PROJECTE DE FI DE CARRERA
**Overview**

Today, working in an open and collaborative skills is one of the most demanded in the ICT world to which we are oriented. To this end we believe is essential to have tools and technologies that facilitate and encourage this ability to develop our capabilities in an appropriate manner.

The purpose of this project is to design and implement a tool to facilitate group work within the different courses and subjects of the EPSC. The basic idea is to establish a social network formed only by users of the EPSC, which can create groups of subjects managed with by the teacher.

This would define different user profiles (student, teacher, guest ...) each having special characteristics and functions different from the others, providing them with more or less depending on the profile permissions.

The information would also be subject to different types of view, depending on which users could see a group of a subject, all users ... being a common forum for both general events (demonstrations, lectures, open tables ...) as a way to share information and discoveries while working with your group, as your class or around the world.

The initial idea is to implement the use of freeware tools and technologies such as Tomcat, My SQL Server, Spring, Struts, Hibernate, J2ee, jquery-ui, etc,. To conduct a pilot project, and if it works, it could generate a stronger infrastructure in the future.
<table>
<thead>
<tr>
<th>INDEX</th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>INTRODUCTION..............................................................................</td>
<td>1</td>
</tr>
<tr>
<td>Introduction.............................................................................</td>
<td>1</td>
</tr>
<tr>
<td>Social Networks History ................................................................</td>
<td>1</td>
</tr>
<tr>
<td>Memory Organization....................................................................</td>
<td>4</td>
</tr>
<tr>
<td>CHAPTER 1. EPSCCOMMUNITY OBJECTIVES....................................</td>
<td>6</td>
</tr>
<tr>
<td>1.1. Existing Social Network................................................</td>
<td>6</td>
</tr>
<tr>
<td>1.2. EPSCCommunity Definition..............................................</td>
<td>7</td>
</tr>
<tr>
<td>1.2.1. Users ...........................................................................</td>
<td>8</td>
</tr>
<tr>
<td>1.2.2. Tools ..........................................................................</td>
<td>9</td>
</tr>
<tr>
<td>1.3. Benefits provided ......................................................</td>
<td>10</td>
</tr>
<tr>
<td>CHAPTER 2. TECHNOLOGIES AND TOOLS CHOSED.............................</td>
<td>11</td>
</tr>
<tr>
<td>2.1. Introduction.................................................................</td>
<td>11</td>
</tr>
<tr>
<td>2.2. Presentation layer technologies (Struts and Jquery-ui).........</td>
<td>11</td>
</tr>
<tr>
<td>2.2.1. Jquery-ui ....................................................................</td>
<td>11</td>
</tr>
<tr>
<td>2.2.2. Struts .......................................................................</td>
<td>14</td>
</tr>
<tr>
<td>2.3. Logic layer technologies (Spring)....................................</td>
<td>16</td>
</tr>
<tr>
<td>2.4. Data layer technologies (Hibernate)...................................</td>
<td>18</td>
</tr>
<tr>
<td>2.5. Integration technologies (Maven and SVN)..........................</td>
<td>21</td>
</tr>
<tr>
<td>2.5.1. Maven ......................................................................</td>
<td>21</td>
</tr>
<tr>
<td>2.5.2. Subversion (SVN) .....................................................</td>
<td>24</td>
</tr>
<tr>
<td>2.6. Implementation unit tests technologies (JUnit) ...................</td>
<td>24</td>
</tr>
<tr>
<td>CHAPTER 3. ARCHITECTURE AND APPLICATION DESIGN ....................</td>
<td>26</td>
</tr>
<tr>
<td>3.1. Introduction.......................................................................</td>
<td>26</td>
</tr>
<tr>
<td>3.2. Funcional requirements..................................................</td>
<td>26</td>
</tr>
<tr>
<td>3.2.1. Introduction ............................................................</td>
<td>26</td>
</tr>
<tr>
<td>3.2.2. Global Class Diagram..................................................</td>
<td>26</td>
</tr>
<tr>
<td>3.2.3. Actors .......................................................................</td>
<td>27</td>
</tr>
<tr>
<td>3.2.4. Use Cases ...................................................................</td>
<td>31</td>
</tr>
<tr>
<td>3.2.5. Principal functions ....................................................</td>
<td>36</td>
</tr>
<tr>
<td>3.3. Architecture.....................................................................</td>
<td>40</td>
</tr>
<tr>
<td>3.3.1. Software architecture..................................................</td>
<td>40</td>
</tr>
<tr>
<td>3.3.2. Data architecture .....................................................</td>
<td>42</td>
</tr>
<tr>
<td>CHAPTER 4. IMPLEMENTATION....................................................</td>
<td>45</td>
</tr>
</tbody>
</table>
4.1. User Interface

4.1.1. System Administrator

4.1.2. Group Administrator

4.1.3. Student and Guest Users

4.1.4. Common Actions

4.2. Not Implemented functions

4.2.1. Asynchronous Chat

4.2.2. Delete group

4.2.3. Delete content by Student and Guest user

4.2.4. Modify content

CHAPTER 5. TESTING

5.1. Unitary Test

5.2. Stress Test

5.2.1. Introduction. What is a Stress Test?

5.2.2. EPSCCommunity Stress Test

5.3. Usability testing

5.3.1. Introduction

5.3.2. Usability Test

5.3.3. Improvement Points

CHAPTER 6. CONCLUSIONS

6.1. Environmental impact

6.2. Objectives achieved

6.3. Future enhancements

6.4. Technologies conclusions

6.5. Personal conclusions

CHAPTER 7. BIBLIOGRAPHY
INTRODUCTION

Introduction

In a world where information is one of the most quoted, and in which the great discoveries are based on knowledge sharing and collaborative work is important to have tools that facilitate this new way of working.

Thanks to new technologies people are much more aware of everything, it's much easier to obtain information on any type of issue but also becomes hard to be expert at something when the contents are changing.

Companies seeking more each day a profile of people who can work in groups and to provide a range of knowledge, but instead a profile of a technology expert profile is much less desired, since this person will consume a number of economic resources much more important.

A person who can work together, within a working group with a tool to facilitate their collaboration is much more efficient than a person who is an expert in something and have to develop it alone, the same group's work.

We, as students believe that currently the Moodle does not provide support that encourages collaboration among its users and is therefore in the end a working group of a subject always ends up creating a Google group, an FTP, a mailing list ...

That is why it is important to encourage this type of skill and therefore believe it is vital to provide the technologies required or at least know which of them might be a good example to use.

Within this world of collaborative work, we find supports existing social networks in which people collaborate and share information, usually associated with leisure and recreation as Facebook, Twitter, Nettby or other much more focused on the workplace as LinkedIn.

Social Networks History

In this section we report briefly what is known as social network and how and when they began to appear
A social network is a structure of nodes and links between them. In our case, at the social networks, each node would represent a member of the network and the bond provide that type of relationship is established between both. In this way, we could identify in a concise and easy form how to get from one node to another and what kind of relationship would be established to add a link.

![Social Network example](image)

**Fig.1. Social Network example.**

This social network would consist of nine nodes or members, which does not link all in all, but including establishing one or more relationships.

Although it seems that the phenomenon of social networking is something really new, it is not. Today the proliferation of these is very large due to the emergence of some of them as Facebook, which has no professional or specific purpose, but is designed to common Internet user, which is not having a specific profile to be part of it.

The great success of such networks has increased the number of users, but look like they were the origins.

The first social network dates from 1997, called SixDegrees. It gave its users the ability to create profiles and friend lists. Since 1997-2001, the “model” began to grow, and three sites (AsianAvenue, BlackPlanet and MiGente) allowed users to meet people and review profiles friends without seeking approval (something “similar but opposite” to Facebook).

The following figure shows the chronology of the emergence of social networks until 2006:
Within this wide range of social networks, can be distinguished by innovation introduced at the time of occurrence as follows:

- **Ryze** to help people work, allowing the search for jobs or business connections. Today, **LinkedIn** is the largest site in this style.

- **MySpace**, which was created in 2003, and in the beginning, was different from other pages to allow users to customize their personal pages. MySpace was a great success especially among teens in its infancy, but does not allow underage users.

- **Facebook**, as many know, was created in the beginning to support the university networks. In 2004, users had to provide email addresses used in the university, so in the beginning it was a very closed model. Then he began to expand to, in principle, include high school students, professionals and then opening the game to all Internet users.

- **Twitter** also revolutionized the social networking market, from a very clear message: "What are you doing?" Users have to share links or talk to each other as if it were text messages of 140 characters. There are
many applications and tools that, one way or another, are supplemented by the service and, in some cases, provide some interesting features.

One of the main characteristics of social networks is the sharing of personal information of users between them. Without going any further, many Facebook users complete their profiles with the direction of their homes, their personal relationships, tastes, moods and even the occasional presence at the event.

This causes many users wary of the treatment given to their data, since in many cases the social network policy is unclear or do not exist. If we focus on the case of Facebook, we find that recently had to reconsider this point because so far his policy was clear:

"You may not claim any right of original content that users upload to the network, once users have closed their accounts"

This resulted in many complaints from users until they finally decided to turn 180 degrees on it.

Still, after this change of policy, this issue is causing much controversy in the social networking world.

**Memory Organization**

The completion of this project presents us with a number of objectives:

1. Study the existing technologies.

2. To investigate whether someone continues the use of social networks and that results have been obtained.

3. To assess whether any existing technology is capable of supporting our idea.

4. In case of not finding any technology, develop a customized application with social networking concepts.

5. Finally, try to deploy this application in a university environment and study the degree satisfaction and use by the user to see if it is a useful tool or not.

The following details how is organized the project memory.
The first chapter provides a brief description of existing social networks, to then go on to define the objectives of EPSCCommunity and the benefits that we believe will contribute to the university environment.

In the second chapter talk about the technologies and Tools chosen, giving reasons for which have been chosen.

The third chapter describes the architecture and application design, based on the application requirements, types of users, tools, and software and data architecture used for development.

In the fourth chapter talk about the implementation and user interface, describing all the features offered by EPSCCommunity.

The following chapter describes the stress and usability tests performed and the conclusions drawn from the tests results.

Finally, with an overview of the product, provides a balance of the objectives achieved and possible future improvements could be made. It is also an evaluation of the technologies chose and the personal conclusions about the project perform
CHAPTER 1. EPSCCommunity Objectives

1.1. Existing Social Network

The objective of this project is to design, from scratch, a social network focused on the needs of students and faculty of a university, providing facilities for communication and obtaining information. In this way they aim to improve outcome in the studies and research.

From the first moment that this idea arises, search other existing applications, which provide functions similar or identical to what we wanted to offer.

We found diversity social networks, including some focused on teaching field, as is the case with this project.

This social network is born with the same idea that our project is “Tic Tac” (Information Technology and Communication) (Also Friends and Colleagues).

Tic Tac intended as a place of networking, professional (TIC) or informal (TAC), among students, former students, teachers, businesses, professionals, friends and colleagues.

Most notable, according to its creator, comparing this social network, with existing ones, is that the goal is to be a social network 3 in 1, fully immersed in Web 2.0 technologies (Triangle 2.0). It is the union of three components of Web 2.0 that are linked to their actors:

- Friends Social network
- Teaching platform
- Network of professional relationships

It now consists of 1068 participants.

It has also seen the use of social networks in a high school education, it used the social network Ning, which offers the possibility of creating a social network itself, with 10GB of space and 20,000 members, free of charge, and without installing software.

The most positive part of creating this social network is that students have increased their interest in the subjects.

The value and social significance of university communities is determined by the need to share and improve knowledge.
The social networking phenomenon and its rapid growth requires analysis at the university level. A part of the success as a social phenomenon, the development of these networks is very useful tools to promote its application in college should not portray.

We must learn and import ideas or interesting tools for teaching and research. Otherwise arrive at a situation in which society in general use more powerful tools used by the university.

Not is “Facebook” or other similar social networks the benchmark for to evaluate whether such tools are useful, but rather belong to the realm of what one might call "social phenomenon."

The idea is to form a specialized social network, to facilitate student learning with greater communication among them, teachers and researchers.

1.2. **EPSCCommunity Definition**

EPSCCommunity social network we are proposing is based on a network to obtain information on study-related content that is taught in the EPSC.

The basic idea is to create a community that can share information related to the courses, examinations, exercises, projects ... making learning not only takes place in class or working or studying in the library, but this can done through direct contact with people in the community, and making all users on teachers and students.

The concept of EPSCComunity would describe as a set of groups (subjects) in which people can subscribe. Here enter the different types of users might have:

- User who already taught the course and are interested in further part of the agenda.
- A user enrolled in the course.
- A user who does not know the subject but want to know how it develops.

Besides these groups, we could define other within the first would try to support the needs of the working groups within a subject.

The idea is that information of the groups have different permission levels, there will be information that only registered users can read the group, one that can read the rest and one that perhaps can only be read by the teacher.
This requires us to establish a set of permissions which make different profiles of users can access information or otherwise.

For example, a registered user could access information on the notes of an examination while a user is registered (with profile guest) perhaps could only access content such as items to be discussed at the next class, the topic of presentations or just access to documentation that provides the teacher.

The application also has an array of tools to help students improve communication within the group, to contribute ideas to the pace of class, to have direct communication with the teacher to resolve doubts or ask questions on the topic of classes outside school hours.

1.2.1. Users

In this section we detail the different types of users can use the application, detailing their main functions on the different aspects which interact in the application (groups, users and content). So let us look different users:

1.2.1.1. System Administrator

This user can have all the information needed to manage the smooth functioning of the social network. It may decide to view all contents, access to all groups (not the user profiles in which your information is confidential).

1.2.1.2. Group Administrator

The Group Administrator user (teacher) will be responsible for managing the information about the group (subject) from which it is responsible.

1.2.1.3. Student User

The student user is one that is registered for the course, and as such will have more privileges than he who is not. The student user can access both public and private information, while unregistered users can only access information that is public.
1.2.1.4. **Guest User**

This member would represent all the people who want to be informed of the progress of the subject without being registered. For example, a person who has already completed, but is interested in being able to attend any session they could not attend or want to deepen some aspect of it. This user is the one with the most restrictive permissions.

1.2.2. **Tools**

In this section we detail the services which will provide in the application. This project aims to improve the interaction of different users who may be interested in the subjects, so it will be very important to emphasize all sorts of services that improve communication between them.

1.2.2.1. **Instant Messaging**

Allows two users at the same time, they are in the same list of contacts, establish communications.

1.2.2.2. **Private Messaging**

Allows a user to communicate in a non-snapshot with another user on your contact list, which may answer at any time.

1.2.2.3. **Forum**

Allows for a historical account of views and information of users differentiated by subject.

1.2.2.4. **Email**

This feature can receive and send emails to any member of the social network to maintain communication.
1.2.2.5. **Content Repository**

Repository with a limit capacity, which allow users to share files.

1.2.2.6. **Groups and Profiles Configuration**

Each user can modify his user profile and each group administrator can configure the profile of the group he manages.

1.2.2.7. **Calendar / Events**

This option allows users to have control over the upcoming events in the classroom; topics, test dates, deadlines, etc.,

### 1.3. **Benefits provided**

The application proposal, EPSCCommunity, provides the functionality described above. Opening the mind to success use of social networks, our application provides benefits at the university level to be mentioned.

With the use of social network, as EPSCCommunity, can be achieved increased sharing of knowledge, dissemination of activities and work, the discussion of problems, increasing interest and collaboration of students, etc., All this by bringing together the university students and former university students who share the same interest in teaching, research, studies, subjects, etc.. Students, teachers and researchers can identify those members who affinities, shared interests, groups, the fact they can contact each other to provide significant advantages.

In this network, teachers and researchers can expose drafts of research papers, share discussions on teaching approaches, resolve doubts without the need to agree an appointment, discuss a particular topic, share documents, etc.,

Students can interact in the same way they do in other social Networks, while giving greater prominence to academic issues and benefit from the presence of teachers in the networking and exchange of knowledge and documents with other network members.
CHAPTER 2. Technologies and Tools chosed

2.1. Introduction

In this chapter we will explain the technologies which have been chosen for the development of EPSCCommunity, discussing their use in each of the three layers in which this split application (presentation, business logic and data access). In turn, try to give an overview on the main aspects that we have decided to use them.

2.2. Presentation layer technologies (Struts and Jquery-ui)

To develop the presentation layer have been used two different tools, one for the client side (Jquery-ui) and another to manage the web-flow application (Struts). Then we will discuss more about them:

2.2.1. Jquery-ui

Jquery-ui is a development framework based on Javascript. This technology provides us with methods and functions that enable the presentation that displays the end user in browser is very appealing and interactive with animations and effects.

This concept of some static web applications is based on the concept of RIA application (Rich Internet Applications), which consist in providing web applications with certain characteristics that only are attributed to the desktop applications, such as moving to deploy components of site, display listings, etc.

For the development of EPSCCommunity we used a lot of components of Jquery-ui:

- Datepicker: We provide a timetable when inserting a date
• Accordion: allows to display a list of objects and navigate through them in accordion form, displaying one and hiding another.

Fig. 2.1. Datepicker user Example

Fig. 2.2 Accordion use Example

• Autocomplete: allows search and autocomplete words of a preset dictionary

Fig. 2.3. Autocomplete use Example
- Dialog: gives us the opportunity to open a modal popup that can be used to display an error or to show a form to insert data.

![Dialog use Example](image)

**Fig. 2.4.** Dialog use Example

- Tabs: its usefulness is to structure the contents on the screen with different tabs, and manage the data load (AJAX) and navigation between them.

![Tabs use Example](image)

**Fig. 2.5** Tabs use Example

- Draggable: gives us the ability to move items around the screen, and define what are the areas where it can move.
For this task there are another technologies like Flex, or the opensource version OpenLazslo or GraniteDS, but finally we have opted by Jquery-ui for its power and simplicity, in addition to requiring little computational load in the browser.

In this section we have seen a small summary of what is Jquery-ui and some of its components used. Let's see what is Struts:

2.2.2. Struts

Struts is one of the frameworks MVC (Model View Controller) or three layers oriented model more elaborate and with more history. This gives us the ability to serve dynamic Web pages and connect with other application layers, as the business logic to process data and the data layer to access the database.
The navigation flow between pages is defined in a configuration file called struts-config.xml, which defines the various actions that exist in the application. From here, these actions are implemented by java classes that are responsible for deciding if the flow goes to a one page or to another. By example:

In the login page, at the form we configure the action that it want to run when we do the submit.

```html
<html:form styleId="loginActionForm" action="/loginAction.do?option=login">
<input type="submit" value="Login"/>
</html:form>
```

**Fig. 2.8** Struts form example

The *option* parameter we use because the loginAction has several actions configured, and thus we can specify the action within the class.

As we see the url ends with the extension .do, this is to differentiate that this type of URLs will serve the Struts servlet will based on the configuration of struts-config.xml

```xml
<action-mappings>
   <action path="/loginAction" type="es.upc.epsc.web.action.LoginAction"
           name="loginForm" scope="request" parameter="option">
      <forward name="success-user" path="/jsp/main/mainpage.jsp"/>
      <forward name="fail" path="/jsp/login/bad-user.jsp"/>
   </action>
</action-mappings>
```

**Fig. 2.9** Struts-config.xml configuration example

As we see, the action loginAction is implemented at the es.upc.epsc.web.action.LoginAction class. In our case not is it exactly because we delegate this functionality to Spring.

Following, the loginAction example:
As you can see, if the user is correct it redirects to "success-user". Otherwise to "fail." If we return to see the image 2.10 will see as success-user leads to /jsp/main/mainpage.jsp making us enter into the application and otherwise, fail, shows the page /jsp/login/bad-user.jsp, indicating an error in the validation process of login and password.

As we have seen, Struts combines simplicity when developing and the robustness of working with a model of layers. These features have made us opt for Struts over other alternatives such as self web-flow of Spring.

2.3. Logic layer technologies (Spring)

In the three-layer model, our next main character is the business logic layer. For this next level we have chosen Spring. This technology provides structure and relationships between the different elements of the application, making optimal performance. Internally built an architecture based on design pattern singgletton getting eliminate instances of unnecessary items. Thus, we can ensure that memory consumption will be optimal.

Let's see how it works:

As discussed in the previous section, we have delegated the use of the actions by Struts to Spring. This we have achieved changing the configuration in the struts-config.xml

```java
public class LoginAction extends DispatchAction {
    public ActionForward login(ActionMapping mapping, ActionForm form,
        HttpServletRequest req, HttpServletResponse res) throws Exception {
        boolean userOK=true;
        /* Check if user is valid and put true
        * or false in userOK variable
        */
        if (userOK) {
            return mapping.findForward("success-user");
        } else {
            return mapping.findForward("fail");
        }
    }
}
```

Fig. 2.10. Action Example
Technologies and Tools chosen

Adding this plug-in and changing the attribute type of the action-mapping get action settings EPSCCommunity-config.xml.

```xml
<bean id="loginAction" class="es.upc.epsc.web.action.LoginAction">
  <property name="userManager">
    <ref bean="userManager"/>
  </property>
</bean>
```

**Fig. 2.12** loginAction with Spring configuration

By the previous configuration setting that we are creating the instance of the application and load the Spring context we have to create an object loginAction that containing an object userManager. This object userManager contain all the business logic required to perform the operations required by the presentation layer.

Let's see how is composed of:

```xml
<bean id="userManager" class="es.upc.epsc.integration.manager.impl.UserManagerImpl">
  <property name="userDAO" ref="userDAO" />
</bean>
```

**Fig. 2.13** userManager configuration

The “bean” of Spring userManager tan comprises only by other bean userDAO, responsible for data access and managed by Hibernate, as shown below:

```xml
<bean id="userDAO" class="es.upc.epsc.integration.dao.impl.UserDACImpl">
  <property name="sessionFactory" ref="sessionFactory" />
</bean>
```

**Fig. 2.14** userDAO configuration

The object that composes, sessionFactory, is part of Hibernate, so we shall see in detail in the next chapter.
As we see, Spring is useful because it structure very well the three levels of application, managing the use of memory server, making it unnecessary that the user instantiate any object type of business or data access.

2.4. Data layer technologies (Hibernate)

To develop this last application layer we have choose Hibernate. This framework is the core of our software, capable of maintaining the absolute integrity of the object model versus the entity relation database model.

Hibernate abstracts developers from having any kind of intervention on the data directly, only acts on objects that define our logical model.

To do this, by setting the sessionFactory object mentioned in the previous section we can map our logic model classes to tables and Hibernate will handle both create scripts and run them as to access or modify data. Let's see how:

```xml
<property name="annotatedClasses">
  <list>
    <!-- Mapeo de objetos para crear la BBDD -->
    <value>cs.upc.epsc.dto.Group</value>
    <value>es.upc.epsc.dto.User</value>
    <value>cs.upc.epsc.dto.Document</value>
    <value>es.upc.epsc.dto.Event</value>
    <value>es.upc.epsc.dto.Forum</value>
    <value>es.upc.epsc.dto.Message</value>
    <value>es.upc.epsc.dto.Topic</value>
    <value>es.upc.epsc.dto.UserGroup</value>
    <value>es.upc.epsc.dto.MailBox</value>
    <value>es.upc.epsc.dto.InBox</value>
    <value>es.upc.epsc.dto.OutBox</value>
    <value>es.upc.epsc.dto.Email</value>
  </list>
</property>
```

![Fig. 2.15 Hibernate object mapping](image)

With this configuration we are indicating what will be our logical model, and we specify sets the fields that are table columns or attributes that establish the relationships between them in the class java code. Here is a simple class
Fig. 2.16 How to establish relations and columns with hibernate annotations

Using annotations we can specify:

- **@Entity**: This class is an entity that is part of our logical model.
- **@Table**: indicates that this class must be mapped in a database table:
  - Name: the name of the table
  - Catalog: schema name of the database we will use.
- **@Column**: indicates that the value in this variable must be saved in a column of the table.
- **@OneToMany**: with this annotation we can specify the relation between this entity with another one in the Project. In this case the relation is one to many
  - Cascade: this attribute indicates what’s the behaviour of the relations of the two entities. For example, if we delete an object that contains a list of other objects and we set CascadeType.ALL this also deleted all contained entities.
  - Fetch: this parameter shows how is getting the object or objects that define the relations. For example, if we set FetchType.LAZY we have to obtain the objects from the database. Otherwise, if we set FetchType.EAGER Hibernate will do that.
  - MappedBy: establish witch is the attribute that defines the relations in the other class of the relation.
- There are other relations like **@ManyToMany**, **@ManyToOne** o **@OneToOne**.
Now is shown how to configure the access to the data base throw a data source.

```xml
<bean id="dataSource"
     class="org.apache.commons.dbcp.BasicDataSource"
     destroy-method="close">
    <property name="driverClassName">
        <value>com.mysql.jdbc.Driver</value>
    </property>
    <property name="url">
        <value>jdbc:mysql://localhost/epsccommunity</value>
    </property>
    <property name="username">
        <value>root</value>
    </property>
    <property name="password">
        <value>root</value>
    </property>
    <property name="initialSize" value="0" />
    <property name="maxActive" value="10" />
    <property name="minIdle" value="2" />
</bean>

Fig. 2.17 Data source configuration

In this configuration we specify which is the driver that we are going to use, the URL of the data base and the user and password.

Finally, we join the two parts of the configuration to obtain the XML configuration file-

```xml
<bean id="sessionFactory"
     class="org.springframework.orm.hibernate3.annotation.AnnotationSessionFactoryBean">
    <property name="dataSource">
        <ref bean="dataSource" />
    </property>
    <property name="hibernateProperties">
        <props>
            <prop key="hibernate.dialect"/>
            org.hibernate.dialect.MySQLDialect
            <prop key="hibernate.hbm2ddl.auto">update</prop>
        </props>
    </property>
</bean>

Fig. 2.18 Hibernate start behaviour

In the previous image is shown that we are going to use the previous data source, hibernate with annotations and the behaviour of the reboot action, in
this case update the tables structure of the database, in case of there is any modifications in the relations between classes.

In summary, we can ensure that Hibernate is a technology that totally abstracts the data layer, making the developer capable to work only with the other layers of the application. In this sense, there are other frameworks like iBatis, able to perform the same functions, but Hibernate is the most powerful framework and easier to use, not to configure.

2.5. Integration technologies (Maven and SVN)

2.5.1. Maven

In a software development project is necessary to have a tool that is able to simplify the integration of the different modules of the application.

For this purpose we choose MAVEN, a tool able to:

- Download the project from a repository (SVN)
- Compile the project
- Execute unitary test (JUNIT)
- Install it
- Download the last updates of the application dependencies.

All this actions are totally automatic and transparent for the Developer after MAVEN configuration

The MAVEN configuration depends only from one file, the pom.xml (Project Object Management). This file contains all the necessary information about the instructions for compile, install and download dependencies. For example:

In the next image it is shown the actions we configure previous to an application installation:

```xml
<executions>
  <execution>
    <id>undeploy-and-deploy</id>
    <phase>pre-integration-test</phase>
    <goals>
      <goal>deployer-undeploy</goal>
      <goal>deployer-deploy</goal>
    </goals>
  </execution>
</executions>

Fig. 2.19 MAVEN deploy configuration
We can see that the first is to execute the unitary test, if they are ok next step is to undeploy the previous installation and then deploy the actual revision of the application.

Now we are going to show how is the configurations to execute the unitary test:

```xml
<plugin>
  <groupId>org.apache.maven.plugins</groupId>
  <artifactId>maven-surefire-plugin</artifactId>
  <executions>
    <execution>
      <id>it-test</id>
      <phase>pre-integration-test</phase>
      <goals>
        <goal>test</goal>
      </goals>
      <configuration>
        <includes>
          <include>/**/*Test.class</include>
        </includes>
      </configuration>
    </execution>
  </executions>
</plugin>
```

**Fig. 2.20** Maven config of automatic JUNIT testing

In image 2.20 we can see that MAVEN will look for the classes with the pattern name *Test to execute them.

Next we can see the MAVEN configuration to deploy the project:
Here the configuration is so descriptive, we configure Tomcat 6.x indicating maven the URL of the manager to deploy or undeploy the application (http://localhost:8080/manager) and the user with privileges to do this, in this case user tomcat password tomcat.

Finally, there is the configuration of one repository:

As we can see, MAVEN will look for new actualizations in the configuration URL before compiling the application. In case of find it MAVEN will download and install it.

MAVEN is a very useful tool that brings us all the tasks necessary to perform a deployment, a fairly costly than simply as an instruction that is executed by the developer.
There are some tools with the same utility, i.e. ANT, but MAVEN is more powerful than them, but rather more difficult to configure. It requires a very labour costs in its first configuration, but once running its maintenance has no cost, not like ANT, which is composed by some scripts that could grow with the project.

### 2.5.2. Subversion (SVN)

Another tool needed to work in a developers team is a source code repository. This is due to everyone has to work with latest version of the code and sometimes many people need work with the same code and something has to organize the possible conflicts.

Subversion is a repository server. Allows you to manage different revisions of a file, to view their historical or manage conflicts of users who have modified the same code.

The subversion server can be installed on a computer as a Windows or Linux service or as an independent daemon, making it accessible to everyone who sees this PC.

There are several alternatives, such as Visual SourceSafe (VSS), although we decided for SVN mainly because it is OpenSource solution and it integrates nicely with eclipse (our development IDE). VSS also not allowed to work concurrently with the same source file to different users.

### 2.6. Implementation unit tests technologies (JUnit)

JUnit is a testing framework for Java applications. JUnit is capable of running very specific code test of the application, being able to decide if the test was successful or not.

Let’s see a simply example about how to find a document using its primary key:

```java
@RunWith(SpringJUnit4ClassRunner.class)
@ContextConfiguration(locations = "file:src/main/resources/spring/dao-config.xml")
public class DocumentDIOTest {
    
    // Code for testing

    Fig. 2.23 Spring context load

First of all configure the environment. The image above shows how to specify that JUNIT will run in Spring context using their inner classes, and first, before launching the test will need to load the context of the application (in this case only charged the context layer data).```
Fig. 2.24 Test main code

The previous image shows how is the source of the test. @Test indicates that the next code has to be executed as a test. Otherwise, the annotation has to be @Ignore.

As we see, the code creates one document, one user and one group, required entities to maintain the persistence of data in the database. Then the document is assigned to the group that owns the document and the user who created the document store. Then we look for it using its identifier. If the search result is null (assertNotNull Tested with) the test will have failed, otherwise the test worked properly and will proceed to delete the document, the user and the group from the database.

There are not many alternatives to JUnit, one possibility is to run the test an independent main, but doing so would lose the seamless integration of JUnit with Spring and MAVEN.
CHAPTER 3. Architecture and application design

3.1. Introduction

This chapter will detail the application functional requirements and software and data architecture design. This is one of the most important steps in developing the application. The architecture design and the requirements should be very clear and detailed, with the aim of allowing deploy the application.

3.2. Funcional requirements

3.2.1. Introduction

One of the first steps in designing an application is the specification of requirements. In this process, which will be assessing the needs of each type of user you will have access to the application and content.

It should define user profiles depending on the information they can access and the actions they can perform.

The following describes the global class diagram with the relationship between the different objects of the social network and the different types of users who can access the application and limitations of each.

3.2.2. Global Class Diagram

The class diagram shown below represents the structure of each system objects and their relationships with other objects.
3.2.3. Actors

3.2.3.1. System Administrator

The system administrator functions will be:

• About the groups:
- Create groups of subjects, and which will assign the user who will manage the group. For example, could create the kind of XSS group and assign the teacher of the subject as the group leader.

- Modify any information concerning the configuration of the groups.

- Delete groups
  
  • *On users:*
  
  - It can assign as responsible for the different groups of subjects (in this case only teachers)
  
  - you can kick users (any profile) to conduct a malicious use of the system.
  
  - Will be in charge of the profiles and change user privileges.
  
  - It has the ability to send and receive e-mail type messages to all users in the group to maintain contact to resolve doubts as to notify changes.

  • *About the contents:*
  
  - The system administrator can create any content of the system (events, documents and topics of forum).
  
  - The user administrator can delete of the system any content that is not appropriate for the purpose of the social network, in this case, the educational aim.

3.2.3.2. *Group Administrator*

The teacher user will be responsible for managing the information about the group (subject) from which it is responsible. Its detailed functions are:

• *About the Groups:*

  - It can modify the properties of groups (name, configuration ...)

• *On Users:*

  -
- Will be responsible for students assigned to users within the groups.
- It can modify the members of the groups (add or delete members).
- It can notify the administrator user misuse of any user (not to expel, task of the administrator who will decide).
- It has the ability to send and receive e-mail type messages to all users in the group to maintain contact to resolve doubts as to notify changes.

• About the Contents:
- It can display the contents within the groups
- The group administrator can create any content of the group (events, documents and topics of forum).
- It can delete any content of the group that it is administrator
- Decide the viewing permissions for each content (if you can see students only or if you can see all users in the group)

3.2.3.3. Student User

The student user is one that is registered for the course, and as such will have more privileges than he who is not. The student user can for example belong to a group project, while unregistered users can only access information that is public. Here are the features that have:

• On Users:
- It can take online communication with users connected at that instant by instant messaging service, both as his project team in their group of subjects.

• About the Contents:
- The user Student can create any content of the group (events, documents and topics of forum).
- It will have the ability to view both public and private content of the group.
- It has the ability to send and receive e-mail type messages to all users in the group.

3.2.3.4. **Guest User**

This member would represent all the people who want to be informed of the progress of the subject without being registered. For example, a person who has already completed, but is interested in being able to attend any session they could not attend or want to deepen some aspect of it. This user is the one with the most restrictive permissions.

Its functions are:

- **About the Groups:**
  - It will have visibility of the groups.
  - It may not modify any aspect of the configuration of the subject groups in which it is registered (not registered)

- **On Users:**
  - It can post in the public forum of the subject to interact with other users.
  - It have instant messaging service.

- **About the Contents:**
  - It can visualize all only the public content supplied by the members of the course (including the teacher).
  - The *Guest* user can create any content of the group (events, documents and topics of forum).
  - It has the ability to send and receive e-mail type messages to all users in the group.
3.2.4. Use Cases

Then, are defined the functions of each of the actors that have been defined above, with a use case diagram.
3.2.4.1. System Administrator Case Use

Fig. 3.2 System Administrator User Case Use
3.2.4.2. Group Administrator Case Use

Fig. 3.3 Group Administrator User Case Use
3.2.4.3. Student Case Use

Fig. 3.4 Student User Case Use
3.2.4.4. **Guest Case Use**

![Guest User Case Use Diagram](image)

**Fig. 3.5** Guest User Case Use
3.2.5. Principal functions

Here are briefly the main features of the application, defining its purpose and the actor who can carry it out.

3.2.5.1. Group Management

Create Group

Use Case: Create Group
Actor: System Administrator
Intent: The System administrator user introduce the data of a new group that wants to enlist in the network. The system registred a new group and associate it with the administrator user.

Table.3.1. Create Group Action

<table>
<thead>
<tr>
<th>Actor actions</th>
<th>System actions</th>
</tr>
</thead>
<tbody>
<tr>
<td>1. The administrator user introduce the data of a new group.</td>
<td>2. The system validate the data.</td>
</tr>
<tr>
<td></td>
<td>3. The systems registred a new group</td>
</tr>
<tr>
<td></td>
<td>4. The system associate a new group with the administrator user.</td>
</tr>
</tbody>
</table>

Delete Group

Use Case: Delete Group
Actor: Administrator
Intent: The administrator user can unssubscribe a group of the network. The system removes the group published by a particular user.

Table.3.2. Delete Group Action

<table>
<thead>
<tr>
<th>Actor actions</th>
<th>System actions</th>
</tr>
</thead>
<tbody>
<tr>
<td>1. The administrator user introduce the data of group that wants delete.</td>
<td>2. The system validates that the group exists.</td>
</tr>
<tr>
<td></td>
<td>3. The system delete the group of the network.</td>
</tr>
</tbody>
</table>
Modify Group

Use Case: Modify Group
Actor: Group administrator
Intent: The Group Administrator user can edit and update the properties of the group that previously had published on the network. The user modifies the data and the system will record and display the new group information.

Table 3.3. Modify Group Action

<table>
<thead>
<tr>
<th>Actor actions</th>
<th>System actions</th>
</tr>
</thead>
<tbody>
<tr>
<td>1. The teacher user introduce the data of the group whose characteristics he want to modify.</td>
<td>2. The system validates that the group exists.</td>
</tr>
<tr>
<td>3. The user changes the data that are interests for it.</td>
<td>4. The system registered data changed by user.</td>
</tr>
<tr>
<td></td>
<td>5. The system displays the new group information.</td>
</tr>
</tbody>
</table>

3.2.5.2. Content Group Management

Add content

Use Case: Add Content
Actor: All users
Intent: A user who belongs to a group can add content in his group. Get a call new object to the system. The system adds the new object of the user and displays it.

Table 3.4. Add Content Action

<table>
<thead>
<tr>
<th>Actor actions</th>
<th>System actions</th>
</tr>
</thead>
<tbody>
<tr>
<td>1. Users introduce new content of his group.</td>
<td>2. The systems collect the requests for a new object.</td>
</tr>
<tr>
<td></td>
<td>3. The system add the new object of the group.</td>
</tr>
</tbody>
</table>
Modify content

Use Case: Modify Content
Actor: Teacher and student
Intent: A user belonging to a group wants to update a content of his group, added by him. The system will get a request to update the object. The system updates the object and displays the modification.

Table.3.5. Modify Content Action

<table>
<thead>
<tr>
<th>Actor actions</th>
<th>System actions</th>
</tr>
</thead>
<tbody>
<tr>
<td>1. The user introduce data of he wants update at the content to created group by him.</td>
<td>2. The system collects the request to update an object.</td>
</tr>
<tr>
<td></td>
<td>3. The system update the object of group.</td>
</tr>
<tr>
<td></td>
<td>4. System displays the update object.</td>
</tr>
</tbody>
</table>

Delete content

Use Case: Delete Content
Actor: All users
Intent: User belonging to a Group wants to remove a content of his Group. User indicate to a system the object that he wants delete. System received request and delete the object associate to a user Group and reports the changes.

Table.3.6. Delete Content Action

<table>
<thead>
<tr>
<th>Actor actions</th>
<th>System actions</th>
</tr>
</thead>
<tbody>
<tr>
<td>1. User introduce content data that he wants delete of the system, and added by him.</td>
<td>2. System received request of delete object.</td>
</tr>
<tr>
<td></td>
<td>3. The system delete object associate to a user’s group.</td>
</tr>
<tr>
<td></td>
<td>4. The system reports the changes.</td>
</tr>
</tbody>
</table>
Send and receive mail

Use Case: Send or receive mail
Actor: All users
Intent: The user wants to contact another user on the network. The user tells the system you want to send a mail. The system receives the request and sends the message via email to indicate recipient user.

Table.3.7. Send and Receive Mail Action

<table>
<thead>
<tr>
<th>Actor actions</th>
<th>System actions</th>
</tr>
</thead>
<tbody>
<tr>
<td>1. User introduce the message that he wants send via mail.</td>
<td>2. System received request of send a mail.</td>
</tr>
<tr>
<td></td>
<td>3. The system send mail to indicate recipient user.</td>
</tr>
</tbody>
</table>

3.2.5.3. Profiles Management

Add permissions

Use Case: Add permissions
Actor: System Administrador and Group Administrator.
Intent: The System Administrator user, with privileged permissions, can get permissions to Group Administrator, and the Group Administrator (teacher) user can get permissions to a student and guest users. The system received new permissions and registered at the user profile.

Table.3.8. Add permissions Action

<table>
<thead>
<tr>
<th>Actor actions</th>
<th>System actions</th>
</tr>
</thead>
<tbody>
<tr>
<td>1. Administrator or teacher user introduce a new permissions that he wants added to a other user profile.</td>
<td>2. The system receive request to introduce a new permissions to user profile.</td>
</tr>
<tr>
<td></td>
<td>3. The system registered the new permissions at the user profile.</td>
</tr>
</tbody>
</table>
3.3. Architecture

3.3.1. Software architecture

The first step has to be done to develop software is to be clear which will be the same architecture. Be based on this architecture the foundations that could help bring the results are robust, scalable and modular, three very important factors in web applications.

The application is divided into three layers:

![Application layers diagram](image)

**Fig. 3.6 Application layers**

Each of them has very specific and different functions.

Let's see what are the main advantages that this architecture provides:

1. Define interfaces between them and able to conduct a completely independent development, provided that the defined interfaces.
It is possible to develop the presentation layer without having the business logic implemented as we can create "dummies" that we return static content, being able to perform a modular development.

2. Modules can change independently, for example, in our application the login is performed against the same database, we could not change the presentation layer and business logic, modify the data layer so that the login is against an LDAP.

3. To perform testing in different layers, being able to identify a much more clearly where the potential problem.

Once discussed the advantages offered by this architecture are detailed below the different functions performed by each one of them.

3.3.1.1. **Presentation layer**

The presentation layer is the user level. It provides the visual interface used by clients to view information and data. From this level are requested and received services by the business logic layer. Importantly, although the business logic is in another layer, this is transparent to the user.

The presentation layer can vary the privileges of each user type, which is controlled by the business logic layer.

For the presentation layer technologies have been chosen jquery-ui and struts, which as detailed in the previous chapter.

3.3.1.2. **Business logic layer**

The business logic is responsible for a bridge between the presentation layer and data layer. It's called business logic because it is where all the rules are set to be met. This layer communicates with the presentation layer to receive the requests and present the results, and communicates with the data layer to ask the database manager to store or retrieve data from it.

The business logic layer is the link between the data displayed on screen and the data stored in the database.

This layer is needed to prevent that the user has direct access to the database, which provides greater security.

The principal functions performed by this layer are:

- Receive requests from the presentation layer
- Execute routines
- Send data requests to the data layer
- Get the answers from the data layer
- Send the data to the presentation layer

The technology chosen for the business logic has been Spring, as detailed in the previous chapter.

3.3.1.3. Data layer

The data layer is responsible for implementing the requests received by the business logic layer against the database.

The actions you can take based on the received request are:
- Connection to the database
- Consult tables
- Create, modify and delete records

3.3.2. Data architecture

The following details how this formed the architecture of the data layer.

3.3.2.1. Database manager

The definition of a model for the data management is important when designing an application in order to provide a quality service for handling the data with which to work reliably, safely and ensure integrity and persistence of them.

We decided to use for our application MySQL server for the database manager. The main reasons for this choice is that easily interacts with other selected technologies, is easy to use, opensource and consider it powerful enough for the amount of data that are to handle.
3.3.2.2. Data Model

Was chosen for the design of our database Entity-Relationship data model, which is known as a relational database. This is the most widely used model because it allows greater efficiency, flexibility and confidence in the data processing.

To undertake the design of the database we used Hibernate. This technology allows us to accurately map the object model of our application to the model entity relationship database.

This gives us a precise data model and persistent, abstracting the programmer from all matters relating to access to data from the data layer, works directly with objects.

The reasons for which this technology has been chosen are explained in detail in the previous chapter.

The scheme that we have located the database is called epscccomunity and consists of 13 tables, each with a specific name:

![Database scheme](image)

**Fig. 3.7 Database scheme**

3.3.2.3. Database configuration

*Spring, Hibernate and MySQL Server configuration*

To manage access to the database, we only had to configure the driver to connect to the database in the Spring configuration file.
As can be seen in the picture above, in this configuration XML specify the class that defines our datasource, how to access (JDBC, Java DataBase Connectivity) and address and schema in which tables entity – relationship model are hosted.

Spring is responsible for managing each of the layers of the application, which in our case, we will delegate the use of data access to Hibernate.

The presentation layer and the navigation is delegated to Struts.
CHAPTER 4. Implementation

After defining the architecture of both hardware and software and to detail the technologies that have been chosen to carry out the development of our project, we will explain how we have implemented EPSCCommunity.

The options, as well as the interface that each user sees, are different depending on the user profile that is login at the application, differentiate between them according to the degree of privilege. The possible users are:

• System Administrator
• Group Administrator
• Student User
• Guest User

The following describes the interface that each of the users see and how can develop the actions for which they have privileges.

4.1. User Interface

For starters, the first page that appears by default is to login, from which we can access the application, depending on the introduce data are correct and are registered in the database or not.

If authentication is successful, move on to the next level of management and vision of the data, in other case, will see a page with a message wrong username or password. It should be noted that depending on the type of user accessing the application, run a screens flow with some options or other. In the following sections will analyze each possible pages flow and options, depending on the type of user.

In the next picture can see a possible pages flow global scheme, with common options between all users, and as exclusive depend of the user privileges.
4.1.1. System Administrator

4.1.1.1. Home Page

The user, on the home page will appear four options, two of which are common to other users:

- “My Profile”
- “My Mail”
And two others which are unique to this profile:

- “View All Groups”
- “Create Groups”

Then explain each of the above options.

**Fig. 4.2 System Administrator User options**

### 4.1.1.2. View All Groups

The system administrator user can see all groups in the network. From here can access the contents of each of them, and upload documents, create events or participate in discussions in the forum, and delete those entries that consider not appropriate for Epsccommunity, or other user request delete it.

**Fig. 4.3 View All Groups**

### Events

In the Events tab in each group will be able to view, create or delete events, as seen in the picture below.

In this tab you can see the titles of the created events, the user who created them, the start and end date, and description of the event. The option to delete events can be made only the system administrator and group administrators.
Create New Event

By clicking on the "Create Event", see the screen show bellow, where should fill in all fields:

- Title: title of event
- Event description: short description of the actions of the event.
- Start date: The date when the event starts
- End date: date that ends the event.
- Security: Privacy level event
  - Public: visible to all users who belong to this group (students and guests)
  - Private: visible only to privileged student users. Guest users can not see these events.
If any of the fields is left unfilled receive the following error message, the same for all fields:

![Create new Event Validation](image)

**Fig. 4.6** Create Event Validation

**Documents**

On the Documents tab, within each group will have the ability to view, upload or delete documents, as you can see in the picture below.

In this tab, it can see the titles of the documents uploaded, the size they occupy and the user who uploaded.

The option of deleting documents can be made only the system administrators and group administrators.

![View/Upload/Delete Documents](image)

**Fig. 4.7** View/Upload/Delete Documents

**Upload New Document**

By clicking on the "Upload new document", see the screen show bellow, where should fill in all fields:

- **Title**: document title
- **Document File**: path where the document
- **Security**: Privacy level document
  - Public: visible to all users who belong to this group (students and guests)
  - Private: visible only to privileged student users. Guest users cannot see these documents.

![Create new document](image)

**Fig. 4.8 Upload new document**

If any of the fields is left unfilled receive the following error message, the same for all fields:

![Create new document validation](image)

**Fig. 4.9 Upload new document validation**
Forum

In the Forum tab, in each group will be able to view, upload or delete topics, as seen in the picture below.

In this tab, it can see the title of the forum uploaded, the user who uploaded and the responses number.

The option of deleting documents can be made only the system administrators and group administrators.

Fig. 4.10 View/Create/Delete Topics

Create New Topic

By clicking on the "Create new topic", see the screen show bellow, where should fill in all fields:

- What’s about: topic theme
- Security: Privacy level topic
  - Public: visible to all users who belong to this group (students and guests)
  - Private: visible only to privileged student users. Guest users can not see these topics.
The field "What's about" to be filled, or else receive the following error message:

**Fig. 4.12** Create new topic validation

**What's new?**

In this tab it can see what the latest news, recent transactions in the group, new events, new documents uploaded, etc., since the last time that the user had entered at page of this group. In this case it is the system administrator user, but this option is active in all user profiles.

**Fig. 4.13** What's new tab
4.1.1.3. Create Group

The system administrator has the privileges to create all the groups within the social network, and assign it to be the group administrator.

On this page you must fill in all fields of group properties is being created, as you must assign a user to be responsible for managing the group later. If the user has to assign as group manager does not exist, the user must first create a user account.

- **Group name**: name that shall identify the group.
- **Group description**: brief description on the theme of the group.
- **User Group admin**: user that will be the Group Administrator.

As you can see in the picture above, when performing a search to assign the Group Administrator, appears all users that match the initial set for the search. To assign the user who wants to be the Group Administrator must click on the checkbox that appears.

If any of the fields is left without completing a screen like the following warning:
Once you create a new group, the System Administrator can see it on the "View All Groups".

4.1.2. Group Administrator

4.1.2.1. Home Page

To Group Administrator user, on the home page will appear three options, two of which are common to other users:

- "My Profile"
- "My Mail"

And other which are unique to this profile:

- "Edit Group Configuration", which will be detailed below.

In addition, the Group Administrator in its home page has news and documents of the group that administrate, it is noteworthy that currently each Group Administrator can only be administrator of a one group, and may from this screen, view, create and remove events, as documents, such as topics within
the forum, like the System Administrator.

The following is an example. The detailed features have been discussed in the section "System Administrator".

Fig. 4.17 Group Administrator Home Page

4.1.2.2. Edit Group Configuration

The Group Administrator user on this page has the potential to change the properties of the group, Group name and Group description, and have the privileges to delete or add users to group those who have made a request to join this group. The group administrator has the ability to add users with Student profile user (user can see the public and private content of the group to which it belongs) or Guest user profile (user can only view the uploaded public content, the group to which it belongs).

In the next picture you can see an example, where the Group Administrator is the administrator of the group "ISE", in which one has the possibility of delete the group member "pau", and adding a new suggestion to belong group, the user "clara".

Fig. 4.18 Edit Group Configuration
4.1.3. Student and Guest Users

4.1.3.1. Home Page

To Student user and the Guest user, on the home page will appear four options, two of which are common to other users, as mentioned earlier:

- “My Profile”
- “My Mail”

And the following are common to Guest users and the Student Users:

- “Find group”
- “My Groups”

Below is detailed.

![Find group: XSS]

\[\text{Fig. 4.19 Student and Guest user options}\]

4.1.3.2. Find Group

The Student and Guest users on this page are able find the group you want to belong and make a request that the group manager will add to that group.

They have two ways to search this group:

1. By name, either the full name or a list of groups that contain the characters introduce to search.
2. Clicking on the magnifying glass or "enter" key, with the blank form, and display all groups in the network at the time of the search.

The next step is to click on the group that wants to belong, and then the Group Administrator will appear the new request to join.

4.1.3.3. **My Groups**

The Student and Guest users on this page are able to see all the groups to which they belong, and to view and create events, documents and topics in the forum, each of them, and view the latest updates from each group, as has been detailed user profiles discussed above.

**My groups**

- **ISE**
- **grupo1**
Comparing them with other users, the only action that can’t make the Students and Guest Users is to eliminate elements of groups to which they belong.

The difference in stress between the Student User and Guest User is that the Student user can see all kinds of elements of the group, whether public or private, and Guest users can view only those items that users have uploaded as public.

4.1.4. Common Actions

As discussed in the preceding paragraphs, there are common options to all user profiles, these options are: "Authenticate", "My Profile" and "My Mail".

There is also the option "Create Account" which is not different by the user profile because the profile is later awarded by the System Administrator or Group Administrator, so any user that connects to the network can create a account will begin by detailing this option.

4.1.4.1. Create Account

On the first page you see when connecting to the network, the Authentication page, appears a link with the option to create a user account. By clicking on the link, appears the following screen, where you must fill in all requested data.

![Create Account option](Fig. 4.23)
In the event that some of the data is not complete, it would appear the following warning window with the corresponding data validation.

![Form validation dialog](image)

**Fig. 4.24** Create Account Validations

### 4.1.4.2. Autenticate

The option to authenticate on the first page displayed when connecting to the network EPSCCommunity, where you must enter the username and password for an existing account on the network, if you want to access the other pages.

![Autenticate option](image)

**Fig. 4.25** Autenticate option
In the event that the credentials entered are incorrect, or the user account does not exist, an error page appears, indicate username or password wrong.

![Login Failed](image)

**Fig. 4.26 Login Failed**

### 4.1.4.3. My Profile

On this page users can view the data with which they created their user account, and can modify any of the fields that appear on the screen.

Each user can only access their own profile, never that of any other user, even the System Administrator.

![My profile](image)

**Fig. 4.27 My profile**
4.1.4.4. My Mail

On this page all users can view your mailbox and create new emails to send to any user.

Mails can be sent and received by any member of the social network to any member of the network without the need to belong both to the same groups.

Then you see the look of the page from the mailbox.

![My mail Option](image1)

**Fig. 4.28** My mail Option

The first post that every user receives is the Welcome to the social network as we see in the image above in the mail that is deployed.

In the In Box you can see the mail received, and if they have any unread or not. Clicking on the link is accessed In Box, but this is the one that appears by default when accessing the page "My Mail".

In the Out Box appear the sent mails.

If you want to create a new email, you must click on the button "Create New Mail", as you can see in the picture above, and you will see a pop up, where you must fill in all data.

![Create New Mail Option](image2)

**Fig. 4.29** Create New Mail Option
In the "Search users" options has been used jquery autocomplete function, and introducing the initial or part of the name, the application looks to all network users coincidences.

![Search users](image)

**Fig. 4.30** Autocomplete

If any of the fields are not filled, a warning message appears, indicating the area that has been left blank.

![Create new email](image)

**Fig. 4.31** Create Mail Validations

### 4.2. Not Implemented functions

At the beginning of this project, we have specify some a number of functional requirements that we thought interesting to implement, but due to time constraints, some of them we could not perform. They are:

#### 4.2.1. Asynchronous Chat

One of the interesting features that we are interested in is the asynchronous chat. It is interesting to facilitate communication between users in system in real time, different than forum or mail system.
To achieve an Asynchronous Chat need to be able to send replies to the browser on their own. To do this, the most direct way is by using a consultation mechanism (HTTP Polling). It consists on sending requests at regular intervals of time, so that the system has continued opportunity to update the presentation.

![HTTP Polling](image)

**Fig. 4.32.** HTTP Polling solution

To implement this using J2EE we need one of the latest updates servlet 3.0. In this package appears the solution implemented, Asynchronous Request Processing (ARP).

Other ways would be:

- Tomcat 6 Comet Processor
- Glassfish Grizzly Connector
- Jetty Continuations
- WebLogic Future Response Servlet
- WebSphere Asynchronous Request Dispatcher

### 4.2.2. Delete group

The option to delete a group would have to be exclusive to the administrator. Implementing this functionality should not be very different from the option to delete forum topics.

This requires defining the relationship between group and forum, documents and events as CascadeType.ALL. Thus, removing a group will remove all the containing entities him on cascade mode.

Otherwise, the relation between group and users must not be like that because removing a group will remove all the users that are joining this group.
4.2.3. **Delete content by Student and Guest user**

To develop this option we only have enabled the function performed by the system administrator and the group administrator for students and guests and monitor that they are the owners of content that want to remove.

It should not be difficult because we store user permissions and who is the owner of the content.

4.2.4. **Modify content**

To implement this function we would have to show the contents in edition mode.

The logic business should be transparent and the data layer is prepared to implement this option.
CHAPTER 5. Testing

5.1. Unitary Test

Unitary Test it's a good methodology to develop unit test in any software project. As we have explained in the chapter about Junit, during this project we have developed tests for all methods of business logic and data layer.

This gives us the possibility to run the test when we want or even when we make a version of the application.

Let's see and example:

Fig. 5.1. Execution of one unitary test

You can see how we ran the data layer documentation test. All method tests have worked well.

The other option is to launch the test during the making version process, like this:

Fig. 5.2. Execution of unitary test during generation version process
In case of a test fail the generation version process could be configured to stop.

5.2. Stress Test

5.2.1. Introduction. What is a Stress Test?

Every Web application has to undergo a series of tests to determine or estimate their behaviour during a high load period. These tests are called stress tests.

Stress tests are usually performed on the production environment (or a clone) to be able to monitor different parameters of the application like response time, CPU usage, memory usage, network statistics ... so we can determine which is the parameter that will be the bottleneck of our application.

Once done, if we understand that the load is not sufficiently well dimensioned and the results were not the expected, we can decide to extend server memory, CPU, improve some programming aspects that are not optimal, improving performance.

5.2.2. EPSCCommunity Stress Test

In our case, these estimations are complicated because it has not done a study dimensioning the number of users who will use the application. Even so, we may take as the value of the number of registered students by adding the number of university teachers.

However, we failed to achieve the goal of deploying the application in a production environment, so stress the application to obtain values such as CPU usage, memory usage does not make much sense.

In the case where we are, we decided to take half of the time on the application takes to serve the different URLs, and can estimate the average time it takes to make a click on the application. Let's see the results.

Table 5.1. Summary of times per transaction

<table>
<thead>
<tr>
<th>Operation</th>
<th>Time (ms)</th>
</tr>
</thead>
<tbody>
<tr>
<td>Create an account</td>
<td>531</td>
</tr>
<tr>
<td>Login a user</td>
<td>47</td>
</tr>
<tr>
<td>Logout session</td>
<td>15</td>
</tr>
<tr>
<td>Search a user in application</td>
<td>62</td>
</tr>
<tr>
<td>Create a group</td>
<td>47</td>
</tr>
</tbody>
</table>
As you can see from the results obtained, the average time per operation is 280.54 ms. This will be more than acceptable for a Web environment, where we could estimate between one and three seconds the maximum time that a user could expect to reach result.

We emphasize that this test is performed on a no production environment in which the database and the application server are located in the same computer. We also note that the test was performed with a single user in the system, although we understand that the robust architecture of the application will not this time multiplied by the number of users due to the use of Spring and Hibernate.

5.3. Usability testing

5.3.1. Introduction

Once the design and application development, and have detected and corrected the majority of technical failures, the objective of evaluation is to check its usability with a small Group of “real” users.

It is important to check the ease of intuitive use of the system, so that a new user be able to move fluidly through the social network.
Evaluate the usability of the application is very useful because errors are discovered at the Web application design, and this is the first step to correct them in the future.

There are several ways to evaluate a web application, we have chosen the usability test, where multiple users will perform a “assisted” navigation by the application and answer a few questions about the application. Because it is an assisted testing we will also have the opportunity to discover failures of our social network.

5.3.2. Usability Test

In usability testing performed, to motivate the user are given some guidelines to follow, with which you can answer all questions that are performed in the test. The guidelines to follow are:

1. Create a user account
2. Sign up for a group
3. Access to group
4. Check mail
5. Send a mail to administrator
6. Upload a document
7. Create one event
8. Create one topic of the forum
9. View latest updates

The test has been carried out at 10 users, whose responses are summarized in the following tables:

Table.5.2. Content questions

<table>
<thead>
<tr>
<th>Questions</th>
<th>Yes</th>
<th>No</th>
<th>Result</th>
</tr>
</thead>
<tbody>
<tr>
<td>1.- Is it easy to distinguish the new content that presents the application?</td>
<td>10</td>
<td>0</td>
<td>1</td>
</tr>
<tr>
<td>2.- Is it possible to know when was the last update of the site?</td>
<td>0</td>
<td>10</td>
<td>0</td>
</tr>
<tr>
<td>3.- Are the text used in the contents of the links sufficiently descriptive of what is offered in the pages to which are accessed through them?</td>
<td>8</td>
<td>2</td>
<td>0,8</td>
</tr>
<tr>
<td>4.- If the contents offered files attachments, it was easy to know its size and format?</td>
<td>9</td>
<td>1</td>
<td>0,9</td>
</tr>
<tr>
<td>5.- If there is information relating to what it saw, have it presents so simple?</td>
<td>8</td>
<td>2</td>
<td>0,8</td>
</tr>
</tbody>
</table>

Result = 3,5/5
Table 5.3. Navigation questions

<table>
<thead>
<tr>
<th>Questions</th>
<th>Yes</th>
<th>No</th>
<th>Result</th>
</tr>
</thead>
<tbody>
<tr>
<td>1.- Can you see on the principal and other pages, how to navigate for the site?</td>
<td>10</td>
<td>0</td>
<td>1</td>
</tr>
<tr>
<td>2.- Are there elements within the pages, which lets you know exactly where it is in this site and how to go back without using the buttons on the browser?</td>
<td>9</td>
<td>1</td>
<td>0,9</td>
</tr>
<tr>
<td>3.- Do you see any way to back from any page of the site to homepage?</td>
<td>8</td>
<td>2</td>
<td>0,8</td>
</tr>
<tr>
<td>4.- Usually, how it manages directly access to the contents without having to navigate? Are there shortcut icons?</td>
<td>10</td>
<td>0</td>
<td>1</td>
</tr>
<tr>
<td>5.- Were you able to graphically distinguish visited links from those who have not visited yet?</td>
<td>4</td>
<td>6</td>
<td>0,4</td>
</tr>
<tr>
<td>6.- Have you ever felt lost within the site?</td>
<td>8</td>
<td>2</td>
<td>0,8</td>
</tr>
</tbody>
</table>

Result = 4,9/6

Table 5.4. Presentation questions

<table>
<thead>
<tr>
<th>Questions</th>
<th>Yes</th>
<th>No</th>
<th>Result</th>
</tr>
</thead>
<tbody>
<tr>
<td>1.- Are correct the images to represent the content of the social network?</td>
<td>9</td>
<td>1</td>
<td>0,9</td>
</tr>
<tr>
<td>2.- Are clear the images?</td>
<td>4</td>
<td>6</td>
<td>0,4</td>
</tr>
<tr>
<td>3.- Are there Graphics animations?</td>
<td>0</td>
<td>10</td>
<td>0</td>
</tr>
<tr>
<td>4.- Do you consider that the submission of the application is balanced? (no simple and no ornate)</td>
<td>7</td>
<td>3</td>
<td>0,7</td>
</tr>
</tbody>
</table>

Result = 2/4

Table 5.5. Utility questions

<table>
<thead>
<tr>
<th>Questions</th>
<th>Yes</th>
<th>No</th>
<th>Result</th>
</tr>
</thead>
<tbody>
<tr>
<td>1.- After a first look, it is clear the purpose of the application?</td>
<td>10</td>
<td>0</td>
<td>1</td>
</tr>
<tr>
<td>2.- Can you easily distinguish the contents and services offered?</td>
<td>8</td>
<td>2</td>
<td>0,8</td>
</tr>
<tr>
<td>3.- Do you think that services and content provided are useful for students?</td>
<td>10</td>
<td>0</td>
<td>1</td>
</tr>
</tbody>
</table>

Result = 2,8/3
Table 5.6. Search questions

<table>
<thead>
<tr>
<th>Questions</th>
<th>Yes</th>
<th>No</th>
<th>Result</th>
</tr>
</thead>
<tbody>
<tr>
<td>1.- Distinguished if the application have search option?</td>
<td>4</td>
<td>6</td>
<td>0,4</td>
</tr>
<tr>
<td>2.- Known how to perform a search?</td>
<td>4</td>
<td>6</td>
<td>0,4</td>
</tr>
<tr>
<td>3.- After perform the search that the application offers, the results are expected to find?</td>
<td>9</td>
<td>1</td>
<td>0,9</td>
</tr>
</tbody>
</table>

Result = 1,7/3

Table 5.7. Tests Results

<table>
<thead>
<tr>
<th>Test</th>
<th>Result</th>
<th>Evaluation</th>
</tr>
</thead>
<tbody>
<tr>
<td>Content questions</td>
<td>3,5/5</td>
<td>7,5</td>
</tr>
<tr>
<td>Navigation questions</td>
<td>4,9/6</td>
<td>8,1</td>
</tr>
<tr>
<td>Presentation questions</td>
<td>2/4</td>
<td>5</td>
</tr>
<tr>
<td>Utility questions</td>
<td>2,8/3</td>
<td>9,3</td>
</tr>
<tr>
<td>Search questions</td>
<td>2,1/3</td>
<td>5,6</td>
</tr>
</tbody>
</table>

Total Evaluation 7,1

5.3.3. Improvement Points

After perform usability tests and study results, the most important points for improvement for students user options are those listed below:

1. Set date for all actions that it can perform; date of documents uploaded, events created, forum topics edited, and send and received mail.

2. Improve the presentation, images and the structure of the application. To some users seemed simple.

3. Add multimedia content to make the application more attractive.

4. Make that it possible to distinguish the visited and not visited sites, through the colour distinction.

5. Add a search option for any social network content, not only the groups.
CHAPTER 6. Conclusions

6.1. Environmental impact

Being a only software project, we can say that there a direct environmental implications, apart from electricity consumption produced by machines that use this software.

6.2. Objectives achieved

Once finished the implementation phase of our education social network is time to take a balance and one step back to see if the original objectives have been met or not.

Assuming that the experience in the development of applications like this was almost nil, and we started following an idea from scratch, the balance has been very positive.

It has managed to have an educational social network complete and ready for use. The results of the tests performed to test usability, have been favorable, thanks to the simplicity of application.

It must be said that some of the objectives set at the beginning of this project have not been met, then list the highlights:

- Instant Messaging
  - Has not been carried out the chat implementation, not only for lack of time, but not be strictly necessary for the effective use of social network, as it consists of Private Messaging (Mail) and Forum to share ideas and debates.
  - It was making the same development as in the Mail case, but in real time, synchronized.

- Delete group
  - Not implemented this option, the fact of being repetitive, and not providing any new value to the application, since there are a delete users and delete content. It would be useful to the System Administrator user in the event that any of the groups ceased to be used, or were repeated, etc,
- Delete Content by Student and Guest user
  
  - As in the previous case is not implemented this option did not bring extra value to the application, because it would be used to repeat code for delete users and delete content, changing the variables, which can be easily done in the future. This option would be implemented for the student user, who would have the opportunity to delete uploaded erroneously.

- Modify content

  - This function is not implemented in application due to lack of time, but it is one of the present points for future application enhancements, as we believe it is important that a user can change the contents once uploaded.

Overall, we can say that it has met the initial expectations and has achieved an application with a structured and robust base, and easily scalable in the future.

### 6.3. Future enhancements

After seeing the final result and compare it with the original objectives, there are parts that we believe would be necessary to implement, such as those which have been discussed in the previous section, Instant Messaging (chat) Groups Delete, Delete by Student and Guest content User and Modify Content. Others comment below:

- The group administrator can only be administrator of a one group, it would be useful to have the option to be administrator of more than one group, as a teacher, can be a teacher of more than one subject, therefore should have the option to manage each group of each subject.

- In the mail, does not appear the date of send or receive, although it is known what is the most current, not knows the day of receive or send, the same for the creation of documents and forum topics.

- Being able to change the contents once uploaded to the network.

- Multilanguage. The application is ready to support it, but the functionality is not implemented.

- Existing validation username and password.

- Receive email notifications.
- Having a wall by user, not by group.
- Uploading pictures and multimedia contents
- Link the system login to some centralized system of the university (LDAP by example)
- Edit documents concurrently

### 6.4. Technologies conclusions

To work in this kind of software development projects gives us the opportunity to experiment and learn how to work in all phases of a project

- Requirement analysis
- Design
- Implementation
- Testing
- Deployment and implantation

In our case we have gone through different stages of the life cycle of the application except the last, the deployment phase.

In each of them, we were able to experiment with technologies, tools and frameworks that have provided pointers as far as possible the achievement of different objectives in each phase.

The fact of having to seek alternative to technologies, discard them, choose one that adapt better to our needs and finally to learn how to use it and configure it has forced us to have to investigate and consult different learning spaces as forums, official web pages, documentation, mailing lists... giving as the learning to accomplish our objectives.

The use of these technologies has given us the knowledge necessary to raise, implement and implant a certain magnitudes project from scratch.

### 6.5. Personal conclusions

After reaching the end point of this project are many conclusions have been reached

First, the fact to have to introduce ourserlves in the world of the social networks, a world with a incredible growth in the lasts years, has been of great interest to
us because of the ignorance of the existence of many social networks, targeted at any topic and type of people.

Otherwise, the fact to have to develop this applications from the requirement analysis point and introduce ourselves in the software engineering, one aspect of the career we did not work so much but we have learned during this months doing the application development or knowing this new technologies.

The fact that our social network can be used in the University that has given us the foundation of knowledge to implement it is very rewarding, and that in the near future may be in use by members of the EPSC, we feel something really personally satisfying.

Definitely, the global valuation of the project is very positive because much of the initial expectatives have been accomplished.

The application has been made from beginning to end, independently and with a very positive result in the tasks performed and knowledge gained.
CHAPTER 7. Bibliography

[1] Spring Source Community
Web Page, URL:  http://www.springsource.org/  June, 2010

Web Page, URL:  http://tomcat.apache.org/  June, 2010

[3] Hibernate
Web Page, URL:  http://www.hibernate.org/  June, 2010

Web Page, URL:  http://struts.apache.org/  June, 2010


[6] Explore the Eclipse Universe
Web Page, URL:  http://www.eclipse.org/  June, 2010

[7] Collabnet
Web Page, URL:  http://www.collab.net/  June, 2010

[8] The Apache Maven Project
Web Page, URL:  http://maven.apache.org/  June, 2010

[9] JUnit.org Resources for Test Driven Development
Web Page, URL:  http://www.junit.org/  June, 2010

[10] The Apache Jakarta Project
Web Page, URL:  http://jakarta.apache.org/jmeter/  June, 2010

Web Page, URL:  http://jqueryui.com/  June, 2010

[12] Internet en el Aula
Web Page, URL: http://www.dosideas.com/noticias/java/547-la-revolucion-de-la-web-asincronica.html June, 2010

Web Page, URL: http://www.wikipedia.org/ May, 2010


[16] Unified Modelling Language
Web Page, URL: http://www.uml.org/ April, 2010

[17] Ning. Create and discover Ning Social Networks
Web Page, URL: http://www.ning.com/ February, 2010