

Prototyping of a Browser-Based Social N-Screen Platform

[Building Up to Boost User Experience]

Begoña Álvarez de la Cruz
Computer Science MSc Thesis
Web & Media Group
Vrije Universiteit, Amsterdam
begona.alvarezd@gmail.com

ABSTRACT

A second screen is a hand-device which is susceptible to provide added value to the TV content consumption. Notube, with their web browser-based second screen application, moved further through this concept, creating an association between the second screen, Web and TV content. Nevertheless, the implementation still lacks completion in order to achieve a full service and its users' satisfaction.

This project shows the development of a social N-Screen prototype based on previous researches and implementations carried out by Notube. This platform is intended to be used by small groups to explore on-demand content. The main goal of the project is consisted on searching and implementing features to the platform in order to offer to the final user an improved user experience. This improvement is led by the features that allow a completion in the user interaction flow with the platform, such as the implementation of a registration and login, the provision of persistence to the user based content and the addition of new functionalities as personal lists and likes/dislikes tracking.

1. INTRODUCTION

1.1 Background

1.1.1 The Multi-Screen World

The human being has become throughout the last years into a multi-screener¹ nation. From the appearance of television in our living rooms, until the incorporation of lighter and portable new devices such as smartphones or tablets, users have been including all these devices in their routines until turning them into everyday objects. Consequently, tablets, smartphones, televisions and computers have become the main group of devices with which an average user consumes most of their media content[28].

Despite each archetype provides a particular motivation and practise to users, an important fact is that screen devices as

¹Multi-screening: use of more than one screen at a time.

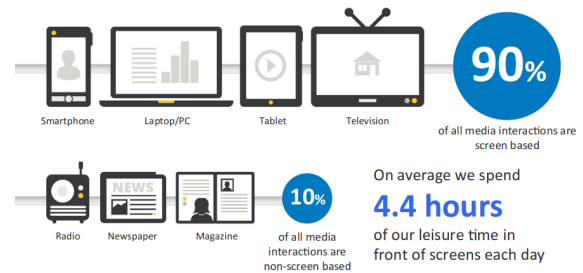


Figure 1: Study of daily media interaction, obtained from [28]

mentioned before are no longer used in isolation but collaboratively. Regarding this device collaboration, two different models of multi-screen[28] behaviour are distinguished: sequential and simultaneous. *Sequential usage* refers to moving through more than one device in order to achieve a task. *Simultaneous* concerns the usage of multiple devices at the same time for either related or unrelated activities. Both consumption forms are increasingly becoming the default mode and is surely influencing the way users engage.

Following this scenario comes the necessity to understand how users interact with these screen devices in combination[14]. The opportunity to decide which device to use, where and how makes possible for users to control their own interaction and content flow.

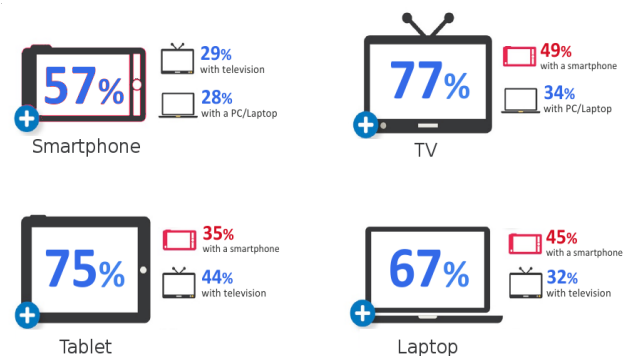


Figure 2: Study of companion devices during simultaneous usage, obtained from [28]

A Google study developed in 2012 [28] exposed how an average consumer makes use of companion devices - smartphone, TV, tablet and laptop - during simultaneous² usage - See Figure 2. There are three main multi-screen combinations:

- Smartphone + TV - 81%
- Smartphone + Laptop/PC - 66%
- Laptop/PC + TV - 66%

Additionally, a research developed by Microsoft in 2013 [17] illustrated how is the user behaviour while multi-screening in simultaneous usage.

- 68% of consumers interact with multiple devices at the same time to access unrelated content; e.g. they may be texting a friend while watching TV.
- 57% of consumers make use of more than one device simultaneously in order to achieve a related activity.

From now on, we will focus our attention in simultaneous usage for *related* activities.

1.1.2 Second Screens

One of our everyday routines that has been altered by this new screen multitasking[35]³ behaviour is that moment while a user watches TV. Viewers no longer focus their entire attention to the TV screen but share it with portable devices.

Figure 3 illustrates how consumers make use of their tablets and smartphones while watching TV. We can observe how indeed users not only surf on the web, but they interact with the device with activities directly related to the program or advertisement that they are watching at that moment. This demonstrates the fact that consumers are not merely interacting with their hand devices as a simple distraction, but sometimes in order to improve their TV content consumption.

This new practice has led to the creation of the new concept *second screen*. Second screen is a hand-device which is susceptible to provide added value to the TV content consumption. These devices such as tablets or smartphones play a role as companion screens that 'connect' viewers to complementary interaction opportunities while they watch TV via applications, additional show-oriented content or in-synch functionalities [12].

This new activity has become such important that a survey developed by Nielsen Holdings N.V. [21] reported 'Using a tablet or smartphone while watching TV is more common than not'. Nearly half of tablet owners - 43%- and smartphone owners - 46% - declared that while watching TV they are making use of their devices as second screen every day. As a consequence of this fact, there are emerging

²Usage for either related or unrelated activities.

³'Human multitasking' is the apparent performance by an individual of handling more than one task at the same time.'

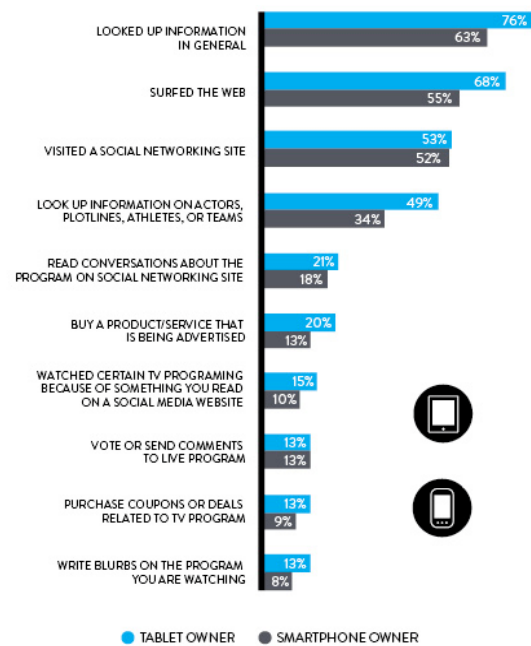


Figure 3: Tablet or smartphone activities while watching TV, obtained from [21]

new apps[32]⁴ which take advantage offering second screen experiences that can improve even more this interactivity while watching TV.

Second screen apps [11] are intended to enable viewers to interact before, during and after the broadcast of a programme using a laptop, smartphone or tablet. The most competent apps, instead of distract, have the potential to increase the viewers' attention and enjoyment on the watching programme. According to [11], eight types are distinguished in order to categorize these apps based on their functionalities:

- Socializing
- Loyalty
- Recommendation
- Transaction
- Information
- Program guides
- Participation
- Creation

On that account, second screen apps arised as a manner to unlock new research and business models due to the wide range of potential possibilities they offer.

⁴'App' is an abbreviation for application. An app is a piece of software. It can run on the Internet, on a computer, on a phone or other electronic device.'

1.1.3 Use case

This project is based on an already existing platform developed by **Notube**[3]. Notube, is a project funded on 2009 that was specialized in second screens and Web merging. Their motivation was getting the Web and TV closer together via shared data models and content across multiple devices. They developed their own N-Screen⁵ prototype, a Web browser-based[10]⁶ second screen application⁷ for small group exploration of on-demand content, both in the same room and remotely, with each individual having their own second screen device. It integrates different combinations of recommendation strategies and allows to decide within a closed group to watch a selected program sharing a 'virtual television', that in their case is the hand device itself.

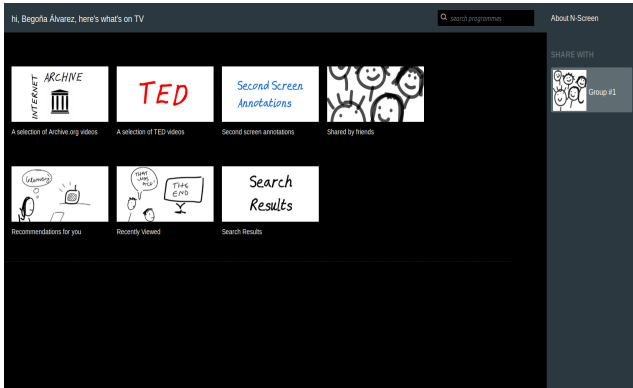


Figure 4: Notube N-Screen screenshot, source code obtained from the public repository <https://github.com/notube/n-screen>

It is developed following a *user-centric approach*⁸ in order to explore main aspects of users' content customisation demands, interaction requirements and entertainment preferences. Their main goal was to investigate if making decisions in collaboration might guide people to find something interesting to watch.

1.2 Goals

The main goal of the project is to re-design, implement and improve the functionality, interactivity and user experience of the browser-based second screen recommender platform[27]⁹ carried out by Notube, in order to provide an attractive functional platform that can be used to graphically test different recommendation strategies.

⁵N-Screen because it might be the primary screen, or one of a bunch of equals.

⁶'Browser-based refers to computer tools and applications which run on a web browser via the Internet without accessing the operating system of any individual computer. These applications are accessed through web pages and can be used by people who are prevented from downloading software applications by firewalls.'

⁷All the source files used to generate this application can be accessed at the public repository: <https://github.com/notube/n-screen>

⁸Placing user's goals at the heart of development.

⁹'A *platform* is a group of technologies that are used as a base upon which other applications, processes or technologies are developed.'

Most of the second screen apps in the market are not successfully developed because they are focused on enhance activities that do not fulfil the real consumer needs[11]. Consequently, the audience engagement fails. This fact is crucial and it must be taken into account in order to make the platform valuable.

In addition, the platform must be design taking into account a possible future recommendation system integration. Furthermore, the project must accomplish certain requirements for a potential release to final users. It needs to be developed offering user-based content, due to its recommendation feature. It is required to provide engaging content and interaction activities, as well as offering an intuitive appearance. Moreover, the platform must be flexible and scalable, so it will not need to be entirely redone with every particular change. Hence, it is needed to reach a complete solution covering all these features, overcoming the gaps and deficiencies that other platforms show.

Here, the main goal is fragmented into more specific sub-goals:

- *Analysis of previous conclusions and results.* Study of already tested aspects. Definition of new tests to provide new data of interest. These data will help in making decisions about the implementation of interaction activities.
- *Requirements definition.* Study of needs and constraints.
- *Software design.* Global design of the new involved software. Debugging and optimization.
- *Platform development.* Implementation and integration in order to achieve the development of the final demo.
- *General purpose tests.* Tests oriented to prove the proper operation and verify the achievement of the requirements and constraints compliance.
- *Results analysis and conclusions.* Achievement evaluation. Study of weaknesses or possible improvements. Definition of further studies.

1.3 Project Organization

The organization of this project is described as follows:

- *Analysis of previous conclusions and results.*

At this first stage, N-Screen-related information and specific knowledge was acquired. An approach to the development tools was also outlined. Goals:

- Second screen state-of-the-art review. Review of the current commercial solutions.
- Adaptation to the development of the platform as well as required tools (repository, client and server side programming languages research and learning, etc...).
- Analysis of related projects results, either completed or under development.

- *Software design.*

This phase covered from the first software definitions and specifications, to the platform implementation until reaching a final demo. Goals:

- Software design and implementation for required features.
- Content dataset migration.
- *Alpha*-version deployment. Source code debugging and improvement.
- *Beta*-version deployment. Source code debugging and improvement.

- *Tests and evaluation.*

At last, the final demo was evaluated and the results were analyzed. Goals:

- Technical Test.
- User experience test.
- Results interpretation and conclusions review. Statement of further studies and future development lines. Found problems evaluation.

- *Documentation generation.*

Dissertation and other required documentation writing.

1.4 Outline

Typically, the body of a paper is organized into a hierarchical structure, with numbered or unnumbered headings for sections, subsections, sub-subsections, and even smaller sections. The command `\section` that precedes this paragraph is part of such a hierarchy.¹⁰ L^AT_EX handles the numbering and placement of these headings for you, when you use the appropriate heading commands around the titles of the headings. If you want a sub-subsection or smaller part to be unnumbered in your output, simply append an asterisk to the command name. Examples of both numbered and unnumbered headings will appear throughout the balance of this sample document.

2. STATE-OF-ART

Typically, the body of a paper is organized into a hierarchical structure, with numbered or unnumbered headings for sections, subsections, sub-subsections, and even smaller sections. The command `\section` that precedes this paragraph is part of such a hierarchy.¹¹ L^AT_EX handles the numbering and placement of these headings for you, when you use the appropriate heading commands around the titles of the headings. If you want a sub-subsection or smaller part to be unnumbered in your output, simply append an asterisk to the command name. Examples of both numbered and unnumbered headings will appear throughout the balance of this sample document.

¹⁰This is the second footnote. It starts a series of three footnotes that add nothing informational, but just give an idea of how footnotes work and look. It is a wordy one, just so you see how a longish one plays out.

¹¹This is the second footnote. It starts a series of three footnotes that add nothing informational, but just give an idea of how footnotes work and look. It is a wordy one, just so you see how a longish one plays out.

3. REVIEW STUDY

A model for the platform is defined in this chapter. Initially, the first version of the N-Screen developed by Notube[3] is evaluated to obtain some guidelines and potential improvements facing our own. The final software implementation requirements and constraints are defined. Finally, a discussion and evaluation over different features is carried out, exposing different options and some final conclusions in order to face a potential deployment to final users.

3.1 Browser-Based Social N-Screen Platform Model

Essentially, the BBSNSP¹² responds to a second screen application based on a recommendation system. However, a further review shows some general differences:

- *Browser-based*: instead of being developed as a mobile or computer app, the BBSNS is a web app. A *web app* refers to software that runs on a web browser. This way, not only the app can be updated and maintained without disturbing potential users by requiring them to re-download. Additionally, it provides implicit support for cross-platform[29]¹³ compatibility.
- *Group decision recommendation system -Social-*. The recommendation feature is focused on 'how collaborating together might help people find something interesting to watch'¹⁴. It is designed to be used within a small group of friends in a collaboratively way to reach together a successful programme to watch.
- *Apart*¹⁵ *group oriented*: this BBSNS is mainly designed to watch together with other friends but being remotely located.
- *N-Screen*: It is not only oriented to be used as a secondary. It is considered that it might be used as a primary screen, secondary screen, or one of a screen devices collection.

3.2 First Browser-Based Social N-Screen Platform Review

The first BBSNSP was a project of Notube developed in 2011. It supposed a big step forward to get the Web and TV closer together using shared data models and content across multiple devices[24]. It is designed to help deciding and enabling to interact using drag and drop over screen devices. It allowed to investigate how helpful is group collaboration in order to find an interesting programme to watch, for limited group of users. It gave the strengths and weaknesses to establish the fundamentals for future developments.

While constituting a successful platform meeting most of its features, the Notube BBSNSP introduced some issues found after its evaluation. This issues are following listed.

¹²Browser-Based Social N-Screen Platform

¹³'*Cross-platform* regards the capability of a software to run identically on different platforms.'

¹⁴<https://notube3.wordpress.com/2011/10/10/n-screen-a-second-screen-application-for-small-group-exploration-of-on-demand-content/>

¹⁵Different physical locations.

I) As a recommender system, the platform is intended to be user based content oriented. User preferences and explicit interaction should provide to the recommendation strategies an on-real-time update to re-rank the displayed personal suggestions[19]. In Notube's platform, user interaction is treated as *volatile* data, each time that the session is closed, the information is lost. This implementation impedes to exploit all this relevant information. Figure6 shows the 5 tasks that are intended to be accomplished by the recommendation engines in the client-side. If the platform lacks tracking the user interaction and preferences, it would not be possible to realize properly the first *training* step due to a shortcoming information.

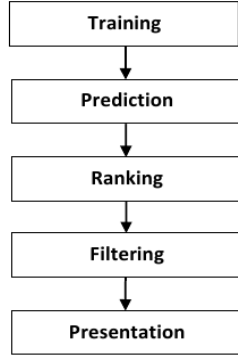


Figure 5: Recommender system tasks, obtained from [19]

II) Their own preliminary findings after the evaluation¹⁶ showed us a weak point. Notube results obtained after testing N-Screen with real users showed how some of them received a successful recommendation and indeed were wishing to watch later that specific selected programme. They received feedback such as "Not for watching something instantly, only for making suggestions for things that could choose to watch later if you wanted to" or "I do not think it is on to recommend things for others to watch instantly". This results revealed an interesting *sequential usage* behaviour. In this case, instead of passing from one device to another, the user is making use of the same device time later to accomplish a task. Therefore, not only the volatile activity of the user, but also a lack of option to save wished programmes for a future moment, makes unlikely the activity to watch *that programme later*.

III) It is very important to offer an attractive set of possible activities for the user to interact with the [26]. The only implicit offered interactions are the browsing and the suggestion part. For this reason, we found that Notube N-Screen slightly incomplete in terms of interactivity actions.

IV) The platform was designed to be not only a recommendation system, but also a place to watch a selected programme. There exist two different ways to realize this action: locally and remotely. The first one is performed when a user decides to watch a programme individually. By contrast, the second one is conducted when a group decides to

watch together a selected programme and the members are located in remote locations. This last visualization mode involves the execution of a browser-based 'virtual TV' in which every person within the group can watch simultaneously the same. Unfortunately, this last feature was not working because the programme visualization part was missing. Making the virtual TV feature working is indeed necessary for a complete success of the platform.

V) As mentioned before in this section, issue I), the platform is designed to provide -future- recommendation strategies. For this purpose, its structure design needs to be flexible and scalable to allow future recommendation scripts compatible in terms of data, in this case, a wide set of programmes. Following a deep study of the source code, we found an important bottleneck. The implementation was not flexible to face data content migrations. The programmes displayed, and all the logic behind it, were designed to correctly work just locally and with a limited flexibility range. It is not possible to afford heavy data as a wide set of programmes running locally due to memory limitations.

VI) The inherit second screen behaviour of the platform makes strongly important to take into account that higher complex interaction in this type of environments is a determinative factor [8], since involves a multi-tasking activity. We found that the homepage was not simple while informative enough in order to not distract the user from the main screen attention. It lacked first sight relevant information, therefore it required more attention than actually needed.

4. IMPLEMENTATION STUDY

Once the first BBSNSP has been reviewed, it is moment to study how to settle every found issue and bottleneck exposed in Section 3.2 and start the implementation. The decisions regarding the implementation of new features are exposed.

4.0.1 Localhost Deployment

Before being able to start programming the platform, it was required preparing the environment.

On the one hand, it was needed to set up a local server to be able to run and test our web application. We used XAMPP on a Linux distribution for this purpose, due to its multiple advantages and intuitiveness. XAMPP is an open source independent server platform which consists mainly of a MySQL database, Apache Web server and required interpreters for scripting languages as PHP and Perl.

On the other hand, the set-up required an Ejabberd installation to manage instant messages in order to enable automatic non-typed communication between remote users and allow them to collaborate -suggest-. Ejabberd is a Jabber/XMPP[38]¹⁷ server [15] which is realized in Erlang¹⁸ language. Its main functionality is that it works as an instant messaging server that allows more than one poeple to

¹⁷ 'Extensible Messaging and Presence Protocol, better known as XMPP (formerly Jabber), is an open and extensible XML-based protocol originally designed for instant messaging.'

¹⁸ Therefore, it was required also the Erlang packages installation for compilation

¹⁶ <https://notube3.wordpress.com/2011/12/12/preliminary-findings-of-n-screen-user-testing/>

communicate and participate in real time, based on typed text or not. Additionally, considering that is intended to be used by a web application as part of a communication tool, it was necessary to enable BOSH in our Ejabberd server configuration. HTTP is synchronous providing simple call and response methods, discordant to XMPP's asynchronous event based protocol. Therefore, BOSH¹⁹ is a protocol that provided us a method to use XMPP in our platform by making http requests with a long time-out.

After this preparation, the environment was therefore ready to start the implementation.

4.1 Registration & Login

The platform is intended to be used to test researches developed about recommendation strategies, therefore, it follows a user content based model. As explained in Section 3.2 issue 1), user preferences and explicit interaction should provide to the recommendation strategies an on-real-time update to re-rank the displayed personal suggestions. In order to deal with the volatile data it was decided to implement a Registration & Login system. This way, it is possible to tack user preferences and provide relevant data to the recommendation side.

An important step before implementing the registration system was to study different types of registration which we could offer. Due to the imminent growth of the social networks popularity such as Facebook²⁰ [25], websites are currently offering two different options to accomplish a registration:

- *Good registration*: known as the one where a user fills in directly the registration form with their personal data.
- *OpenID*: OpenID Registration is an extension to the OpenID[7]²¹ Authentication protocol that allows for very light-weight profile (mostly social networks profile, as Facebook or Twitter²²) exchange . It is designed to pass eight commonly requested pieces of information when an end user goes to register a new account with a web service.

The OpenID provides an even faster registration than before. This '1-click' activity can definitely be very attractive to implement as unique registration for a second screen platform[2], where users distraction has a crucial importance. But despite its significance through the last years, not every user is willing to share their social networks' personal profile due to a lack of confidence in such websites[23]. For this reason, we decided to include both in our platform, selecting the OpenID registration provided by Facebook, since this social network can offer valuable information for the recommendation system.

¹⁹Bidirectional-streams Over Synchronous HTTP

²⁰<https://www.facebook.com/>

²¹'OpenID is an interoperable authentication protocol designed to be safe, faster and easier way to log in to web sites.'

²²<https://www.twitter.com/>

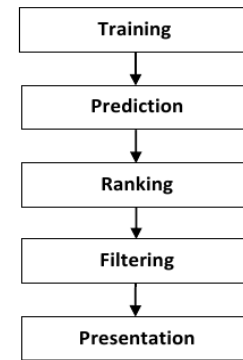


Figure 6: Recommender system tasks, obtained from [19]

The implementation for registration & login has been carried out making use of:

- *MySQL database*
- *Database handler scripts*:
 - HTML5 submit forms & PHP for the good registration
 - Javascript Facebook SDK[30]²³ & PHP for the OpenID registration

4.1.1 Database

The structure of our database has been changing as the platform evolved. At the end, intending to match every needed requirement, we reached the following schema based on two tables:

Table **members**. Column fields:

- **member_id**: Primary key²⁴
- **firstname**: First Name
- **lastname**: Last Name
- **login**: Unique user-name
- **passwd**: MD5 encrypted[37]²⁵
- **facebook_id**: unique ID extracted from Facebook OpenID.

Table **content**. Column fields:

²³'*Software Development Kit* (SDK) refers to a programming package that enables a programmer to develop applications for a specific platform.'

²⁴The *primary key* of a relational table uniquely identifies each record in the table.

²⁵'The *MD5* message-digest algorithm is a widely used cryptographic hash function producing a 128-bit (16-byte) hash value, typically expressed in text format as a 32 digit hexadecimal number.'

- **member_id**: Primary key²⁶
- **recommendations**: Personal recommendations list, JSON²⁷ format.
- **recently_viewed**: Recently viewed or watched programmes personal list, JSON format.
- **watch_later**: Watch Later personal list, JSON format.
- **like_dislike**: List containing personal likes and dislikes of the user, JSON format.
- **shared_by_friends**: List containing suggestions to the user made by members within his/her same group.

For further details in terms of database fields definition, the following code (`mysql.sql`) shows in SQL language our tables initialization:

```
CREATE TABLE IF NOT EXISTS 'members' (
  'member_id' int(11)
    unsigned NOT NULL AUTO_INCREMENT,
  'firstname' varchar(100) DEFAULT NULL,
  'lastname' varchar(100) DEFAULT NULL,
  'login' varchar(100) NOT NULL DEFAULT '',
  'passwd' varchar(32) NOT NULL DEFAULT '',
  'facebook_id' bigint(11) DEFAULT NULL,
  PRIMARY KEY ('member_id')
) ENGINE=MyISAM
  DEFAULT CHARSET=latin1 AUTO_INCREMENT=
  1 ;

CREATE TABLE IF NOT EXISTS 'content' (
  'member_id' int(11)
    NOT NULL AUTO_INCREMENT,
  'recommendations' longtext NOT NULL,
  'recently_viewed' longtext NOT NULL,
  'watch_later' longtext NOT NULL,
  'like_dislike' longtext NOT NULL,
  'shared_by_friends' longtext NOT NULL,
  PRIMARY KEY ('member_id')
) ENGINE=InnoDB
  DEFAULT CHARSET=latin1 AUTO_INCREMENT=
  1 ;
```

Picture whatever shows how the content is initialize for each user the very first moment during the registration.

****INCLUDE AN IMAGE OF THE DATABASE STRUCTURE BEING POSTED BY REGISTRATION****

4.1.2 Login Security

An important implementation to mention is that, in order to access to the personal webpage after the login action, a security checking has been implemented. The access to the welcome page containing personal user-based content is only possible if the login has been correctly done, case

²⁶The *primary key* of a relational table uniquely identifies each record in the table.

²⁷'JSON' (JavaScript Object Notation) is a lightweight format that is used for data interchanging.'

in which the authentication `SESSION VARIABLES[1]`²⁸ are set and authenticated.

The following script (`auth.php`) shows this implementation:²⁹

```
<?php
//Start session
session_start();

//Check whether the session variable
//SESS_MEMBER_ID is present or not
if(!isset($_SESSION['SESS_MEMBER_ID'])
    || (trim($_SESSION['SESS_MEMBER_ID'])
        == '')){
    header("location:index.html");
    exit();
}
?>
```

In addition, for localhost testing purposes, it has been included a management of the browser cookies³⁴³⁰ to allow more than one sessions running simultaneously in the same browser.

4.2 Interaction Activities

4.2.1 Watch Later

It is severely important to understand how is the user behaviour when interacting with a web service. Developers and designers work together to deploy a final product, but sometimes users can present unpredictable conduct and wishes^[22]. It is only after realizing tests on final users when the success or failure can be declared. Notube N-Screen, for example, was designed to be a second screen recommendation platform to watch immediately a selected programme. But after a evaluation with real users they found out how some users where wishing to watch that successful recommended programme not immediately, but in another later moment. This result revealed a very interesting *sequential usage* behaviour in which users utilize the same platform -and sometimes the same device- but in separated moments to accomplish a task, in this case, watch the desired programme. The platform therefore showed a general successful feedback, but was lacking an important user conduct that was not taking into account at the beginning.

Our simplest and effective solution to this situation^[16] has been the implementation of the possibility to add programmes to a personal '*watch later*' list. This list is directly populated with the implicit user interaction of clicking a '*watch later*' button. The list can be edited by the addition or removal of a programme. It is important to mention that each time the user edits this personal list, it is directly updated in the database making use of the PHP script `set_channel.php`, which source code is as follows:

²⁸'A *Session Variable* is an associative array containing session variables available to the current script.'

²⁹`SESS_MEMBER_ID` is only initialized when the login authentication had been successful.

³⁰'A *cookie* is a small piece of data sent from a website and stored in a user's web browser while the user is browsing that website.'

```

<?php
//Start session
session_start();
$data = mysql_escape_string($_POST['data',
]);
$channel = $_POST['channel'];
ini_set( 'default_charset', 'UTF-8' );

//Include database connection details
require_once('config.php');

//Connect to mysql server
$link = mysql_connect(DB_HOST, DB_USER,
DB_PASSWORD);
if(!$link) {
die('Failed to connect to server:'.
mysql_error());
}

//Selecting database for the user
$db = mysql_select_db(DB_DATABASE);
if(!$db) {
die('Unable to select database');
}
$member_id = $_SESSION['SESS_MEMBER_ID'];

mysql_query('UPDATE content SET $channel
=
'$data' WHERE member_id = '$member_id
''');
?>

```

****INCLUDE SCREENSHOT WATCH LATER ****

4.2.2 Like & Dislike

As mentioned in Section 3.2 issue III), we analysed Notube N-Screen and we found that the platform was missing interaction activities. The implementation of the Watch Later list explained in Section 4.2.1 is actually an addition that improves this issue. Nevertheless, after a deep study we came to the result that just adding that feature was not enough. Based on the recommendation system that is going to be included afterwards, we converged to an idea to include the possibility to elicit user preferences by ranking a programme with a simple Like or Dislike action.

The idea came from the project developed by Vista-TV Sibyl³¹. Sibyl is a TV and radio programme recommender system designed for tablets and personal computers which uses a novel drag-and-drop system to extract user preferences. A user is able to express preferences by dragging individual programmes into 'like' and 'dislike' boxes. These preferences are immediately used by the client-side recommender to re-rank the programmes and refresh the recommendation list. Therefore, the implementation in N-Screen not only improves the user interaction, but also can be used to improve the recommendation strategy using directly relevant information provided by the user.

As well as with the case of the Watch Later list, likes and dis-

³¹<http://sibyl.prototyping.bbc.co.uk/>

likes are directly populated with the implicit user interaction of clicking a button; simultaneously updating the database with PHP script `set_channel.php`.

****PONER IMAGEN DE LIKES AND DISLIKES****

4.2.3 Hyperlink Metadata

In order to keep improving the platform without distracting the user from the main purpose[2] that is watch a programme, we decided to implement another more additional feature: Hyperlink[36]³² Metadata³³. Inside each selected programme description, we included the possibility to click hyperlink metadata to browse further information of this selected concept. The first idea was to add the possibility to include clickable actions to actor, director, genre and general metadata of a TV programme. Unfortunately, it has been only possible to implement the hyperlink with 'tags' due to content dataset restrictions - data explanation in Section 4.4-. For that reason it has been possible only to display only a small part of the goal, but the implementation is developed to work successfully with other datasets that provide wider content extraction.

****PONER IMAGEN DE POSIBILIDAD DE CLICKAR METADATA****

4.2.4 Random Selection

4.3 Remote TV

Social activities through the Web are becoming exponentially popular since the web encourages users to participate socially active without even moving from their rooms[24]. On the other hand, TV remains a largely passive experience that usually requires physical presence to become social in terms of simultaneous experience sharing, i.e. if you want to watch the same programme at the same time with a friend, you usually meet together in order to do it. For this reason, one of the most innovative and attractive features included in Notube N-Screen was the possibility to share a cross-platform 'virtual TV' in which every person within a group can watch simultaneously the same. Watching TV with faraway friends through a virtual living room.

Unfortunately, as explained in 3.2 issue IV), this last feature was not working because the programme visualization part was missing. We decided that it was necessary to resuscitate the virtual television in order to achieve a successful platform. After long time studying the source code `player.html`, the bug was found. The problem was mainly residing in data compatibility and how `player.html` handled this data to enable a visualization. Further details concerning data structure are explained in section 4.4. The following `player.html` section of code shows how we handled the issue, replacing the local '*manifest*'³⁴ extraction for a `http-request` in order to retrieve the manifest, and therefore, fixing the problem.

³² '*Hyperlink*' is a reference to data that the reader can directly follow either by clicking or by hovering or that is followed automatically.'

³³ *Metadata*: data about data. It is descriptive information about a particular data set, object, or resource.

³⁴ The *manifest* provides relevant metadata for a specific programme such as video-url and video-format

```

$(document).bind('tv_changed', function (e
    ,item) {
    console.log(item);
    var programme = item.nowp;
    var id = item.nowp.id;
    me.nowp = item.nowp;
    $("#title").html(programme["title"]);

    var action = "Play";
    if(programme && programme["action"]){
        action = programme["action"];
    }
    show_message(action+'ing'+programme["
        title"]);

    if(action=="Play"){

$.ajax({
    url: "get_tedtalks_by_id.php",
    type: "POST",
    async: false,
    data: {id: id},
    dataType: "json",
    success: function (data) {
        item = changeData(data);
        //JSON with suggestions format
        var manifest = item.suggestions[0].
            manifest;
        process_manifest(manifest,programme)
        ;
    }
});
// ---- PREVIOUS CODE ----
//pretty much everything should have a
    manifest
    // var manifest = programme["manifest"
    // ];
    // var manifest = item.manifest;
    // if(manifest){
    //     console.log("manifest is "+
    //         manifest);

    //     $.ajax({
    //         url: manifest,
    //         dataType: "json",
    //         success: function(data){
    //             process_manifest(data,
    //                 programme);
    //         }
    //     });
    // }else{
    //     alert('no manifest');
    // }
});

```

*****PONER IMAGENES QUE DEMUESTRAN EL FUNCIONAMIENTO DE LA REMOTE****

4.4 Content Data

This project presents a functional platform that can be used, between others, to test future recommendation strategies.

One of the possibilities that are currently being researched in our Web & Media Department ³⁵ is a recommendation system based on BBC[33]³⁶ programmes through the webpage <http://www.bbc.co.uk/programmes/>. As mentioned in section 3.2 issue V), the Notube N-Screen structure was designed to work with locally stored data. This approach was inefficient since facing future data migrations or including heavier data would follow not only memory but also functional issues. Consequently, we decided to change how the platform deals with this. We converted it to a platform structure designed to deal with remotely stored data based on programmes' IDs including metadata extraction. In order to make this happen, a set of scripts to handle **http-requests**³⁷ has been implemented. These scripts have been programmed making use of Javascript[6]³⁸ AJAX in the client side, and PHP in the server side. This way, it is not needed to locally store testing datasets but to provide to the scripts a suitable URL to extract information. As a result, the functionality speed is maintained at the same time the platform total weight is severely reduced, since it does not locally store any dataset.

In addition, it has been taking into account the possible data structure that the recommendation system may use as input. For this purpose, a deep study of data structure compatibility has been conducted to facilitate as much a possible future implementations carried out by recommendation engines researchers. Based on various data extraction APIs³⁹ such as the *BBC Developers API*⁴⁰ or the *TED Talks Lab API*⁴¹ we decided to set up the following structure⁴² in JSON format for every object contained in our N-Screen, as programmes or videos:

```

{
  "pid":1000,
  "title":"Gero Miesenboeck: Re-engineering
    the brain",
  "description":"In the quest to map the
    brain, many scientists have attempted
    the incredibly daunting task of
    recording the activity of each neuron.
    Gero Miesenboeck works backward --
    manipulating specific neurons to
    figure out exactly what they do,
    through a series of stunning
    experiments that reengineer the way
    fruit flies perceive light.",
  "date_time":"2010-11-03 22:44:00",
  "url":"http://download.ted.com/talks/

```

³⁵<http://wm.cs.vu.nl/>

³⁶ 'British Broadcasting Corporation (BBC) is a UK-based international public-service broadcaster head-quartered at Broadcasting House in London.'

³⁷ *HTTP request/response* protocol, which means a client-side application sends a request for some file, and the web server sends back a response.

³⁸ 'Asynchronous JavaScript and XML (AJAX) is the method of exchanging data with a server, and updating parts of a web page without reloading the entire page.'

³⁹ *Application-Programming Interface*

⁴⁰<https://developer.bbc.co.uk/>

⁴¹http://developer.ted.com/API_Docs

⁴² Notice that it is a random example using a extracted TED Talks Video.

```

    GeroMiesenboeck_2010G-950k.mp4",
    "video": "http://download.ted.com/talks/
    GeroMiesenboeck_2010G-950k.mp4",
    "speaker": [
    {
    "speaker": {
        "id": 741,
        "title": "",
        "firstname": "Gero",
        "middleinitial": "",
        "lastname": "Miesenboeck",
        "description": "Optogeneticist",
        "whotheyare": "Using light and a little
        genetic engineering -- optogenetics
        -- Gero Miesenboeck has developed
        a way to control how living nerve
        cells work, and advanced
        understanding of how the brain
        controls behavior.",
        "whylisten": "<p>Gero Miesenboeck is
        pioneering the field of
        optogenetics: genetically modifying
        nerve cells to respond to light.
        By flashing light at a modified
        neuron in a living nervous system,
        Miesenboeck and his collaborators
        can mimic a brain impulse -- and
        then study what happens next.
        Optogenetics will allow ever more
        precise experiments on living
        brains, allowing us to gather
        better evidence on how electrical
        impulses on tissue translate into
        actual behavior and thoughts...</p>
        >",
        "slug": "gero_miesenboeck",
        "published_at": "2010-06-09 08:14:00",
        "updated_at": "2010-11-04 15:11:51"
    }
    ],
    "image": "http://images.ted.com/images/ted/
    51f652b9ff6854867d1d7abb2683caf1d8dd22
    fb_240x180.jpg",
    "manifest": {
    "pid": 1000,
    "id": 1000,
    "title": "Gero Miesenboeck: Re-engineering
    the brain",
    "image": "http://images.ted.com/images/ted/
    51f652b9ff6854867d1d7abb2683caf1d8dd22
    fb_240x180.jpg",
    "provider": "ted",
    "duration": 1750,
    "media": {
    "mp4": {
        "uri": "http://download.ted.com/talks/
        GeroMiesenboeck_2010G-950k.mp4",
        "is_live": "false"
    }
    },
    "type": "video/mp4"
    },

```

```

    "tags": {
    "biology": "biology",
    "brain": "brain",
    "neurology": "neurology",
    "science": "science"
    },
    },

```

In order to parse every possible response provided by the API selected to extract videos and programmes collection, it has been implemented a JavaScript function to set up a total compatibility in our platform. This script is named `changeData(data)` which source code is as follows:

```

//Adapt any http request to our own data
format

function changeData(data){

    var random_ted = {
        suggestions: []
    };

    if(data.talks == null){
        return random_ted;
    }

    for(var i = 0; i < data.talks.length; i
    ++){ var item = data.talks[i];
        for(var j = 0; j < data.talks[i].
        talk.photo_urls.length; j++){
            if(data.talks[i].talk.photo_urls[j
            ].size == "240x180"){
                var image = data.talks[i].talk.
                photo_urls[j].url;
            }
        }

        if(item.talk.media_profile_uris["
        internal"]){

            random_ted.suggestions.push({
                "pid" : item.talk.id,
                "title" : item.talk.name,
                "description" : item.talk.
                description,
                "date_time" : item.talk.
                published_at,
                // "media_profile_uris" : item.
                talk.media_profile_uris,
                "url" : item.talk.
                media_profile_uris["internal
                "]["950k"].uri, //TODO
                CHANGE THIS
                "video" : item.talk.
                media_profile_uris["internal
                "]["950k"].uri,
                "speaker" : item.talk.speakers,
                "image" : image,
                "manifest" : {
                    "pid" : item.talk.id,
                    "id" : item.talk.id,
                    "title" : item.talk.name,

```

```

        "image" : image,
        "provider" : "ted",
        "duration" : 1750,
        "media": {
            "mp4": {
                // "type": "video/x-swf"
                ,
                "uri": item.talk.
                    media_profile_uris["
                    internal"]["950k"].
                    uri,
                "is_live": "false"
            }
        },
        "type": "video/mp4"
    },
    "tags" : item.talk.tags
});

}
else{

random_ted.suggestions.push({
    "pid" : item.talk.id,
    "title" : item.talk.name,
    "description" : item.talk.
        description,
    "date_time" : item.talk.
        published_at,
    // "media_profile_uris" : item.
        talk.media_profile_uris,
    "url" : "", //TODO CHANGE THIS
    "video" : "",
    "speaker" : item.talk.speakers,
    "image" : image,
    "manifest" : {
        "pid" : item.talk.id,
        "id" : item.talk.id,
        "title" : item.talk.name,
        "image" : image,
        "provider" : "ted",
        "duration" : 1750,
        "media": {
            "mp4": {
                // "type": "video/x-swf"
                ,
                "uri": "",
                "is_live": "false"
            }
        }
    },
    "type": "video/mp4"
},
    "tags" : item.talk.tags
});

}

}return random_ted;
}

```

It is important to mention that in order to introduce an attractive platform to interact with, a complete data mi-

gration has been implemented. Due to its advantages and popularity we decided to implement the final demo making use of videos provided by TED Talks API[4].

4.5 Web Design

While building a web platform, it is important to design how to improve its functionality and also its graphical interface. A wide set of studies [2] show how a not only well-built website, but also well-designed can increase its user traffic, due to an improved user enjoyment. Nowadays a aesthetic design can be even more influential in affecting user preferences than traditional operational usability, since its effect seems to transcend the product and influence other judgements, known as the *halo effect*[9][31]⁴³. Concerning the halo effect, several studies [9] presented a correlation between the aesthetic factor of a graphical interface, its perceived usability and the final user satisfaction with that interactive system.

For this reason, an extensive part of this project has been focused in web design research and implementation in order to reach an improved user experience.

Through previous subsections included in our Implementation Study, a list of solutions for *functionality* issues have been exposed. In this subsection, we are going to introduce our implementations concerning *design* issues.

4.5.1 Home Page

Since the beginning of web pages, human-computer interaction has been an important factor concerning experience evaluations[18]. Usability considerations played an important role and influence the way interactive systems are designed and developed[13]. But determining user satisfaction, there have been a fluctuation from functional fulfilment to how is the provided global experience[9]. How intuitive an interface is and how it looks is influencing the capability to engage users more than ever[2]. Now users are demanding functionality, usability and aesthetics in order to generate affective responses.

For this reason, it is severely important trying to understand who are going to be the final users of an interactive service, where is it going to be used and how. Following this, we converge to a main aspect that the homepage demanded to accomplish due to its inherit second screen conduct: It needed to be easy to use in order to not distract the user from the main viewing experience, in our case, mostly in front of a TV; and at the same time meet aesthetic design to boost users engagement. We found that the homepage was lacking first sight relevant information, therefore it required more attention than actually needed. Furthermore, its aesthetics could be improved. Figure 7 shows a screenshot of Notube N-Screen homepage. This homepage was showing, during first sight view, the set of different channels 'Recommendations for you', 'Recently Viewed', 'Search Results' and 'Shared by friends', but without providing any further details about them.

⁴³The *halo effect* is a cognitive bias in which an observer's overall impression of a person, company, brand, or product influences the observer's feelings and thoughts about that entity's character or properties'

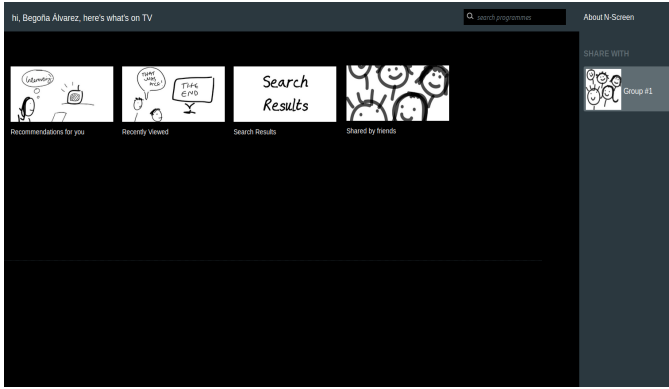


Figure 7: Notube N-Screen Homepage

Consequently, we decided to display the homepage providing further information meanwhile trying to not disturb the user[5][16][22]. We opted following the homepage design of <http://nscreen.notu.be/ted/> due to its simplicity and at the same time its further information provided [20]. Figure 8 shows the result of our implemented homepage.

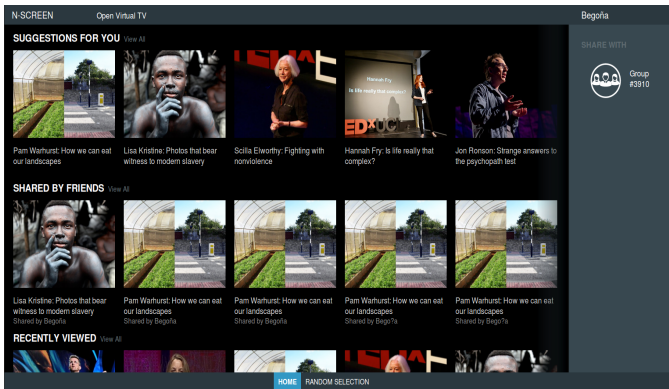


Figure 8: New N-Screen Homepage

Concerning channels replacement, the new design shows for every channel as 'Suggestions for you' or 'Shared by friends' a first sight view of five programmes -or less if the screen has a smaller size -. Furthermore, it has been added the possibility to view all the programmes contained with the addition of a clickable 'View All' next to the channel title. Figure 9 shows this addition and Figure 10 shows the result after clicking 'View All' for one of the channels.

Every channel contained in the new N-Screen homepage - Suggestions for You, Shared By Friends, Recently Viewed, Watch Later, Likes and Dislikes - present the same design approach.

In addition, it has been implemented more design improvements concerning the homepage aesthetics. Most of the websites are currently showing information related to the user profile in the header at the right side[5][16][22]. For this reason, the name of the user has been moved in the header from the left to the right side in order to maintain its placement customary. Moreover, the images places next to group number and members of the same group been redesigned,

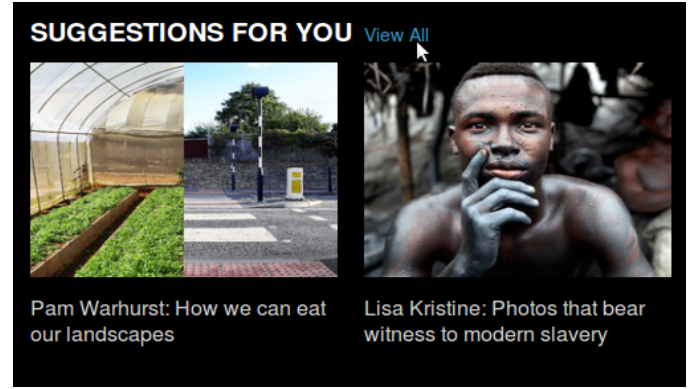


Figure 9: View All Addition next to a Channel Title, New N-Screen Homepage

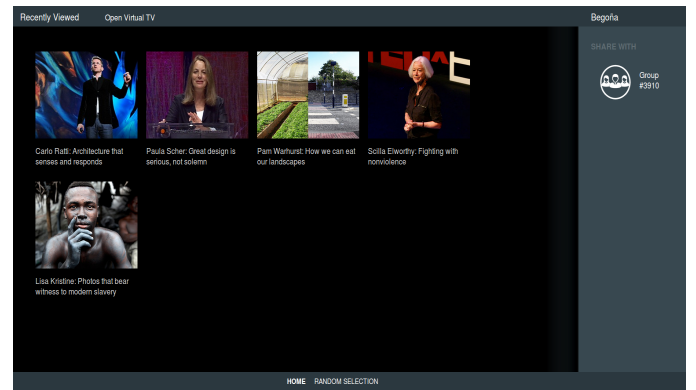


Figure 10: Programmes Visualization after 'View All' click, New N-Screen

see Figure 11.



Figure 11: Images Redesign in Vertical-Right Bar, New N-Screen

4.5.2 Selected Programme

One of the most tedious parts during the design has been re-designing the visualization of a selected programme. During this phase, aesthetics and usability have been the most important factors. It has been included a vertical-right interactive bar next to the programme default image, see Figure 12. This bar contains a set of five icons: Watch Later, Like, Dislike, Shared by Friends and Recently Viewed. It is important to mention that three of this icons -Watch Later, Like and Dislike- are call-to-action icons, which means that if the user clicks on one of them, the programme will be automatically added to the personal list which name is that precise icon. It is important to mention that if a programme

is contained in one of the personal lists, the icon will be automatically rendered to be displayed in a different colour, as a reminder for the user, see Figure 13.

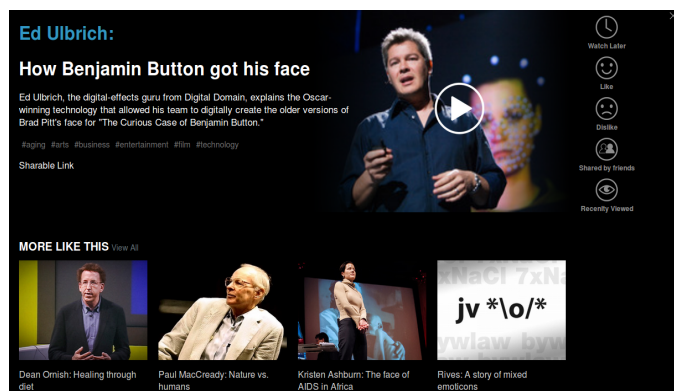


Figure 12: Select Programme Visualization, New N-Screen

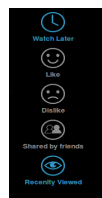


Figure 13: Example of Coloured Icons, New N-Screen

Additionally, it has been added below a programme description an inline set of hyperlinked tags extracted for the programme metadata, see Figure 14. A user can click to a desired tag to explore further related programmes. Figure 15 show the visualization of this action.

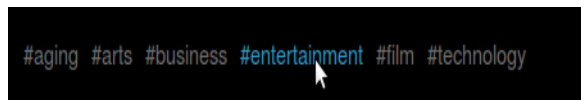


Figure 14: Hyperlinked Tags, New N-Screen

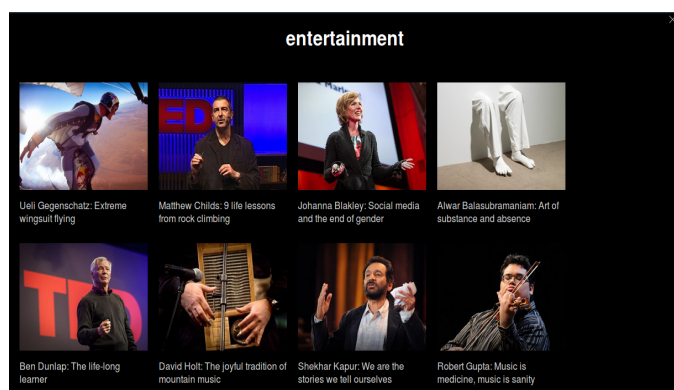


Figure 15: Visualization of related programmes after the selection of a tag, New N-Screen

At last, it has been included a clickable selection of related programmes below its summary, placed on the 'More Like This' section.

5. EVALUATION AND RESULTS

Initially, some tests were applied to the final platform for the purpose of proving its proper functionality. The main features that make the platform valuable such as registration, personal lists updating, sharing to a faraway user and remote TV have been tested. Afterwards, when the functionalities were demonstrated to work, an evaluation of the user experience has been carried out. The descriptions of the tests and corresponding results are exposed in this section.

5.1 Technical Tests

This tests tried to prove that every functional part included in the source code was working. The test was carried out by ourselves and we tested the following features:

- Registration
 - Username not already taken by another user
 - Double entering password
 - OpenID authorization
 - Database members table initialization
 - Database content table initialization
 - Cache
- Login
 - Username and corresponding password
 - OpenID authorization
 - Security
 - User personal row in database members table
 - User personal row in database content table
 - Database content table initialization
 - Cache
- Homepage
 - Content
 - Channels - Personal Lists
 - View All
 - Shared by in Shared by Friends
 - Random Selection
 - User name
 - Group number
 - Group member names
 - Drag&Drop
 - Virtual TV
- Personal Lists Update
 - Shared by Friends
 - Recently Viewed
 - Watch Later
 - Likes

- Dislikes
- Call-to-Action icons
 - Watch Later
 - Like
 - Dislike
- Non-Call-to-Action icons
 - Shared by Friends
 - Recently Viewed
- Faraway Programme Sharing to a Single User
 - Unique notifications
 - Unique Shared by friends channel
- Faraway Programme Sharing to a Group
 - Group Notifications
 - Group Shared by friends channel
- Programme selection
 - Information displayed
 - Local play
 - Call-to-action icons
 - Tags
 - More Like this
- Virtual TV
 - Sharing
 - Play

After the test we ascertained that every part listed above was correctly working.

5.2 User Experience Evaluation

To accomplish the success of our platform, it was necessary to consider testing its interface as well. An interface is the intermediary between users and content. It helps the user to interpreter and guides them through different complexities found on an interactive web system. "The happy marriage of architecture and interface—of logical structure and visual meaning—creates a cohesive user experience"[?].

6. CONCLUSIONS

This paragraph will end the body of this sample document. Remember that you might still have Acknowledgments or Appendices; brief samples of these follow. There is still the Bibliography to deal with; and we will make a disclaimer about that here: with the exception of the reference to the L^AT_EX book, the citations in this paper are to articles which have nothing to do with the present subject and are used as examples only.

7. ACKNOWLEDGMENTS

This section is optional; it is a location for you to acknowledge grants, funding, editing assistance and what have you. In the present case, for example, the authors would like to thank Gerald Murray of ACM for his help in codifying this *Author's Guide* and the .cls and .tex files that it describes.

8. REFERENCES

- [1] \$_session — php manual. [Online, Available at <http://php.net/manual/es/reserved.variables.session.php>].
- [2] J. Allen and J. Chudley. *Smashing UX design: Foundations for designing online user experiences*, volume 34. John Wiley & Sons, 2012.
- [3] L. Aroyo, A. Conconi, S. Dietze, A. Kaptein, L. Nixon, C. Nufer, D. Palmisano, L. Vignaroli, and M. Yankova. Notube—making tv a medium for personalized interaction. 2009.
- [4] M. Cettolo, C. Girardi, and M. Federico. Wit3: Web inventory of transcribed and translated talks. In *Proceedings of the 16th Conference of the European Association for Machine Translation (EAMT)*, pages 261–268, 2012.
- [5] G. Colborne. *Simple and usable web, mobile, and interaction design*. New Riders, 2010.
- [6] W. W. W. Consortium. Ajax — w3c. [Online; Available at http://www.w3schools.com/ajax/ajax_intro.asp].
- [7] W. W. W. Consortium. Openid — w3c. [Online; Available at <http://www.w3.org/2005/Incubator/federatedsocialweb/wiki/Protocols#0penID>].
- [8] L. Cruickshank, E. Tseklevs, R. Whitham, A. Hill, and K. Kondo. Making interactive tv easier to use: Interface design for a second screen approach. *The Design Journal*, 10(3):41–53, 2007.
- [9] A. De Angeli, A. Sutcliffe, and J. Hartmann. Interaction, usability and aesthetics: what influences users' preferences? In *Proceedings of the 6th conference on Designing Interactive systems*, pages 271–280. ACM, 2006.
- [10] B. Dictionary. Browser-based — business dictionary. [Online, Available at <http://www.businessdictionary.com/definition/browser-based-applications.html>].
- [11] Evolumedia Group & Canadian Media Fund. The second screen and television. white paper series number 1. overview and growth perspectives. October 2012.
- [12] Evolumedia Group & Canadian Media Fund. The second screen and television. white paper series number 2. benefits & impacts. March 2013.
- [13] W. S. Green and P. W. Jordan. *Pleasure with products: Beyond usability*. CRC Press, 2003.
- [14] N. Hritzuk and K. Jones. *Multiscreen Marketing: The Seven Things You Need to Know to Reach Your Customers Across TVs, Computers, Tablets, and Mobile Phones*. John Wiley & Sons, 2014.
- [15] G. Z.-h. L. Jia-xi and W. Jin. Xmpp-based ejabberd server privacy-list implementation. *Microcomputer Information*, 18:074, 2010.
- [16] S. Krug. Dont make me think, revisited: A common sense approach to web (and mobile) usability. *Berkley: New Riders*, 2014.
- [17] Microsoft Advertising, Flamingo & Ipsos OTX. Cross-screen engagement: Multi-screen pathways reveal new opportunities for marketers to reach and engage consumers. March 2013.
- [18] A. Monk. The product as a fixed-effect fallacy.

- Human-Computer Interaction*, 19(4):371–375, 2004.
- [19] C. Newell and L. Miller. Design and Evaluation of a Client-side Recommender System. Technical report, British Broadcasting Corporation, 01 2014.
 - [20] J. Nielsen, M. Tahir, and M. Tahir. *Homepage usability: 50 websites deconstructed*, volume 50. New Riders Indianapolis, IN, 2002.
 - [21] Nielsen Holdings N.V. Action figures: How second screens are transforming tv viewing. Technical report, June 2013.
 - [22] D. A. Norman. *The design of everyday things*. Basic books, 2002.
 - [23] P. Pu and L. Chen. Trust building with explanation interfaces. In *Proceedings of the 11th international conference on Intelligent user interfaces*, pages 93–100. ACM, 2006.
 - [24] B. Schopman, D. Brickly, L. Aroyo, C. van Aart, V. Buser, R. Siebes, L. Nixon, L. Miller, V. Malaise, M. Minno, et al. Notube: making the web part of personalised tv. 2010.
 - [25] C. Shih. *The Facebook era: Tapping online social networks to build better products, reach new audiences, and sell more stuff*. Prentice Hall, 2009.
 - [26] Y. X. Skadberg and J. R. Kimmel. Visitors flow experience while browsing a web site: its measurement, contributing factors and consequences. *Computers in human behavior*, 20(3):403–422, 2004.
 - [27] Techopedia. Platform — techopedia. [Online, Available at <http://www.techopedia.com/definition/3411/platform>].
 - [28] Think With Google. The New Multi-Screen World: Understanding Cross-Platform Consumer Behavior. *Think Newsletter*, August 2012.
 - [29] Webopedia. Cross-platform — webopedia. [Online, Available at http://www.webopedia.com/TERM/C/cross_platform.html].
 - [30] Webopedia. Sdk — webopedia. [Online, Available at <http://www.webopedia.com/TERM/S/SDK.html>].
 - [31] Wikipedia. Halo effect — wikipedia, the free encyclopedia, 2014. [Online; accessed 22-January-2015, Available at http://en.wikipedia.org/w/index.php?title=Halo_effect&oldid=638449052].
 - [32] Wikipedia. Application software — wikipedia, the free encyclopedia, 2015. [Online; accessed 21-January-2015, Available at http://en.wikipedia.org/w/index.php?title=Application_software&oldid=641246771].
 - [33] Wikipedia. Bbc — wikipedia, the free encyclopedia, 2015. [Online; accessed 21-January-2015, Available at <http://en.wikipedia.org/w/index.php?title=BBC&oldid=643446361>].
 - [34] Wikipedia. Http cookie — wikipedia, the free encyclopedia, 2015. [Online; accessed 21-January-2015, Available at http://en.wikipedia.org/w/index.php?title=HTTP_cookie&oldid=643349690].
 - [35] Wikipedia. Human multitasking — wikipedia, the free encyclopedia, 2015. [Online; accessed 21-January-2015, Available at http://en.wikipedia.org/w/index.php?title=Human_multitasking&oldid=642991102].
 - [36] Wikipedia. Hyperlink — wikipedia, the free encyclopedia, 2015. [Online; accessed 21-January-2015, Available at <http://en.wikipedia.org/w/index.php?title=Hyperlink&oldid=643531082>].

- [37] Wikipedia. Md5 — wikipedia, the free encyclopedia, 2015. [Online; accessed 21-January-2015, Available at <http://en.wikipedia.org/w/index.php?title=MD5&oldid=641279744>].
- [38] Wikipedia. Xmpp — wikipedia, the free encyclopedia, 2015. [Online; accessed 21-January-2015, Available at <http://en.wikipedia.org/w/index.php?title=XMPP&oldid=643175467>].

APPENDIX

A. HEADINGS IN APPENDICES

The rules about hierarchical headings discussed above for the body of the article are different in the appendices. In the **appendix** environment, the command **section** is used to indicate the start of each Appendix, with alphabetic order designation (i.e. the first is A, the second B, etc.) and a title (if you include one). So, if you need hierarchical structure *within* an Appendix, start with **subsection** as the highest level. Here is an outline of the body of this document in Appendix-appropriate form:

A.1 Introduction

A.2 The Body of the Paper

A.2.1 Type Changes and Special Characters

A.2.2 Math Equations

Inline (In-text) Equations

Display Equations

A.2.3 Citations

A.2.4 Tables

A.2.5 Figures

A.2.6 Theorem-like Constructs

A Caveat for the \TeX Expert

A.3 Conclusions

A.4 Acknowledgments

A.5 Additional Authors

This section is inserted by \LaTeX ; you do not insert it. You just add the names and information in the `\additionalauthors` command at the start of the document.

A.6 References

Generated by bibtex from your `.bib` file. Run latex, then bibtex, then latex twice (to resolve references) to create the `.bbl` file. Insert that `.bbl` file into the `.tex` source file and comment out the command `\thebibliography`.

B. MORE HELP FOR THE HARDY

The `acm_proc_article-sp` document class file itself is chock-full of succinct and helpful comments. If you consider yourself a moderately experienced to expert user of \LaTeX , you may find reading it useful but please remember not to change it.