SEACW DELIVERABLE D.1.3

"Contribution to Standardization Report"

Project Acronym	**SEACW**
Grant Agreement No.	325146
Project Title	Social Ecosystem for Antiaging, Capacitation and Well-Being
Deliverable Reference Number	SEACW_D.1.3
Deliverable Title	Contribution to Standardization Report
Revision Number	2.0
Deliverable Editors *(main writers)*	UC3M ÁLIAD

Project co-funded by the European Commission within the ICT Policy Support Programme	
Dissemination Level	
RE	RESTRICTED

Revision	Date	Description
0	09/09/2013	Definition of objectives and expected results
0.1	18/11/2013	First document revision
0.2	11/12/2013	Second document revision
1.0	03/02/2014	Deliverable Version 1
1.1	15/09/2014	Review of standardization report section
1.2	17/12/2014	Completion of standardization report with monitoring service and segmented user statistics service
1.3	16/02/2015	Completion of social developer specific actions
2.0	23/03/2015	Deliverable Version 2

Statement of originality:

This deliverable contains original unpublished work except where clearly indicated otherwise. Acknowledgement of previously published material and work of others has been made through appropriate citation, quotation or both.

Abstract:

This document introduces the contributions proposed by the consortium who is taking part of the project for the improvement and standardization of European projects.

This document is the second and last of two deliveries. The current delivery takes place in the 24th month of the project, while the first one took place in the 12th month.

TABLE OF CONTENTS

1 Introduction

This document introduces the contributions for the standardization of European projects.

The consortium proposes the next contributions for standardization:

- A style guide document. This document explains why such a style guide should be a standard and displays some examples of the guide made for this project. The complete style guide is available in document D3.6 “The human interface requirements document” (appendix I).

- A monitoring service. This generic service is able to track the behavior of the physical architecture of the ecosystem, in order to record availability, response times and specific performance metrics. It also incorporates alerts to report system administrators when some component is malfunctioning, and artificial intelligence behavior to predict future malfunctions based on early symptoms.

- A segmented user statistics service, which records the number of users registered for each country, in order to provide partners with this information in real time, so that they can check their own progress. This service can be adapted to observe any other metric.

- The creation of the concept of “social development”, and specific actions carried out during this project to achieve the elaboration of this concept.

- The creation of an opensource community which is aimed at developing software for the scope of Active and Healthy Ageing.

With these proposals, the consortium intends to improve the execution of European projects and to carry out activities to promote the concept of Active and Healthy Ageing.

2 Standardization Report

For contributing to the standardization, SEACW consortium proposes to standardize a set of documents serving as guide for the future projects that will be developed.

The proposed document is the **style guide**. This document is especially important in a project aimed at developing software for social inclusion, as the outcomes themselves have very specific needs and thus, special care should be taken when studying the style and developing in the development phase of the project.

Moreover, two reports about the project have been proposed. The first one provides continuos monitoring of the physical infrastructure to system administrators while the second provides partner countries with their progress in acquiring users. Both reports are proposed as standard software which could be easily adapted to any other project with similar requirements.

2.1 Style guide

Why a style guide is needed?

We consider that, before development, it is very important to carefully analyse the scope and possible difficulties that final users could find.

For that reason, the style guide should be designed and written following the next steps:

➡ First, the concept of what is going to be developed must be clearly detailed, taking into consideration the target users and the objectives to be accomplished after the development.

➡ Design the structure for the development, what outcomes will be shown to users and what elements are common to all these outcomes.

➡ Study the accessibility guidelines which should be followed given the users and norms established by the W3C in the "Web Accessibility Initiative".

➡ Design the architecture of information, a schema which details page by page where and when will be shown the information during the development, making mockups for the design and, finally, understanding what the final product will be.

➡ Define the style, where all the mockups made in the previous phase are analysed, and all the technical specifications which should be followed during the development are observed, such as:

- Metrics and grids
- Colour
- Typography
- States and feedback.
- Iconography
- Pictures and media

This document aims at facilitating the work of developers, such as the specification of the design and functionality of software products. Moreover, the communication between the different departments or partners taking part in the project gets simpler and efficient, as all of them already have a copy of the specifications of the final product before the development phases start.

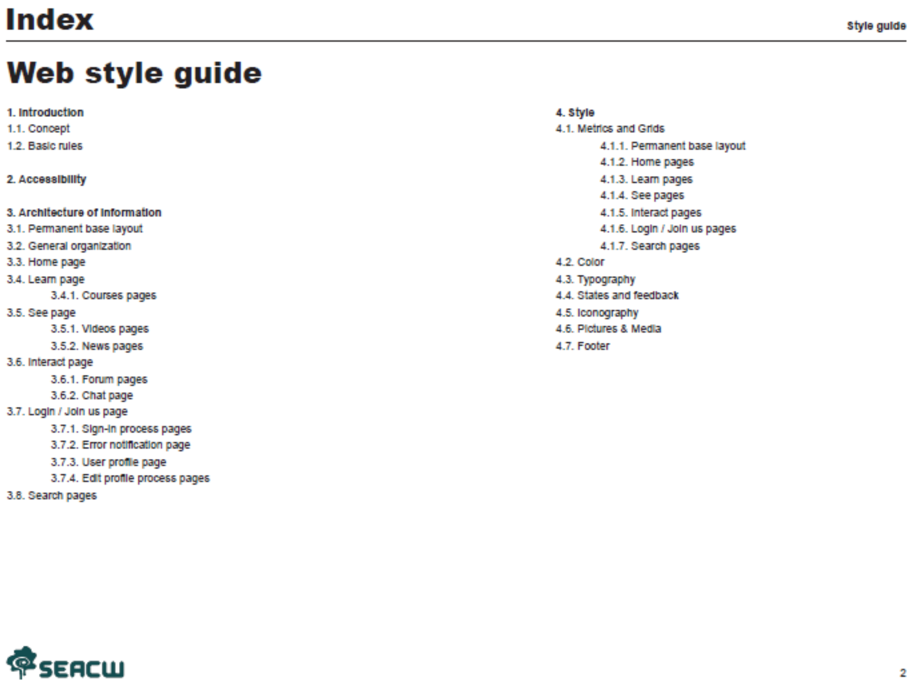
Because of all the advantages described so far, the style guide provides many benefits during the development of a project, regardless its size, as it increases the efficiency of the different teams or partners, both in terms of communication and of technical work.

It is very important that this document is elaborated for all different devices and/or environments from which the development will be accessible.

Style guide samples

Next, a set of screenshots for the style guide developed for the SEACW Project are displayed.

2.1.1.1 Web style guide index



Index

Style guide

Web style guide

1. Introduction
1.1. Concept
1.2. Basic rules

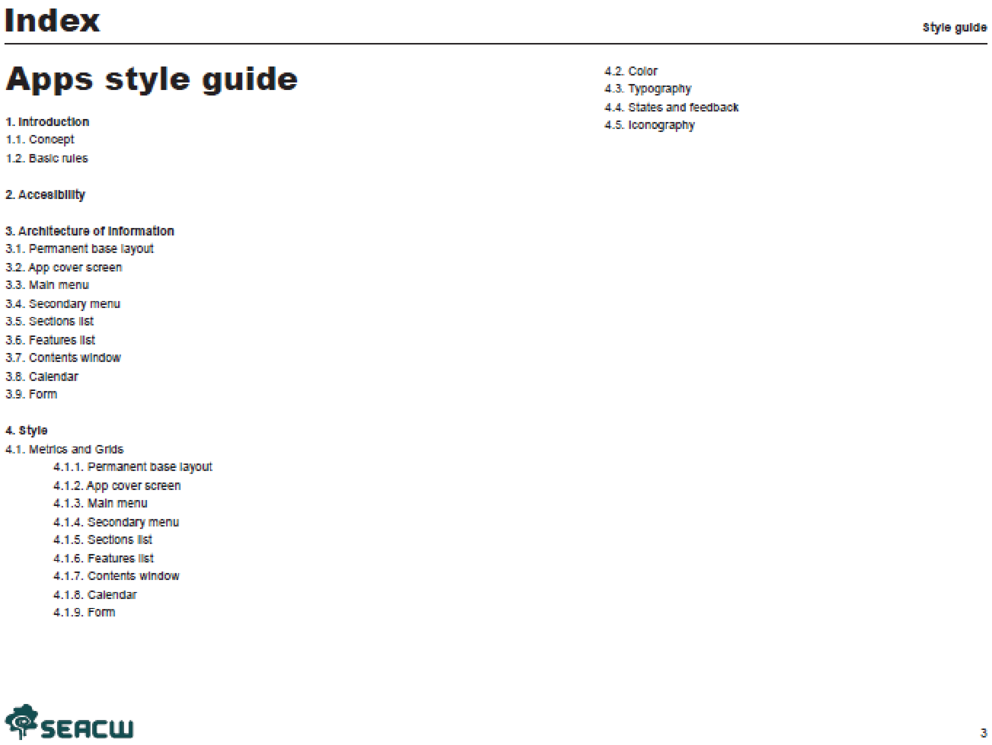
2. Accessibility

3. Architecture of information
3.1. Permanent base layout
3.2. General organization
3.3. Home page
3.4. Learn page
3.4.1. Courses pages
3.5. See page
3.5.1. Videos pages
3.5.2. News pages
3.6. Interact page
3.6.1. Forum pages
3.6.2. Chat page
3.7. Login / Join us page
3.7.1. Sign-in process pages
3.7.2. Error notification page
3.7.3. User profile page
3.7.4. Edit profile process pages
3.8. Search pages

4. Style
4.1. Metrics and Grids
4.1.1. Permanent base layout
4.1.2. Home pages
4.1.3. Learn pages
4.1.4. See pages
4.1.5. Interact pages
4.1.6. Login / Join us pages
4.1.7. Search pages
4.2. Color
4.3. Typography
4.4. States and feedback
4.5. Iconography
4.6. Pictures & Media
4.7. Footer

2

2.1.1.2 Apps style guide index

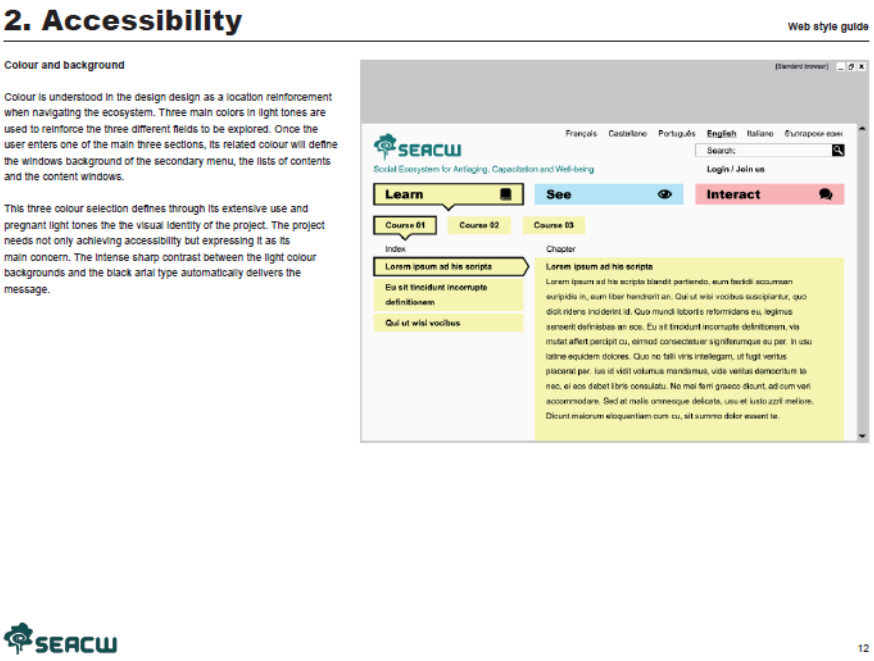


Index

Style guide

Apps style guide

2.1.1.3 Accessibility section



2. Accessibility

Web style guide

Colour and background

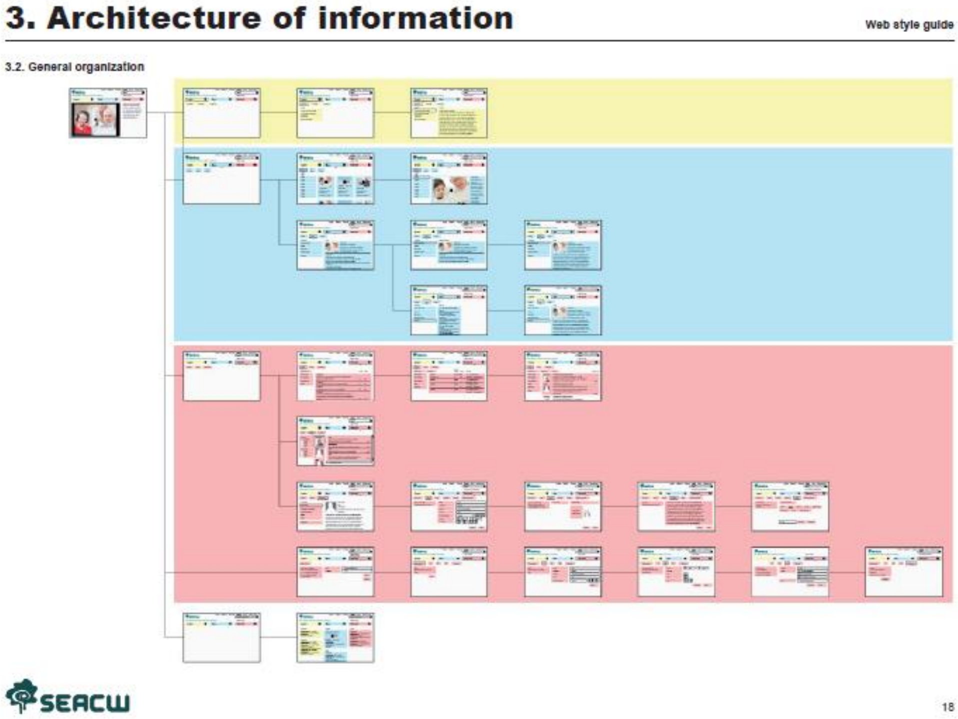
Colour is understood in the design design as a location reinforcement when navigating the ecosystem. Three main colors in light tones are used to reinforce the three different fields to be explored. Once the user enters one of the main three sections, its related colour will define the windows background of the secondary menu, the lists of contents and the content windows.

This three colour selection defines through its extensive use and pregnant light tones the the visual identity of the project. The project needs not only achieving accessibility but expressing it as its main concern. The intense sharp contrast between the light colour backgrounds and the black arial type automatically delivers the message.

Lorem ipsum ad his scripta

Lorem ipsum ad his scripta blandit partiendo, eum fastidii accumsan euripidis in, eum liber hendrerit an. Qui ut wisi vocibus suscipiantur, quo dicit ridens inciderint id. Quo mundi lobortis reformidans eu, legimus senserit definiebas an eos. Eu sit tincidunt incorrupte definitionem, vis mutat affert percipit cu, eirmod consectetuer signiferumque eu per. In usu latine equidem dolores. Quo no falli viris intellegam, ut fugit veritus placerat per. Ius id vidit volumus mandamus, vide veritus democritum te nec, ei eos debet libris consulatu. No mei ferri graeco dicunt, ad cum veri accommodare. Sed at malis omnesque delicata, usu et iusto zzril meliore. Dicunt maiorum eloquentiam cum cu, sit summo dolor essent te.

2.1.1.4 *Architecture of information – General Organization*



3. Architecture of information

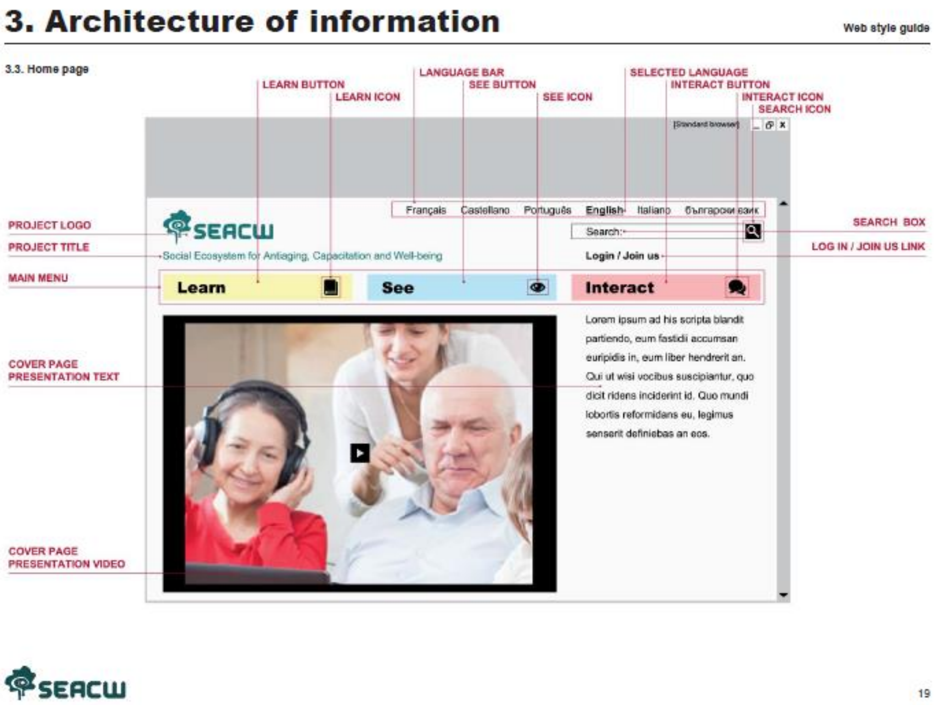
Web style guide

3.2. General organization

SEACW

18

2.1.1.5 *Architecture of information – Main Page*



3. Architecture of information

Web style guide

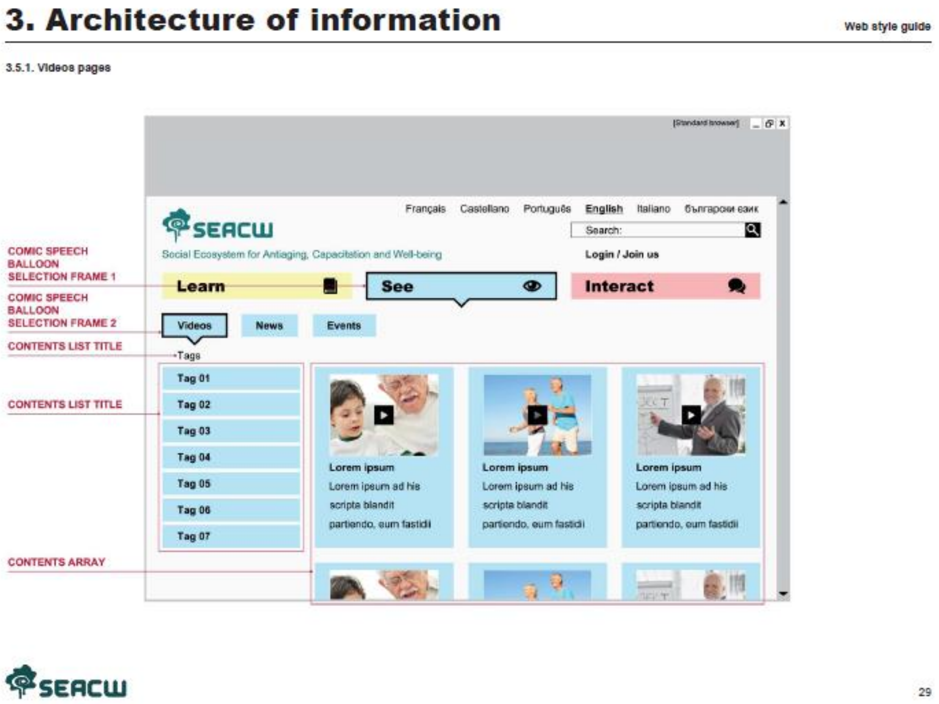
3.3. Home page

Lorem ipsum ad his scripta blandit partiendo, eum fastidii accumsan euripidis in, eum liber hendrerit an. Qui ut wisi vocibus suscipiantur, quo dicit ridens inciderint id. Quo mundi lobortis reformidans eu, legimus senserit definiebas an eos.

SEACW

19

2.1.1.6 *Architecture of information – Videos Page*



3. Architecture of information

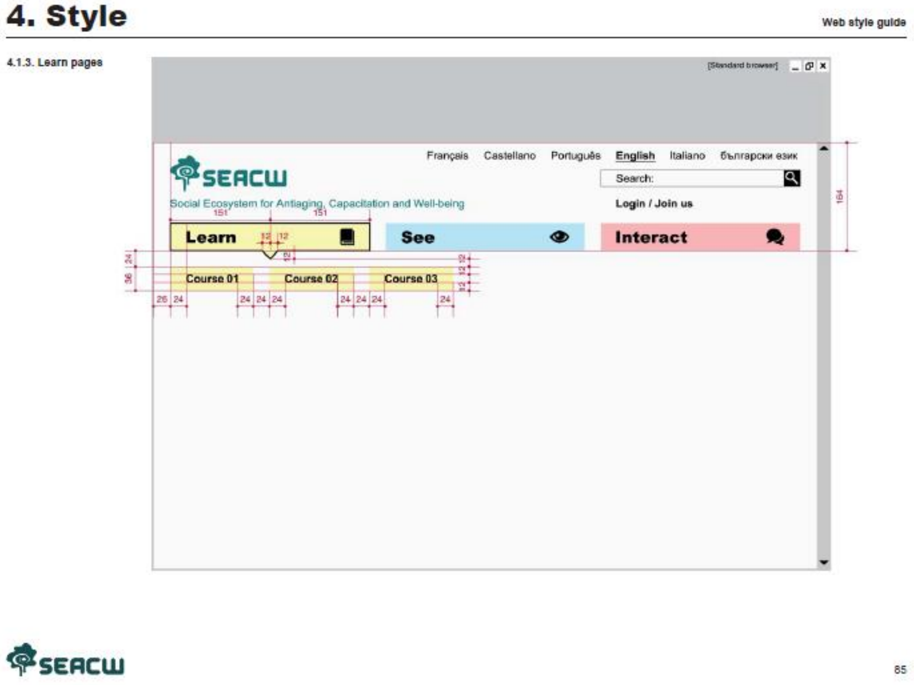
Web style guide

3.5.1. Videos pages

[Standard browser]
Français
Castellano
Português
English
Italiano
български език
SEACW
Social Ecosystem for Antiaging, Capacitation and Well-being
Search:
Login / Join us
Learn
See
Interact
Videos
News
Events
Tags
Tag 01
Tag 02
Tag 03
Tag 04
Tag 05
Tag 06
Tag 07
Lorem ipsum
Lorem ipsum ad his scripta blandit partiendo, eum fastidii
Lorem ipsum
Lorem ipsum ad his scripta blandit partiendo, eum fastidii
Lorem ipsum
Lorem ipsum ad his scripta blandit partiendo, eum fastidii
COMIC SPEECH BALLOON SELECTION FRAME 1
COMIC SPEECH BALLOON SELECTION FRAME 2
CONTENTS LIST TITLE
CONTENTS LIST TITLE
CONTENTS ARRAY
SEACW

29

2.1.1.7 *Style – Learn Menu*



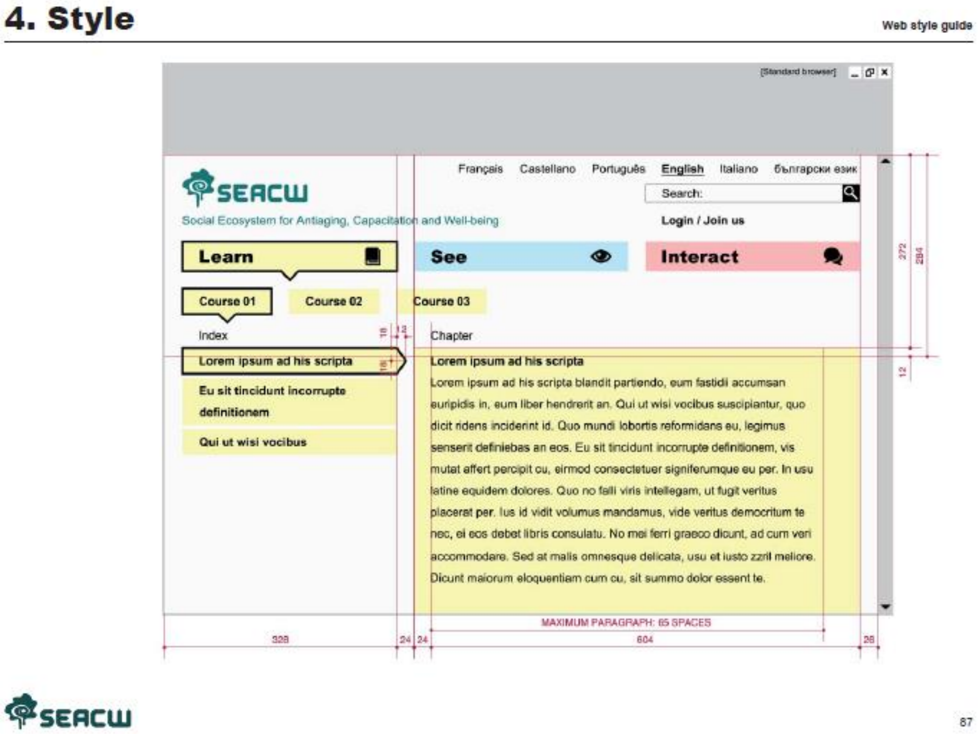
4. Style

Web style guide

4.1.3. Learn pages

85

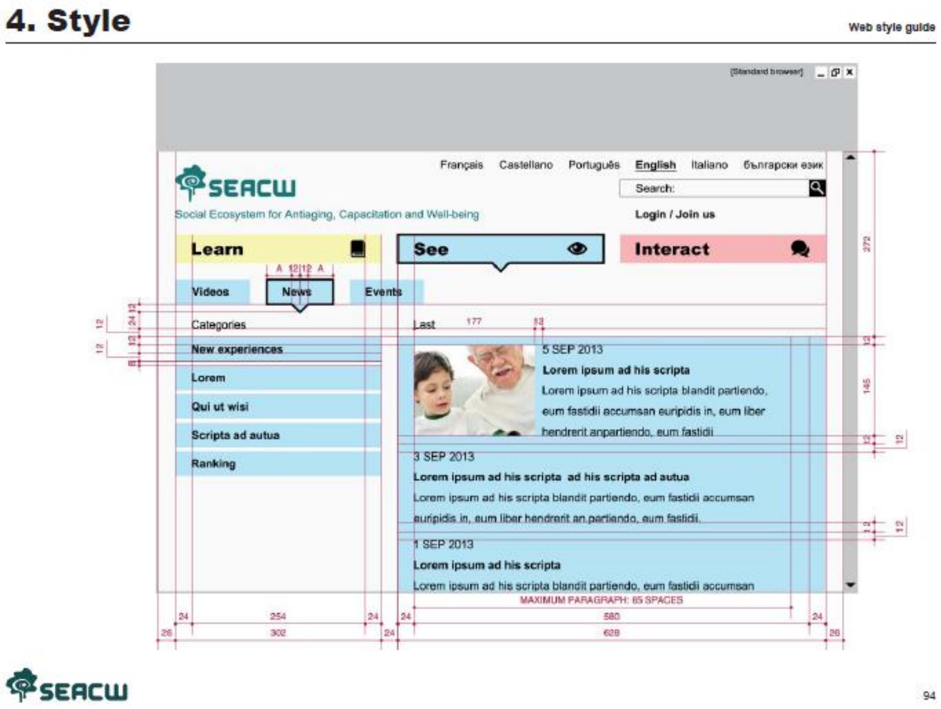
2.1.1.8 Style – Course Page



Lorem ipsum ad his scripta

Lorem ipsum ad his scripta blandit partiendo, eum fastidi accumsan euripidis in, eum liber hendrerit an. Qui ut wisi vocibus suscipiantur, quo dicit ridens inciderint id. Quo mundi lobortis reformidans eu, legimus senserit definiebas an eos. Eu sit tincidunt incorrupte definitionem, vis mutat affert percipit cu, eirmod consectetuer signiferumque eu per. In usu latine equidem dolores. Quo no falli viris intellegam, ut fugit veritus placerat per. Ius id vidit volumus mandamus, vide veritus democritum te nec, ei eos debet libris consulatu. No mei ferri graeco dicunt, ad cum veri accommodare. Sed at malis omnesque delicata, usu et iusto zzril meliore. Dicunt maiorum eloquentiam cum cu, sit summo dolor essent te.

2.1.1.9 Style – News Page



5 SEP 2013

Lorem ipsum ad his scripta

Lorem ipsum ad his scripta blandit partiendo, eum fastidii accumsan euripidis in, eum liber hendrerit anpartiendo, eum fastidi

3 SEP 2013

Lorem ipsum ad his scripta ad his scripta ad autua

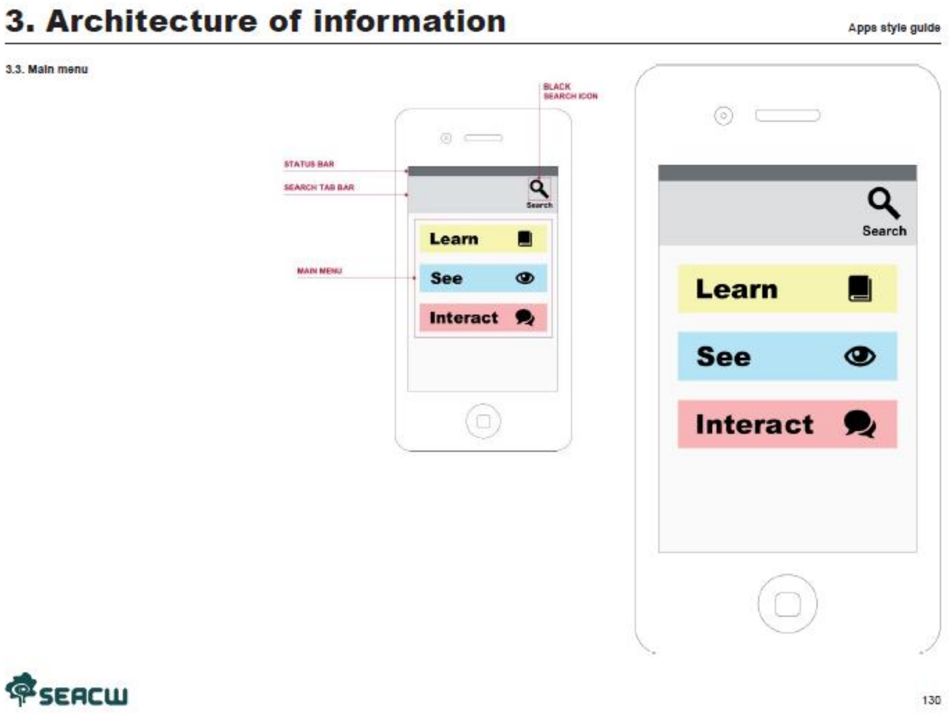
Lorem ipsum ad his scripta blandit partiendo, eum fastidi accumsan euripidis in, eum liber hendrerit an partiendo, eum fastidi.

1 SEP 2013

Lorem ipsum ad his scripta

Lorem ipsum ad his scripta blandit partiendo, eum fastidi accumsan

2.1.1.10 Architecture of information – Apps Main Menu



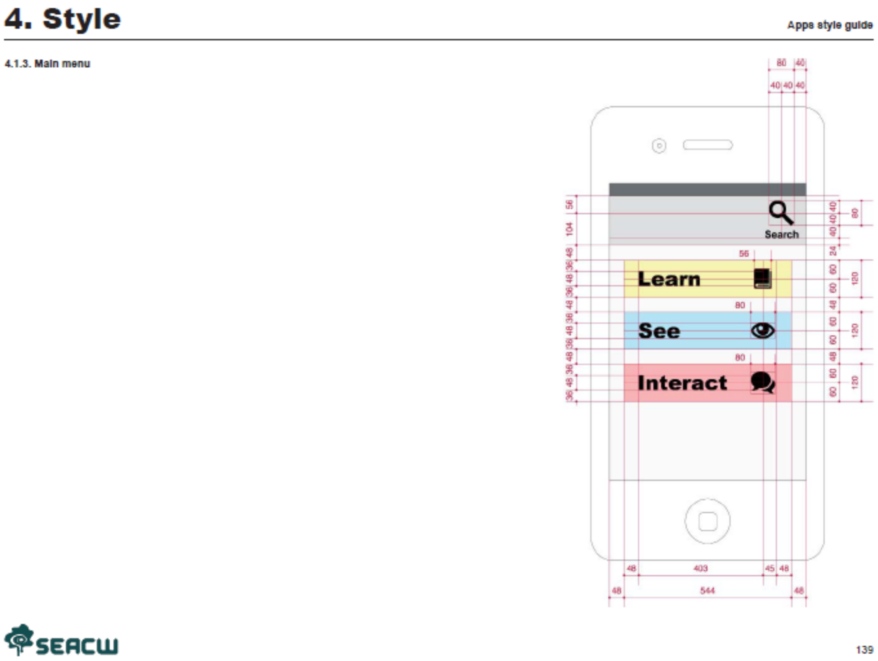
3. Architecture of information

Apps style guide

3.3. Main menu

130

2.1.1.11 Style – Apps Main Menu



4. Style

Apps style guide

4.1.3. Main menu

139

2.2 Monitoring service

2.3.1. Why a monitoring service is needed?

It is often the case that web applications in the server side require high availability in order to provide a functioning service to the users and clients.

In order to check for malfunctioning behaviors and detect issues as soon as possible, it is required to implement a 24/7 automated monitoring services which record at the very least the availability of all servers and services, and the response times. Additional metrics can also be recorded in order to improve the monitoring task.

The use of monitoring alerts is important in order to let the system administrators know the fact that some component or service is not working properly, and providing information about the detected issues so that they can be easily identified and fixed as soon as possible in order to have all the services working. Of course, these alerts also prevent system administrators from having to perform repetitive tasks, such as:

- Checking continuously the availability of physical servers.
- Checking that response times are acceptable for all services.
- Checking that all services are listening and responding to requests.
- Checking periodically that the servers free space and resources are within an acceptable range (i.e. there is enough free memory and storage to perform daily tasks).

As a result, system administrators will be able to spend this time in performing other administration tasks.

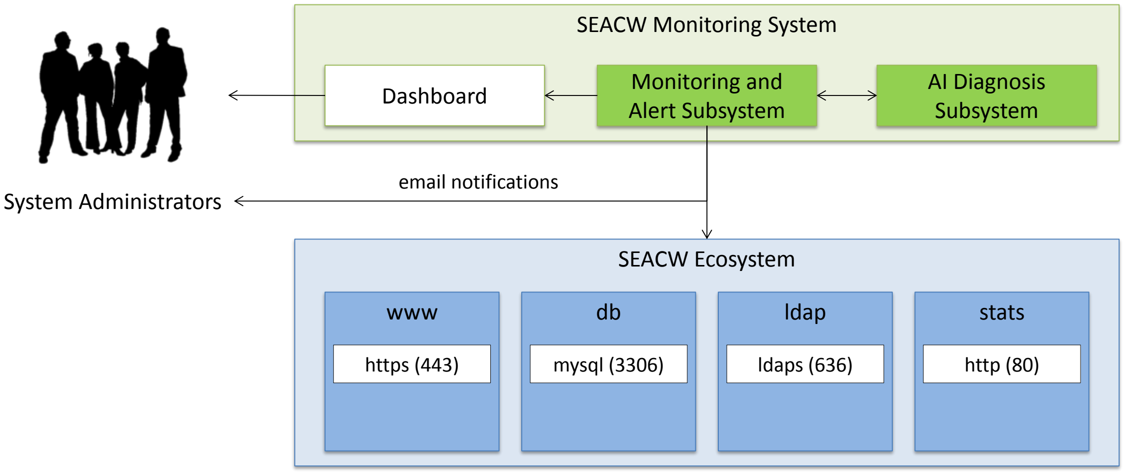
Moreover, this system can be improved with machine learning in order to diagnose potential malfunctions when they start to show early symptoms, for instance, if the response time for a server is increasing over time, it may be due to hardware malfunctioning or because of too much load, and may be an indicator of the system failing in the shortcoming future. Early diagnosis makes it possible for system administrators to anticipate these problems and provide actual solutions before failure.

2.3.2. SEACW monitoring system

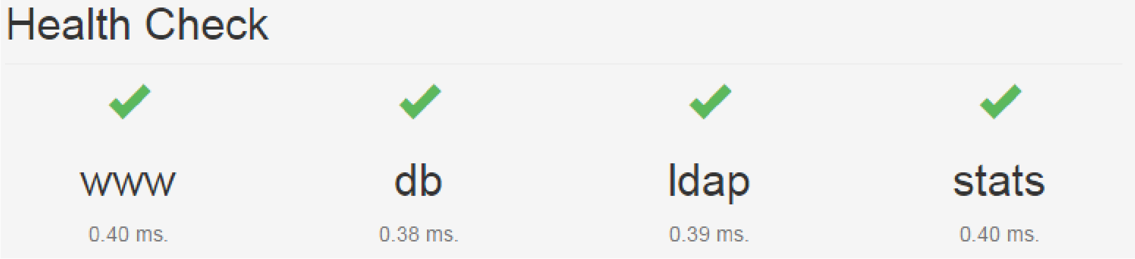
SEACW's technical architecture is very complex, as it involves different physical and virtual machines each providing specific services either to final users, to other services or to project partners; as described in deliverables D1.6 and D3.4.

Such a complex infrastructure requires from an intelligent monitoring service as described above in order to track any possible failure in the system and detect it as soon as it happens (or better yet, when it is displaying early symptoms and before the real failure is noticeable by other users).

An external (in a different machine and network), highly available service provides continuous monitoring of the whole SEACW ecosystem, displaying an intuitive interface for system administrators to query.



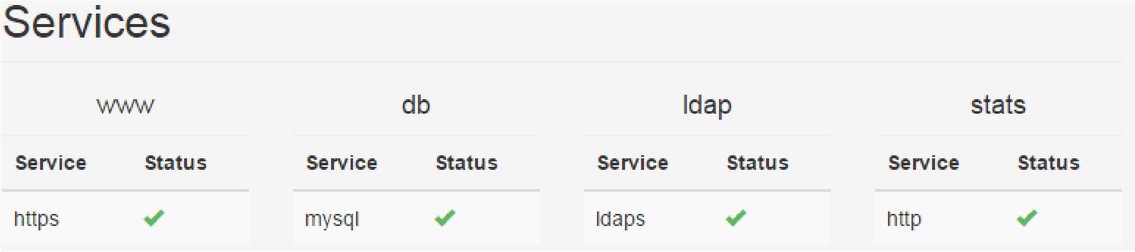
The monitoring service checks first the availability of each machine and their responses time. Green check marks are shown for each machine available and with an acceptable response time. On the other hand, red crosses are shown if any machine is not available and is not responding, which most likely would indicate a failure.



Health Check

✓	✓	✓	✓
www	db	ldap	stats
0.40 ms.	0.38 ms.	0.39 ms.	0.40 ms.

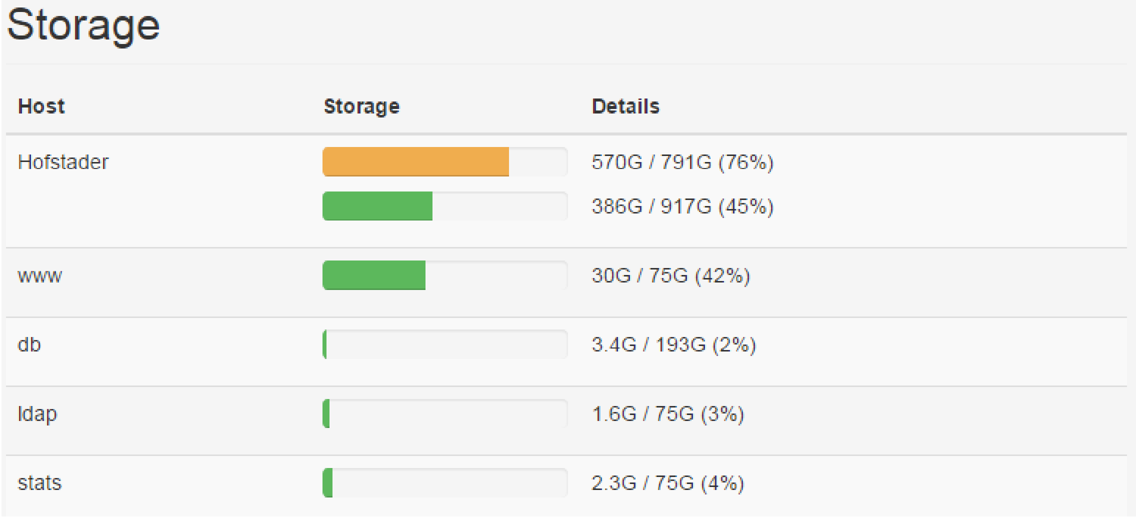
In the second place, the monitoring service checks the availability of each individual service. Each machine may host one or more services, and even if the machine is responding (as shown in the previous figure), it may be the case that the service is not really attending requests. In that case, a red cross would be shown near to the service name.



Services

www		db		ldap		stats	
Service	Status	Service	Status	Service	Status	Service	Status
https	✓	mysql	✓	ldaps	✓	http	✓

Finally, the panel shows the storage for each machine with a bar, which changes the color depending on whether the free space is ideal (green), enough (yellow) or not enough (red). If a machine is running out of free storage, it should be known with enough time to provision more storage or acquire new disks.



Storage

Host	Storage	Details
Hofstader		570G / 791G (76%)
		386G / 917G (45%)
www		30G / 75G (42%)
db		3.4G / 193G (2%)
ldap		1.6G / 75G (3%)
stats		2.3G / 75G (4%)

Finally, besides the monitoring panel, email alerts are configured so that in the case of critical failures or symptoms being detected, system administrators receive all the information about the failure or symptom detected so that they can provide a fast response.

While this monitoring system has been developed for SEACW project, it has been designed with reutilization in mind, and it is easy to adapt to other websites or ecosystems, regardless of the number of physical or virtual machines and services.

2.3 Segmented user statistics service

2.3.1. Why a segmented user statistics service?

Most of the sites allowing registration of users (including most common CMSs) have simple ways to get the total number of registered users, but they often fail to provide a more detailed segmentation about these user (source, gender, or other dimensions).

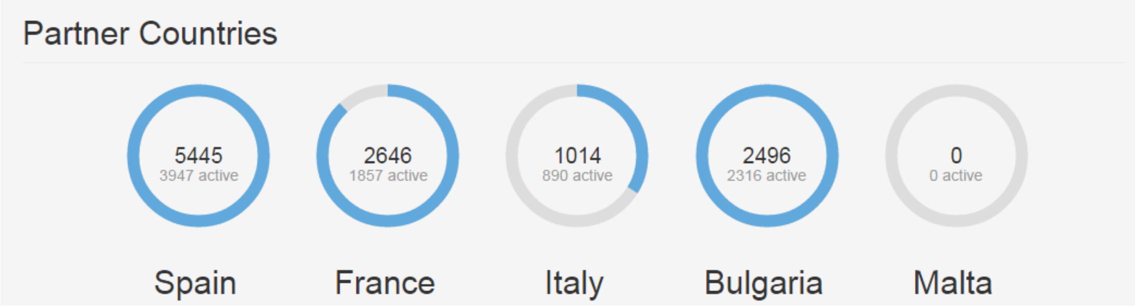
On the other hand, analytics tracking software usually provide this segmentation, but they require specific configuration and most importantly, they do not guarantee recording accurate information about all users, as some may have JavaScript disabled or specific software or browser plugins to prevent navigation to be tracked.

For this kind of sites, a service providing updated statistics about users’ registration for specifically designed segments is interesting.

2.3.2. SEACW segmented user statistics service

For the SEACW project and for reporting accurate information about the users’ origin a specific service has been developed. By means of this service, this information can be easily collected by all SEACW partners, without requiring specific permisions in an analytics tracking tool. Moreover, the information is shown in an intuitive and minimalist way so that each country can know the number of users that has registered from that specific place.

This dashboard displays the total number of users as well as the total number among them who have activated their accounts. Moreover, this same information is displayed for each partner country, showing a circular progress chart indicating the percentage of users in a country with respect to the minimum to be acquired for that country.



Partner Countries

Spain	France	Italy	Bulgaria	Malta
5445 3947 active	2646 1857 active	1014 890 active	2496 2316 active	0 0 active

This service has been developed for the SEACW project but with reutilization in mind, so that the segments can be defined for any other dimension (e.g. regions instead of countries).

3 Social Developer

SEACW Project is framed within a subset of European projects with a strong social nature, not only because of its topic, but also for its objectives, which are aimed at changing society, improving it and making it more human, relying on ICTs to achieve these goals.

This kind of initiatives could be framed within a volunteering movement which is appearing in different cities all over the world, also referred to as "technological volunteering".

The objective being technological volunteering is to bring the New Information and Communication Technologies closer to everybody, and especially to those people who have higher handicaps to get access to those, or who are in risk of digital exclusion because of their lack of technological knowledge, thus reducing the inequality in the access to the Information Society and the Digital World.

Technological volunteering is a new, different way of becoming a volunteer. Any person with a certain minimum knowledge can turn into the perfect volunteer to help those who are taking the first steps to the use of new technologies. If you can find some spare time and you enjoy sharing your knowledge, you definitely should try this experience.

Once the concept of technological volunteering is established, we also want to go a step beyond and incorporate to this movement the concept of "social development".

Figure 1. A volunteer with an elderly

3.1 Context

Within the technological world, during the last years different initiatives have arised whose objective were to help those people who were in risk of being excluded of the digital society. A famous example would be the OLPC (One Laptop per Child) project, where technology is put to the service of society.

Additionally, other altruistic movements have also arised, such as the Open Source movement where, with completely different purposes (in this case, pursuing the freedom of information in software), many resources and people time is spent in an unselfish manner, achieving spectacular outcomes.

Following the trail of this movement, we definitely think that a new role and a new term should be coined, which is built above the current concept of open source software, and should be called “social developer”.

The concept of open source was first used by the communities in 1998, trying to replace the ambiguous existing term of free software, where “free” could be understood either by the perspective of freedom or from the perspective of cost. For our project, it is intended to understand that the software can be used, read, modified and redistributed with no cost, and users and developers can use it freely.

The term was not accepted by some, considering that it removed the idea of freedom which could be found being “free software”, and could lead to confusion as the software could be understood just as free of cost. However, the new term still can be used by people who, even if they do not distribute the software free of cost, they deliver the source code so that it can be reviewed of modified by customers.

From now on, we will refer with the term “free software” to those pieces of software which are published with total freedom of use, modification and redistribution under the only explicit constraint of not modifying or restricting that freedom in the future.

From a semantic point of view, the literal meaning of “open source” is that the source code from a computer software can be observed, so it is clearly more restrictive and weak that the concept of “free software”. However, in most cases both terms are used to refer to the same set of licenses, keeping the same philosophy.

However, an explicit difference should be remarked between that open source software that enable users to modify and improve the software, and that software which only allows to visualize the code, restricting its use or modification.

Nowadays, the open source concept is also used to refer to a new software movement (the Open Source Initiative), which is different to the free software movement, as they both have different philosophical principles but with the same freedoms in practice.

The idea behind the concept is simple: when a community of developers can read, modify and redistribute the source code of a software application, then it evolves and improves. Users can adapt the software to their needs, can fix its bugs in very short periods of time and, eventually, produce better software.

3.2 Needs

There are many different social challenges which we confront nowadays. Most of us agree to a certain extent that Internet and the ICTs are a key element in the creation of resources for people. There are many different volunteering movements where people help learning the use of new technologies to people who don’t have any technological background. Our proposal goes a step beyond and aims at providing people tools aimed at improving people wealth, and solving some of their daily problems.

There are different environments where a social developer could be useful:

- Helping disadvantaged communities to digitalize their cultural background
- Developing automatic translation tools for languages in minor use
- Adapting daily-use applications for handicapped people.

The list of possible environments and tasks could be unlimited.

3.3 Specific actions

A hackathon is a development competition where groups of programmers compete for building the best application. The definition of “best” is usually conferred by the organizers, which may award the most innovative idea or the best designed application.

While a hackathon does not directly fit the definition of social development, it certainly has some similarities. In a hackathon, groups often compete for a prize of any kind (monetary prize, gifts or just recognition).

Moreover, the values and motivations of a hackathon involves not only winning the prize (as most of the teams will most likely not be awarded with any) but having a good time, sharing knowledge and programming experience, improving technical abilities and creativity and overall, having fun. In particular, the main values are:

- “Your ideas are your not treasure, please share them.”
- “Collaborate with the other teams to improve your ideas.”
- “Do networking and search for multidisciplinary teams.”
- “Divide your work and conquer.”
- “Share your code with the community and use opensource libraries if possible.”
- “Focus your efforts in achieving success.”

Additionally, it is often the case that developers just program for altruistic purposes, such as in the case of the international competition Hack4Good, where computer experts put their knowledge and abilities at work for achieving a better world, for instance, fighting against catastrophic climate change.

In SEACW, we have tried to apply this philosophy and values in order to promote the figure of the social developer. To do so, we have contributed with T3chFest 2015, the Computer Science and New Technologies fair celebrated annually in Universidad Carlos III de Madrid, one of the SEACW partners.

Figure 2. T3chFest 2015 Hackathon contestants

SEACW has been gold-sponsor of T3chFest and has contributed to the development of the hackathon by providing three Fitbit Charge activity wristbands to be given out as prizes for the best team. This wristband is state-of-the-art wearable technology which records the user steps along the day and allows competition among friends to achieve the largest number of steps during the last 7 days. Moreover, it also tracks the number of active minutes, distance, burnt calories and the nightdream quality.

During the hackathon, twelve groups of three developers each were asked to develop an application promoting and engaging users in having an active and healthy life and behaviour. They arrived at 9AM and they attended a training talk communicating the values of the hackathon, where the organizers insisted on the fact that winning the prizes was not intended to be the hackathon reward, but rather the experience and fun themselves.

They engaged in the programming task continuously until 7PM, with only a short pause for the lunch at 2PM. After all this time and effort, a total of five groups achieved the development of full-competitive apps. From these groups, the organizing committee decided the app most suitable to the purpose of the hackathon, and they were awarded the Fitbit Charge wristbands.

Figure 3. Fitbit Charge activity wristband

4 Software Development Communities

4.1 Why a community?

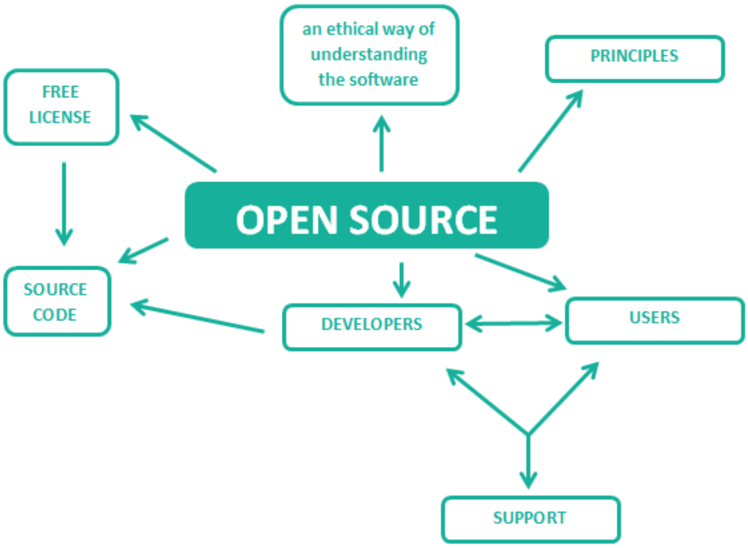


Figure 4. Open source structure

To a management extent, opensource communities are the best example. The functioning of these communities is based on the next features:

➡ Opensource software is an ethical way to understand software from the development, and also in the use and distribution.

- Opensource refers also to a significant set of values: ethics, creativity, efficiency, collaboration, equality, science, transparency, security, competitiveness, privacy, solidarity and, overall, freedom.

- Opensource software requires a free license which guarantee:
    - The right to publish improvements from which the community can take benefit.
    - The right to redistribute copies for anybody.
    - The right to adapt the program for each one's needs.
    - The right to use it with any (legal) purpose.

- Opensource software is distributed with the source code and is compiled with applications which respect open standards and free contents, such as fonts, translations, templates, multimedia, manuals, etc.

- Opensource software is designed for users and developers to cooperate among them, giving and receiving support.

provide
interface freedom fact
online computer browser users
help one members new
offer Library SIDEBAR web makes
system make use systems
libraries community
right different
work tools known available find
source
like many
without code Evergreen see
using development
software applications application librarians
access easy
library
used FIGURE information people
open
CAPTION time just
figure
search services Source site
Internet Open patrons also need
operating way free Firefox website OPAC
Greenstone projects
popular tool Koha another

Figure 5. Open source

While it is clear that a single developer can contribute to some extent to improve social inclusion and projects aimed at achieving it, it is really a team of developers who are aching to contribute who can produce significant benefit to achieve the goals for this kind of projects.

Figure 6. A social developer

For that to happen, it is required that a meeting point for all volunteers is designed so that the Open Source community of social developers can meet and share ideas with the ultimate purpose of achieving digital inclusion, and where they can communicate with each other and coordinate their efforts.

Of course, this community should establish some working and behaviour rules in order to favour the pace of the community and to encourage the cooperation between all the members in the community.

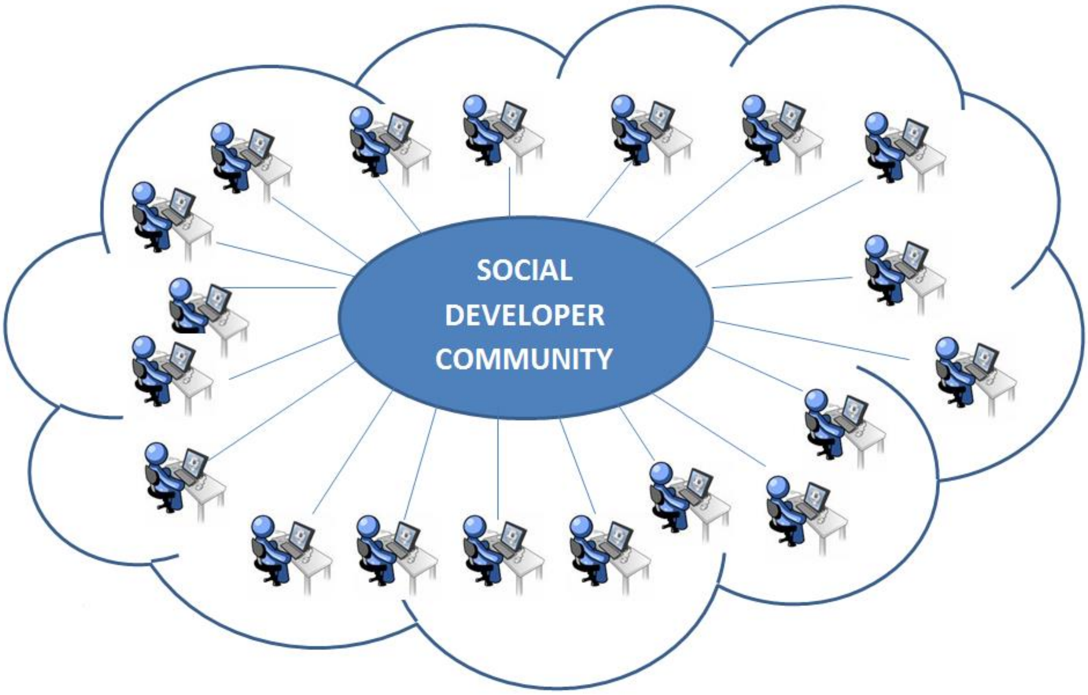


Figure 7. Social developer community

4.2 Actions

To create this community of social developers, a sequence of complex steps should be taken, which require technical, legal and promotional efforts. In a first phase, the next step could be enumerated:

1. Launching a website illustrating the concept of a social developer and his objectives.

2. Publishing software tools under opensource licence and a "social project" or "social developer" seal. The first projects could involve some of the tools of the ecosystem, and already existing opensource projects could be invited to take part.

3. Promote the concept and projects in social networks.

4. Define a procedure to evaluate projects in order to obtain the "social project" or "social developer" seal.