, the information that the DataBait application will get from Facebook using the Graph API are presented in Figure 9-c.

After the authentication phase, the user can use the DataBait Services that are provided by the web application and the web browser plugin. After the login the DataBait user is re-directed to the home page (i.e., screen) of the web application, which has the following core options (Sub-Screens) (Figure 10-a):

- USEMP My Privacy
- USEMP User Trackers (and Do Not Track)
- USEMP Monetization Insights
- USEMP Audience / Influence.

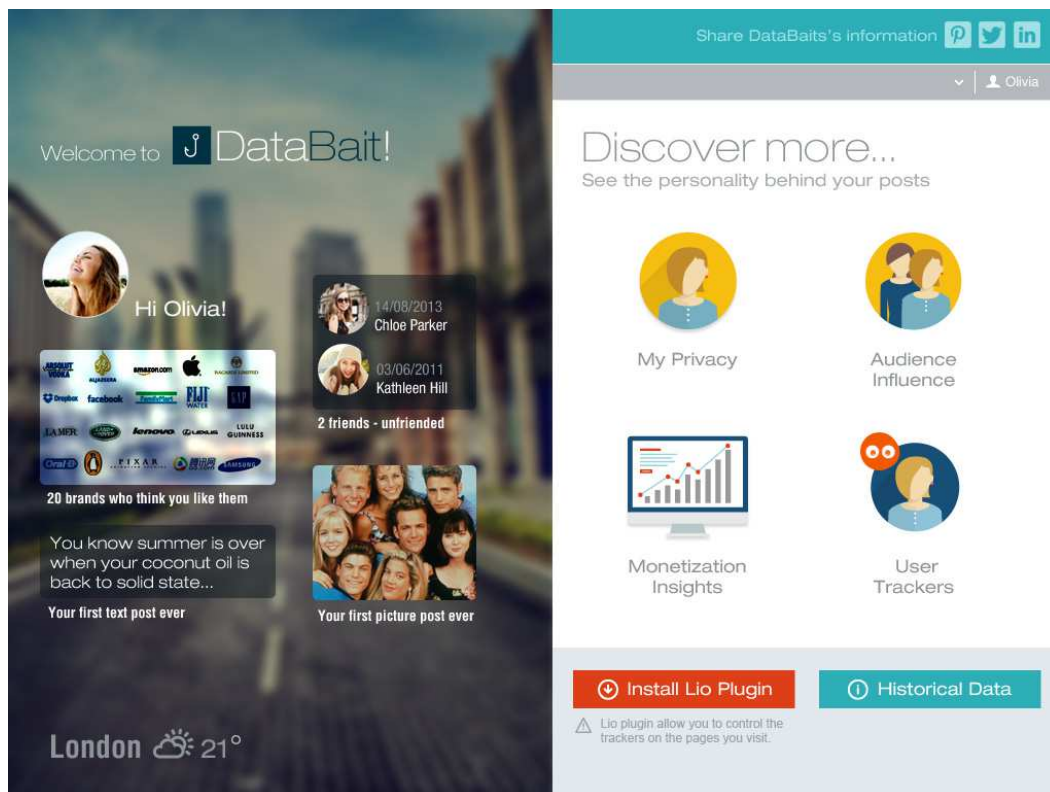
In the same screen the user has also the following capabilities (Figure 10-b):

- DataBaitp_BROWSER Plug-in Install/Uninstall Option (button)
- Historical Data Collection Option and Notification Option (button). The DataBait user is able to enable (or disable) the collection and storage of personal data collected from the Facebook account of the user ('Historical Data'). The UI of this screen is presented in (Figure 11).

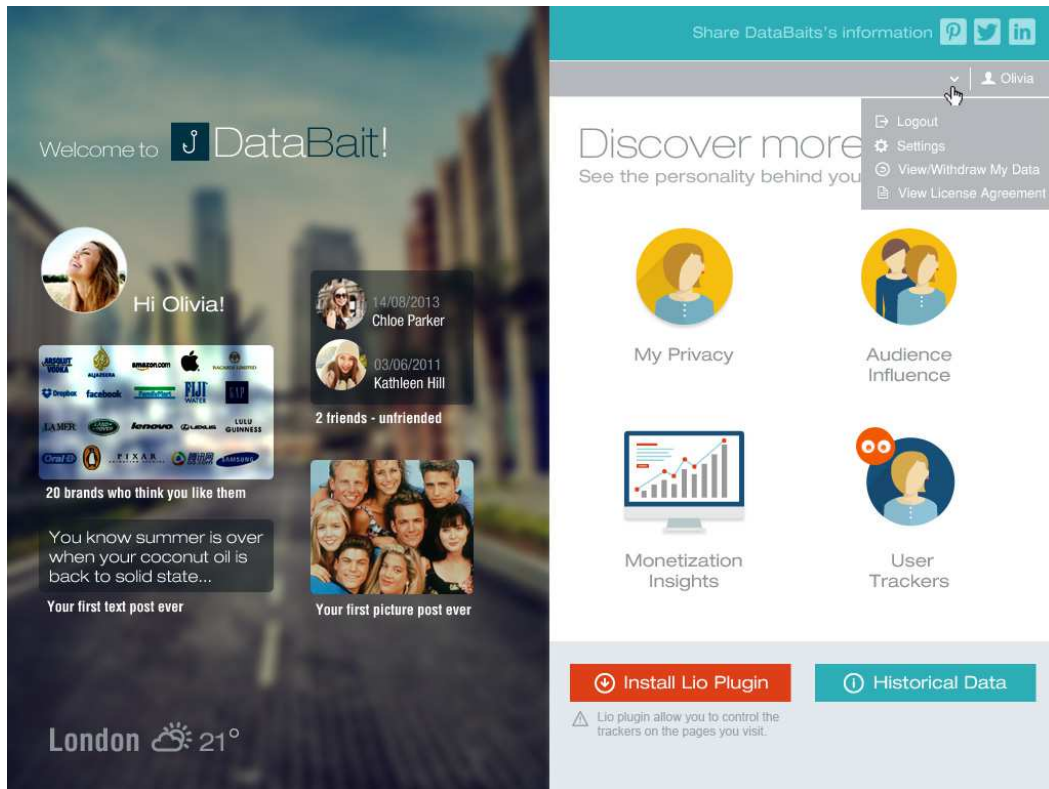
From the Drop-down list in the upper right part of the home page, the DataBait user has the following capabilities (Figure 10-b):

- DataBait Log-Out Option
- User Settings
- View and withdraw collected data. Figure 12, depicts the screenshot of the page.
- View the signed Data License Agreement.

The left side of the home screen provides some personal information to the DataBait user (Figure 10).



(a)



(b)

Figure 10. DataBait main screen

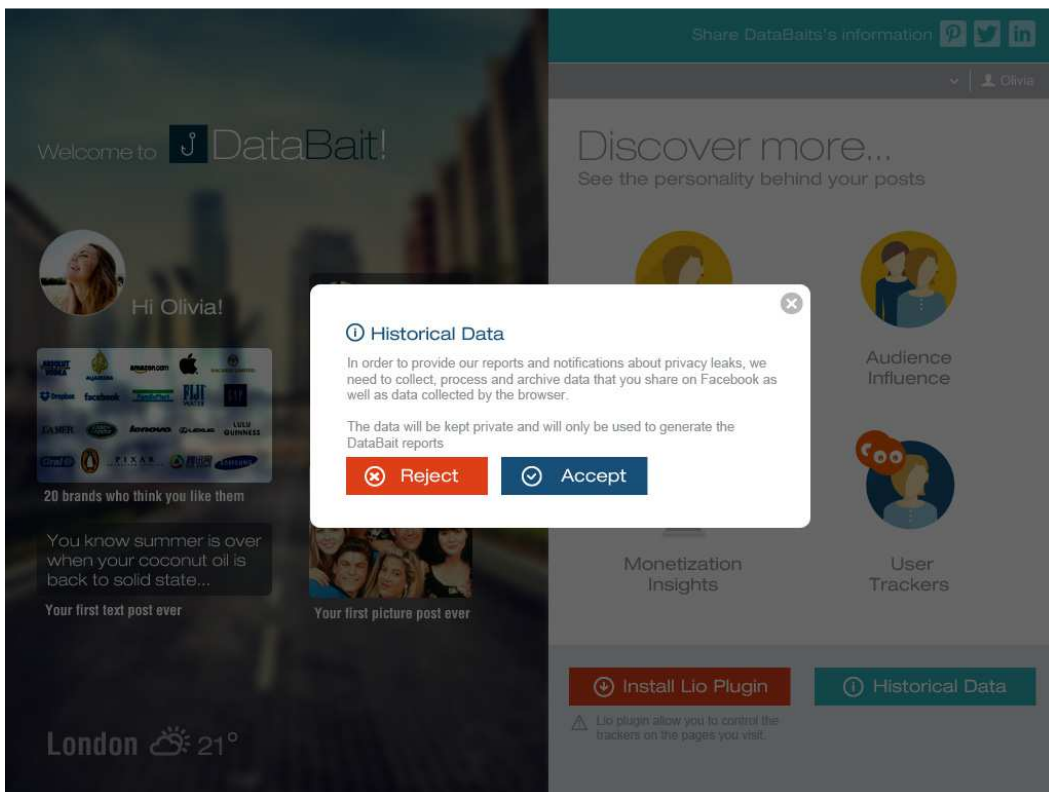


Figure 11. DataBait authentication screen for storage of historical data



Figure 12. DataBait View and Withdraw collected data

When user selects the “My Privacy” screen, the following categories of functionalities are supported, which have been introduced in D7.1 and D2.2:

- **Profile Overview:** A basic overview of the user’s current privacy state. The depicted information is retrieved either from user’s OSN provided information or inferred from user’s behaviour. In the second case the user has the capability to modify or correct the output of the inference.
- **Friends:** Provide information about how our friends affect our privacy profile and specifically how our friends have affected or contributed to the identification of our privacy leaks. In addition, based on the settings of each DataBait user, it may be possible to view the privacy profile of our DataBait friends. At this point should clarify that we are referring only to friends that are also registered with DataBait.
- **Privacy Inferences:** Informs users on potential personal data privacy inferences and compromises. The personal attribute behavioural predictor function takes a set of user activities as input and produces a set of prediction scores about personal attributes that could be inferred by these activities
- **Multimedia:** presents a set of prediction scores about personal attributes depicted in images related to the DataBait user. For instance, such attributes could be smoking, drinking, partying, doing extreme sports, etc.

More information about candidate visualization method of Privacy Inferences and Notification Management are provided in Sections 3.3.1 and 3.3.2. “My Privacy” screen depicts the list of Facebook friends of the corresponding DataBait user that are also members of the USEMP platform (i.e. DataBait users).

The screenshot shows the 'DataBait My Privacy' interface. At the top, there's a teal header with the DataBait logo and social media share buttons. Below the header is a navigation bar with icons for home, profile, friends, and settings. The main content area is divided into sections: 'DataBait Friends' (a vertical list of user avatars), 'My Privacy' (a profile card for Olivia Hayes with details like Date of Birth, Zodiac, Professional, and Home Town), a birthday banner for Friday 12 Sept 2014, and two featured posts under 'Most Liked!' and 'I Really Like This!'. The bottom of the screen shows the account creation date: 'LID ACCOUNT CREATED ON 26 MAY 2014'.

Figure 13 DataBait My Privacy screen

When the user selects the User Trackers from the home screen (Figure 10-a), the screen that opens (Figure 14) provides the capability to identify the USEMP user's trackers, in terms of 3rd party online tracking and analytics services that monitor and collect the end-user's digital trail. In addition this screen enables DataBait users to identify their web browser tracker(s) and thus, define fine-grained do not track (DNT) rules (in a flexible and intuitive manner). The key functionalities of Figure 14 are the following:

- View the trackers for the visited web sites. There is the option to filter the view according to a) the last session, b) the ten (10) most frequent visited sites, c) last day web sites, d) last week's web site, or e) all visited web sites. In addition two types of views are provided, which are analysed further in Section 0:
 - List view
 - Graph View.
- View specific information about each web site and tracker (e.g., last access, number of connected web sites).
- View some aggregate results about the number of visited web sites and associated trackers.
- Block (Unblock) specific trackers that monitor web sites.
- General Settings for the tracker application.

(a)

Type	Title	First access	Last access	Connections
Website	outlook.com	Sept 2, 2014	Sept 11, 2014	6
Website	twitter.com	Sept 3, 2014	Sept 11, 2014	12
Tracker	tracker.facebook.com	Sept 3, 2014	Sept 11, 2014	10
Website	mindbodygreen.com	Sept 4, 2014	Sept 11, 2014	12
Tracker	tracker.akamai.net	Sept 8, 2014	Sept 11, 2014	3
Tracker	tracker.akamai.net	Sept 8, 2014	Sept 11, 2014	1
Website	news247.gr	Sept 8, 2014	Sept 11, 2014	8
Tracker	tracker.gigya.com	Sept 8, 2014	Sept 11, 2014	2

(b)

Figure 14 DataBait User Trackers screen (a) Graph view, (b) List view

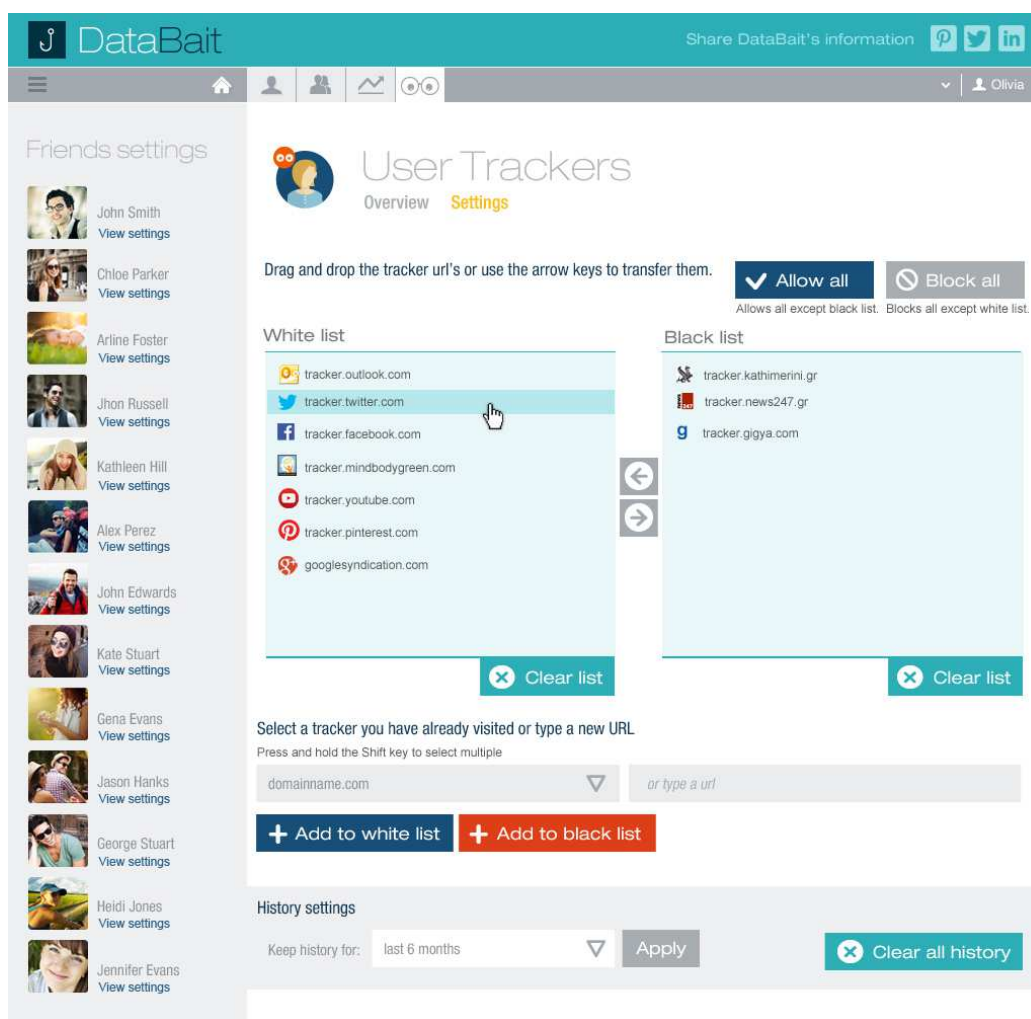


Figure 15 DataBait User Trackers Screen – Settings

When the user selects the Settings option from the Tab Menu of the User Trackers screen (Figure 15), the following functionalities are provided:

- Strategy adoption, selecting between: a) Block all Trackers expect for White List, and b) Allow all Trackers expect for Black List. The default strategy is option b). White List includes those trackers that are allowed to track our online behaviour, while the Black List includes those trackers that are not allowed to monitor end user's behaviour when she visits a web site.
- Transfer identified trackers from the White List to the Black List (and vice versa) and clear the list from all or specific trackers.
- Add a tracker that has been discovered either in the White List or in the Black List that is selected from a drop-list or manually.
- Modify the History size of discovered trackers and the associated web sites.
- Reset Data.

An additional functionality that is provided in this part of the DataBait application is the view of the settings of the trackers of USEMP friends. The list of the all USEMP friends is presented in the left side of Figure 15. After the selection of a specific friend, a pop-up window appears and presents the list of her white-listed and black-listed trackers (Figure 16). From this list, a DataBait user can select one or more trackers and apply similar settings.

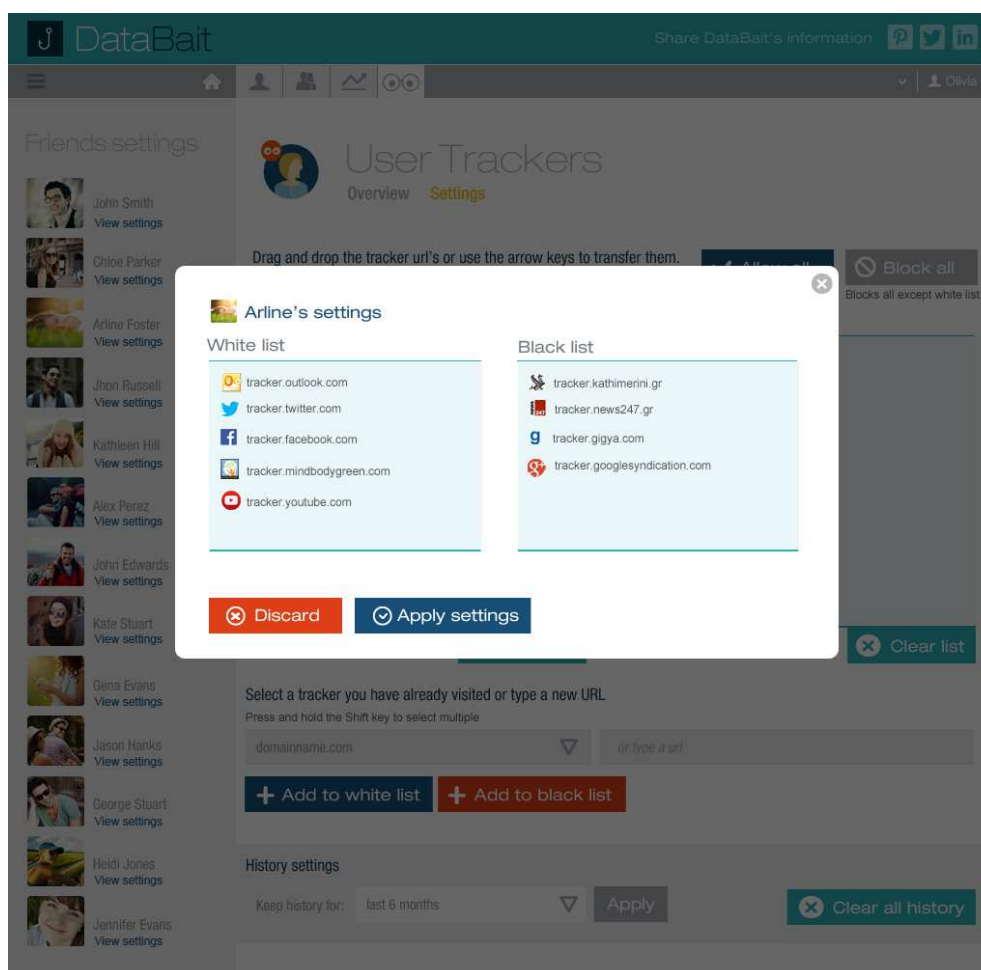


Figure 16 DataBait User Trackers Screen – Get settings of DataBait friends

2.2.6. Web and Mobile Device Technology

Taking into account the increasing number of mobile devices for the development of the DataBait application the Bootstrap framework has been used. Bootstrap is the one of the most popular HTML, CSS, and JS framework for developing responsive and mobile-aware projects on the web. As it is described in previous section “Responsive web design” allows the creation of web sites which adjust and align themselves according to what media tool you use for viewing it (Figure 17).

Bootstrap is a framework for creating websites and web application, which contains HTML and CSS-based design templates for text, forms, buttons, navigation and other components. It also contains optional JavaScript extensions. Bootstrap is open source and available on GitHub. The key features of Bootstrap are the following:

- Mobile-first approach: Since Bootstrap 3, the framework consists of mobile-first styles throughout the entire library.
- Browser support: Bootstrap is supported by all popular browsers.
- Responsive web design: Bootstrap's responsive CSS adjusts to Desktops, Tablets, and Mobile phones.
- Easy to get started: With just the knowledge of HTML and CSS anyone can get started with Bootstrap.

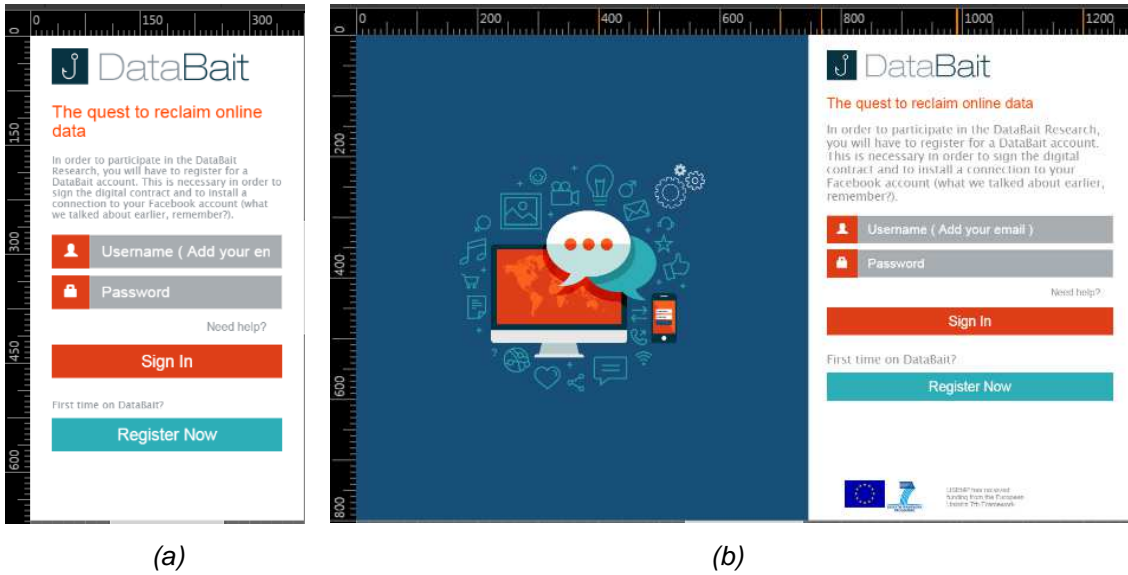


Figure 17 DataBait Responsive design (a) Mobile device screen (b) Tablet screen

Bootstrap supports various desktop or mobile browsers and several platforms, which are presented in Table 2.

Operating System	Chrome	Firefox	IE	Opera	Safari
Android	Yes	Yes	Not available	No	Not available
iOS	Yes	Not available	Not available	No	Yes
Mac OS X	Yes	Yes	Not available	Yes	Yes
Windows	Yes	Yes	Yes ³	Yes	No

Table 2 Bootstrap – Supported Browsers

³ * Bootstrap supports Internet Explorer 8 and higher

2.3. Client side communication with backend

The backend service will provide a set of interfaces in order to manage and setup the backend processing operations as well as user creation and login. Interfaces will use a restful architecture with session tokens for accessing specific user data. Transactions will take place over https to ensure all communications to the backend system remain secure.

2.3.1. Registration Interaction

When a registration request comes into the backend from the frontend, the given credentials are stored provided they do not clash with any other user credential. On creation, an email notification is sent to the provided email address to confirm the account creation process. Upon confirmation of the email, the account is activated; otherwise the account is deleted after the period of 30 days.

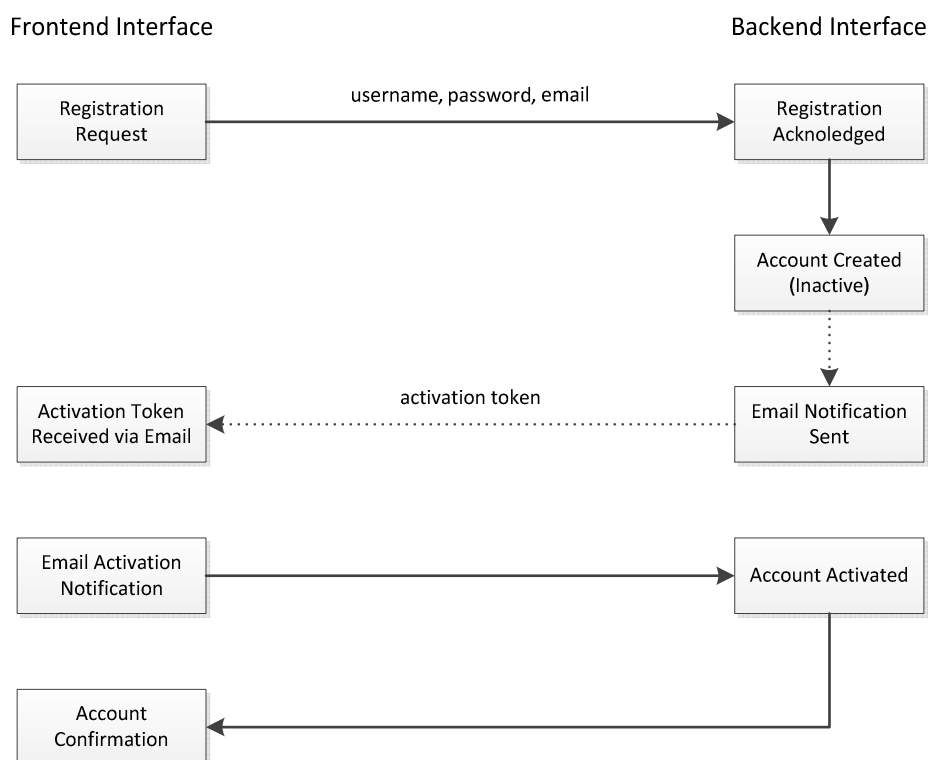


Figure 18 Registration interaction

2.3.2. Login Interaction

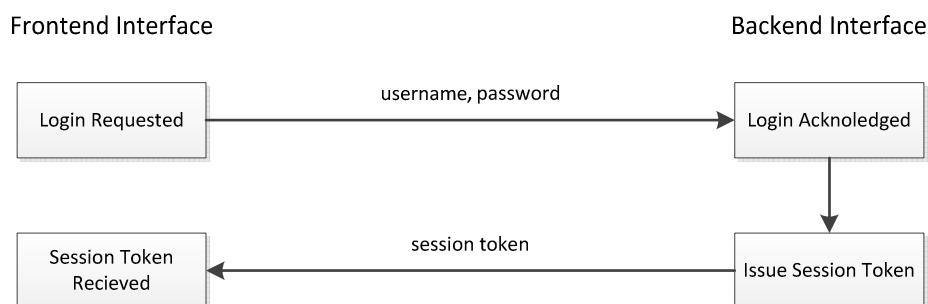


Figure 19 Login interaction

Login requires the backend system to be sent a matching pair of username and password. Once confirmed, the frontend is then provided with a session token, which can then be used for all further communication.

2.3.3. Facebook Login Interaction

Facebook interaction adheres to Facebook's requirement for a user to first login via their system, which provides the frontend with a short lived access token. This short lived access token is then passed to the backend to allow for it to be exchanged for a long term token which the backend can then use to access the user's Facebook resources and start processing their data for monetary value and privacy alerts as per the users requests.

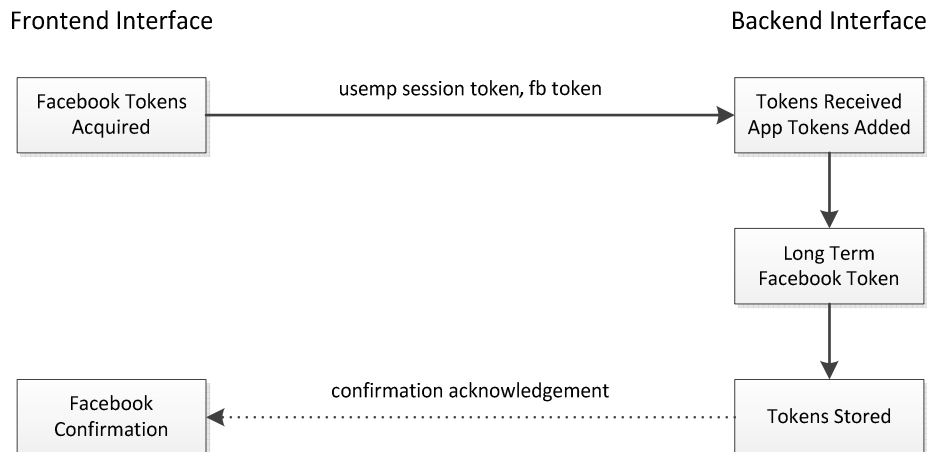


Figure 20 Facebook Login interaction

3. Personal Data Visualisation

The scope of this section is to analyse the state of the art for the visualization of data and introduce the current USEMP work for the visualization focusing on data related to privacy, notification management, and user trackers. In the next WP7 deliverables more information will be provided for the above types of data as well as for data that concern audience influence and monetisation insights.

3.1. USEMP Personal Data and Privacy Profiling

One of the aims of the DataBait platform is to provide a user of the system with the ability to rapidly and efficiently judge the state of their 'privacy profile'. The privacy profile that we refer to in this case will be discussed in D6.1, but for the sake of simplicity it may be considered as a vector of values that correspond to various 'privacy dimensions' that have been deemed to be important to users of online social networks. These privacy dimensions include factors such as the demographic information of the user, their psychological traits or their individual sexual preferences.

The USEMP visualization work will attempt to provide a user with a view of the varying levels of exposure of each of these privacy dimensions in a rapid and efficient way and in a way that does not require a user to understand complex lists of numbers or tables. It is thus the aim of this work to present this information in a tangible way that is simple to understand for the average user.

3.2. SOTA Analysis of Data Visualisation Methods

Data visualization has emerged in recent years as a powerful way of representing otherwise complex data in an engaging and clear way. The visual representation of data aims at exploiting the ability of the human visual system to recognize spatial structure and patterns [8]. This approach is used to provide a clear and comprehensive understanding of a set of data that can then aid subsequent analysis or, as is important in this case, to aid the decision making process for users of social networks when confronted with a visual representation of their privacy profile.

Tufte [11] in his book on the quantitative display of information explains that the use of Graphics can 'reveal' data that was not otherwise evident. He describes some basic rules of visualization that include the rather fundamental rule that the visualization should "show the data" but that they should also induce the viewer to think about the substance of what they are seeing rather than the visualization itself. He also states that the visualization should encourage the eye to compare different pieces of data and, if possible, reveal the data at several levels of detail from a broad overview to fine detail.

Mahoney [6] employed some of these rules in an attempt to visualize university data, including degree awards and the taught modules that led to that award all in an effort to expose the dependencies between different departments in a university. This was thought to potentially aid the decision making process for curriculum planning. Mahoney investigated several kinds of visualization including a networked graph and circular graphs finding that, due to the complexity of the visualisations, that it was necessary to refine the data before any visualization became useful.

One of the classic visualization techniques from the academic literature is the Treemap [5], used to display visual hierarchies of family trees or corporate organizational structures in an efficient way that uses 100% of the available space. This work was later extended by the use of circular partitions that enabled the use of polygons to fill the space [7]. Baudel [1] presented an improvement over these ‘traditional’ visualization techniques using a “Sequential Space-Filling Layout” and a proposed ‘design space’ consisting of five independent dimensions: Order, Size, Score, Recurse and Phrase, with each of the dimensions describing a particular aspect of such layout tasks. Baudel notes that this approach is interesting as it moves beyond encompassing simple grids, tables and trees and also encompasses classic statistics visualizations, such as stacked bar charts, mosaic plots and dimensional stacking.

This notion of a ‘design space’ is important as it reduces the complexity of a visualisation task. Shultz [10] describes his work on building a ‘design space’ for visualization tasks and describes five design dimensions that characterize the main aspects of a visualization task. These dimensions include the ‘goal’, which defines the intent with which the task is pursued, the ‘means’, which determines the method for reaching the goal, the ‘characteristics’, which defines the facets of the task that we wish to reveal, the ‘target’, which defines the part of the data on which the task is carried out and the ‘cardinality’, which determines the number of instances of the chosen target that are to be considered. Shultz argues that these five dimensions may prove to be a useful fragmentation of what might otherwise be a somewhat blurry and overloaded notion of what a visualization task actually is.

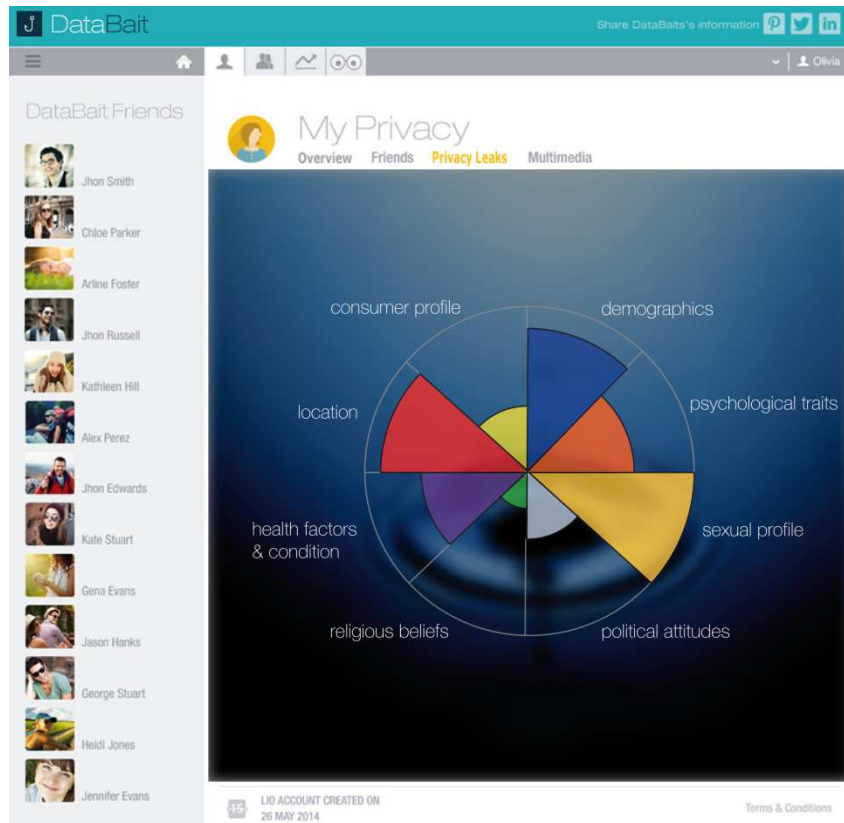
While work in the academic field is wide ranging and abundant, the so-called infographics have emerged in recent years as a popular method for displaying otherwise complicated data in a simple way to massive audiences. While they have long been used for the display of scientific data to aid the interpretation of scientific results [2] they have more recently begun to be used in the classroom [3] or for public outreach [9]. One of the aims of this work is to both learn from the academic work and work which has been conducted on infographics to produce a visual representation of a user’s privacy profile that is both interesting and full of information for the user.

3.3. Towards USEMP Visualisation Methods

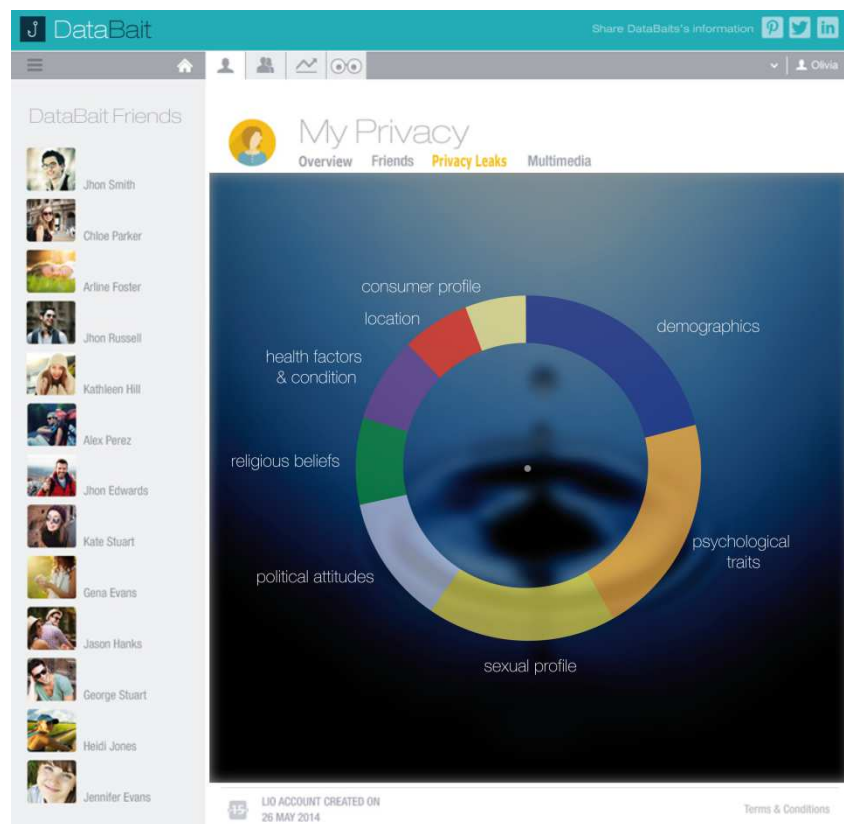
This section presents the USEMP visualization methods for personal data related to privacy, notification management, and user trackers have been introduced. In D7.4 more information will be provided for the above-mentioned types of data as well as for data that concern audience influence and monetisation insights.

3.3.1. Privacy

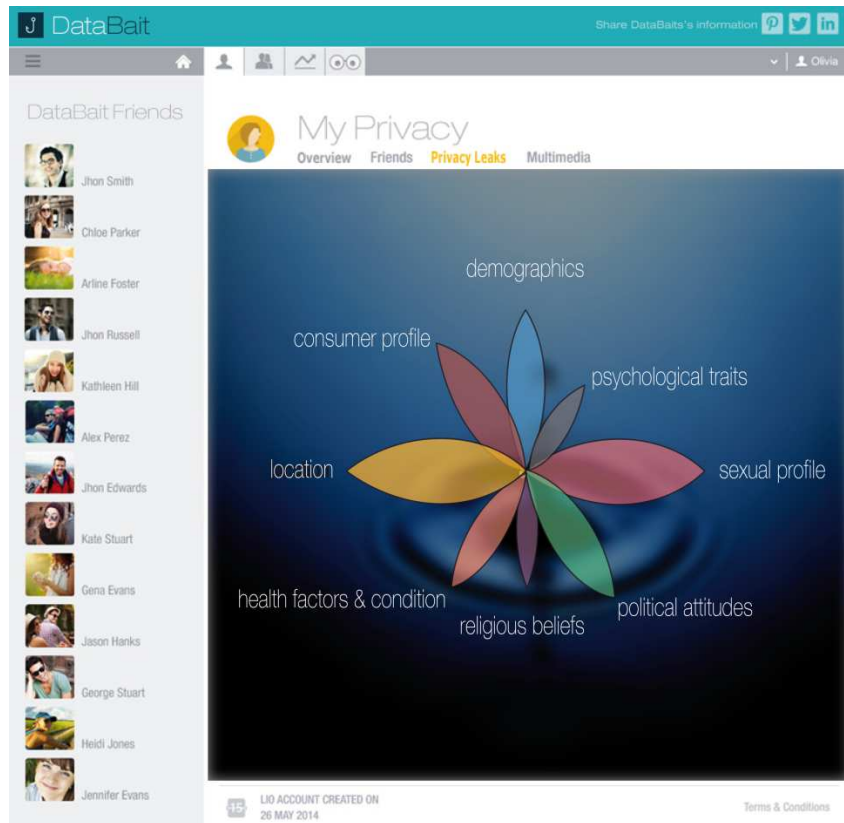
The task of presenting a user with a visualization of their privacy profile is tackled in detail in tasks 4.3 of WP4 and 6.3 of WP6. A number of potential visualisations are generated based on the so-called “privacy vector”, which provides a ‘score’ for each of the privacy dimensions outlined in WP6. This score gives a sense of the exposure of a particular privacy dimension, which can then be further subdivided into more sub-dimensions. As a visualization task, this representation is relatively simple since the number of available dimensions is limited. What we are required to do is to provide an overview to the user of their privacy profile [4], i.e. a visual representation that is simple enough to provide a user with an instant image of the current state of their privacy profile. Four different examples of potential visualisations of this privacy vector are shown in Figure 21.



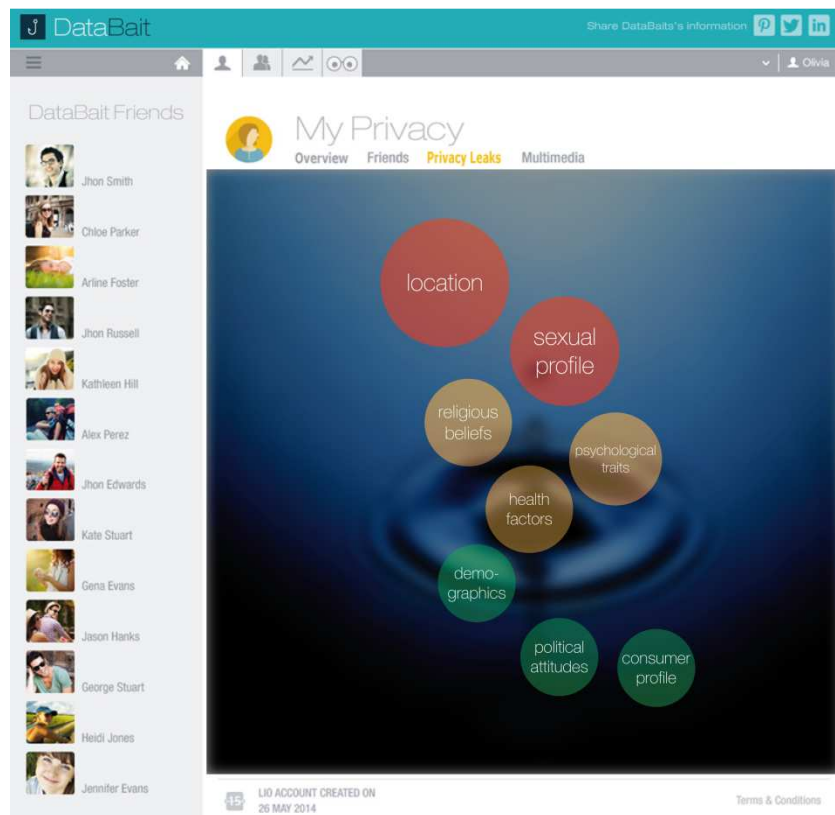
(a)



(b)



(c)



(d)

Figure 21. Four examples of potential visualisations of a user’s privacy profile. The privacy vector is converted to spatial and colour representations.

What we visualize here is the level of the exposure of each of the defined privacy dimensions. This level of exposure is represented by colour or size or spatial area. The final representation that will be used for the DataBait application will be decided in WP4, T4.3 and further refined with feedback from potential users of the system WP6, T6.3. and a pre-pilot living lab study in WP8, T8.2.

3.3.2. Notification Management

As part of the privacy profile visualization work, some emphasis was placed on the production of a system that enables a form of the so called 'future control', a part of D2.2. This future control effectively provides the user of the DataBait system with the ability to see the effect of an action on their privacy profile before that action is committed. For example, as a user types their status update, they might be prompted with a visualization of the effects of certain words on their privacy profile, which will act as a form of notification system to the user.

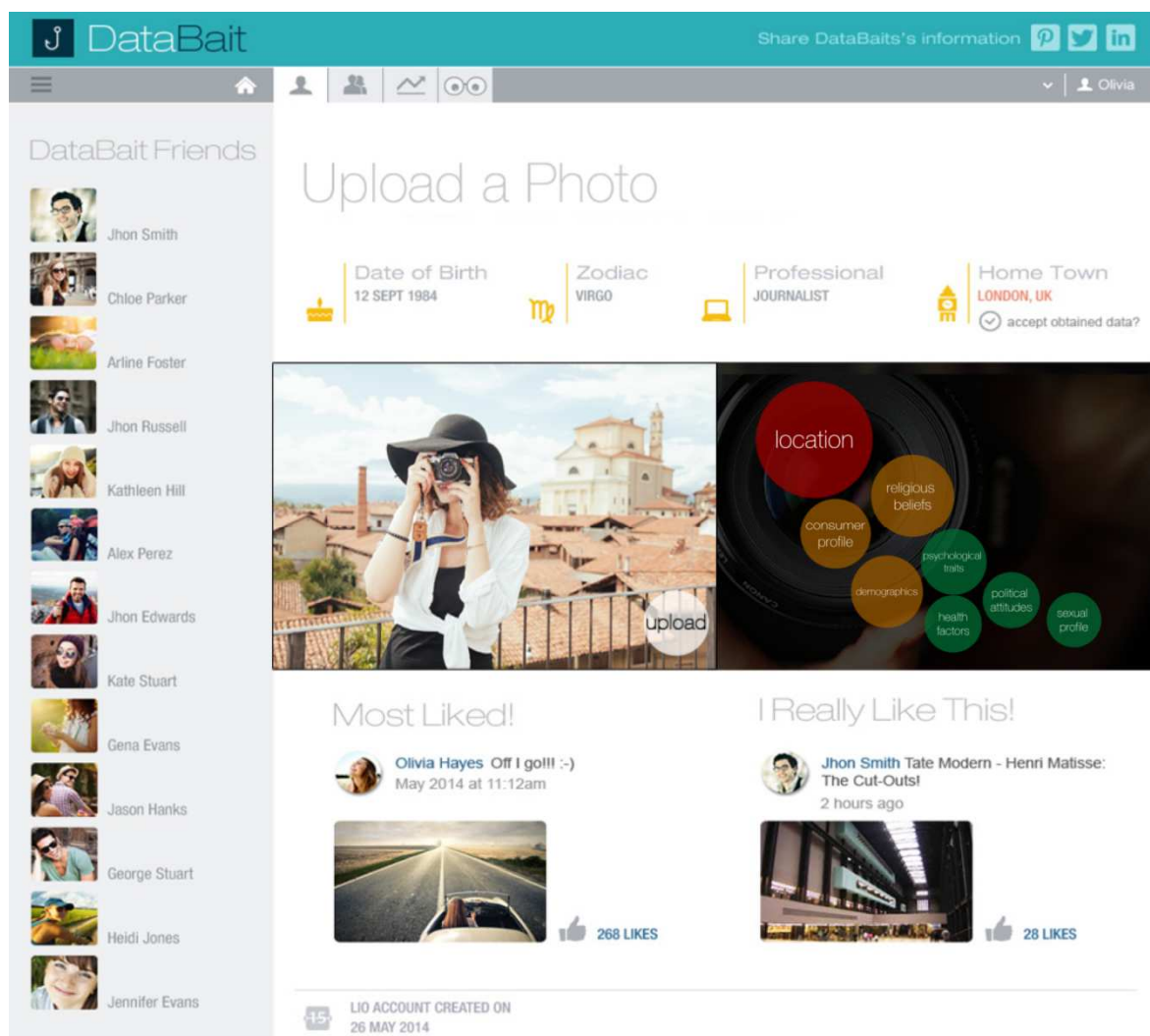


Figure 22. An example of the kind of notification that could be presented to a user. The system analyses their photo before presenting the user with potential privacy issues.

Figure 22 shows a first effort of representing this kind of notification system for the upload of an image. What the user sees is the effect on their privacy profile, which should allow them to decide whether they want to complete the upload of this image or set appropriate privacy

settings to it (e.g. share with only close friends). The picture that a user wants to upload is retrieved using the DataBaitp-Browser plugin or is directly provided to the DataBait web app for processing and analysis by the USEMP system, using the USEMP upload platform.

3.3.3. User Trackers

As it is mentioned above, the USEMP Platform helps the end users to identify potential trackers that monitor end user's behaviour via the web-based services that service providers deliver. The USEMP Web Browser software plugin tracks third party online tracking and analytics services that monitor and collect end-users digital trail on their web browser(s), providing end users the capability to dynamically disable or enable trackers (i.e., define fine-grained "do not track rules"). For the visualization of the information about the identified trackers per web site and the available control capabilities over the trackers, two visualization paradigms have been adopted:

- Graph view
- List view

This approach has been adopted following the design paradigm of the Lightbeam application⁴. The Graph view provides a graphic representation of global schema nodes and the relationships between the nodes (Figure 23). The figure of the node denotes the type of the node i.e., web site or tracker as well as whether the tracker is blocked or not. The graph view is a fast representation approach, which has the capability to provide more information about each tracker e.g., number of associated web sites. However, Graph view might become unreadable and is not considered convenient for a dense graph, which consists of a large number of nodes. In this case, the List view is considered as the most appropriate way to visualize the available information about the detected trackers. List view is a view group that displays a list of scrollable items (Figure 24).

⁴ Lightbeam, <https://www.mozilla.org/en-US/lightbeam/>



Figure 23 User Trackers – Graph view

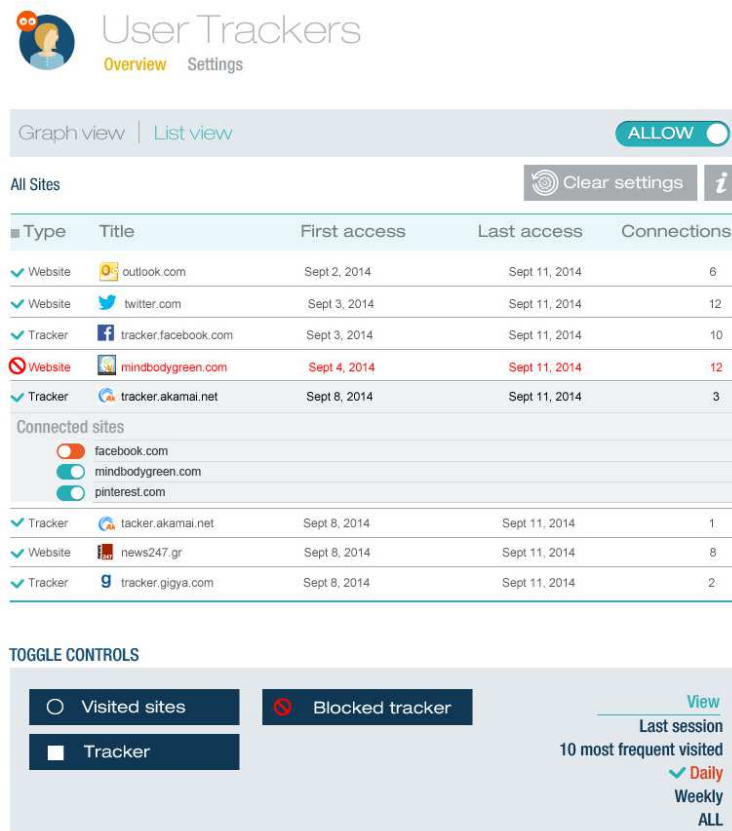


Figure 24 User Trackers – List view

4. Conclusions

In this deliverable the web and mobile interface of the DataBait application has been introduced. DataBait acts as the graphical user interface through which the USEMP system conveys to the end user results of the USEMP system processes and services. Considering the increasing number of mobile devices for the development of the DataBait application the Bootstrap framework has been used, which is one of the most popular approaches for developing responsive and mobile-aware projects on the web, based on HTML, CSS, and JS technologies. Responsive web design model allows the creation of web sites, which adjust and align themselves according to the media tool that is used for viewing it. The development of a web app that can run in multiple platforms (operating systems, web browsers) is one of the main challenges that we have to address.

The requirements and design principles of the USEMP system and specifically of the USEMP client side tools (DATABAITp-GUI, the DATABAITp-BROWSER) have been briefly presented in D7.2. In addition, a number of guidelines for the creation content and different approaches on delivering that content, which is intended to be used by the USEMP users using different means (i.e., smart phones, tablets, personal computers) have been described.

Based on the described functional requirements and the UI/UX design principles, the application map of the DataBait has been introduced and the users' activity flow is presented and mapped into the expected application features. The challenges that were taken into account for the development of the application map are: a) ease of use, b) high efficiency and simplicity, c) low level of complexity for the end users, d) high comprehensibility e) low difficulty to learn and navigate.

Thereinafter, screens that provide the end user the UI to use USEMP functionalities have been presented and analysed. In this deliverable the following functionalities have been taken into consideration: USEMP log-in and registration, USEMP OSN Log-in, user profile, user privacy inferences, user trackers, tracking services/activities control. In Section 2, a set of interfaces has been briefly presented in order to manage and setup backend processing operations addressed in this deliverable as well as user creation and login.

In addition, the state of the art for personal data visualization methods has been provided and the USEMP work for the visualization focusing on personal data related to privacy, notification management, and user trackers has been introduced.

In D7.4 (M24) the UI and visualisation methods regarding audience influence and monetisation insights will be provided. Specifically, monetisation insights will provide to the end user the awareness of economic value of personal information. They will be able to view the estimation of how their data worth from an economic point of view and how they are exploited by OSNs. Also, Audience Influence will enable end-users to manage their audience either explicitly, via allowing the monitoring and collection of their actions/profile by specific stakeholders, or implicitly, via their postings (and social actions in general). This functionality will provide information about how friends affect the online behaviour of each end-user and how relevant a user's profile is with different friends. Finally, the UI of the DataBait Browser Plugin, modifications on existing screens, and specifically in the context of 'My Privacy' area will be reported in D7.4.

References

- [1] Baudel, T. and Broeksema, B. 2012. Capturing the Design Space of Sequential Space-Filling Layouts. *IEEE Transactions on Visualization and Computer Graphics*. 18, 12 (Dec. 2012), 2593–2602.
- [2] Cressey, D. 2014. Infographics: Truth is beauty. *Nature*. 507, 7492 (Mar. 2014), 304–305.
- [3] Davidson, R. 2014. Using Infographics in the Science Classroom: Three Investigations in Which Students Present Their Results in Infographics. *The Science Teacher*. 81, 3 (Mar. 2014), 38.
- [4] Hornbæk, K. and Hertzum, M. 2011. The notion of overview in information visualization. *International Journal of Human-Computer Studies*. 69, 7-8 (Jul. 2011), 509–525.
- [5] Johnson, B. and Shneiderman, B. 1991. Tree-maps: A space-filling approach to the visualization of hierarchical information structures. *Visualization, 1991. Visualization'.* (1991).
- [6] Mahoney, J. 2013. Data visualisation and statistical analysis within the decision making process. University of Lincoln - ePrints.
- [7] Onak, K. and Sidiropoulos, A. 2008. Circular partitions with applications to visualization and embeddings. *Proceedings of the twenty-fourth annual symposium on Computational geometry - SCG '08* (New York, New York, USA, Jun. 2008), 28.
- [8] Robertson, P.K. 1990. A methodology for scientific data visualisation: choosing representations based on a natural scene paradigm. *Proceedings of the First IEEE Conference on Visualization: Visualization '90* (1990), 114–123.
- [9] Saavedra, F. and Lozano, L. 2013. Planetary Science Multimedia: Animated Infographics for Scientific Education and Public Outreach. *Lunar and Planetary Science Conference*. 44, (Mar. 2013), 2961.
- [10] Schulz, H.-J. et al. 2013. A design space of visualization tasks. *IEEE transactions on visualization and computer graphics*. 19, 12 (Dec. 2013), 2366–75.
- [11] Tufte, E. and Graves-Morris, P. 1983. *The visual display of quantitative information*. Graphics Press.