Internet Technologies

Lab1 2011
Introduction
Overview

- What will we do in the labs?

- Project
  - Requirements
  - Examples
  - Evaluation

- Tools
How to reach us?

- Cavada Dario:
  - cavada@ectrlsolutions.com

- Mehdi Elahi:
  - mehdi.elahi@stud-inf.unibz.it

- The lab material is on
  - http://www.inf.unibz.it/~ricci/IT/index.html
We have 12 labs in which we will:

- Run the software examples
- Solve small exercises
- and also work on the project
Structure of the Project

- The **application** must run on the application server that we shall indicate in the labs.
- The **report** must describe clearly in **min 2000, max 3000** words, plus images.
  - The **functions** of the web application and their **motivation**
  - The **architecture** of the application (modules and their roles) – use figures
  - Main **classes** and main **methods**
  - Major **technical problems** found during the work
- The project will be **evaluated** according to: complexity of the implemented functions, user interface usability and completeness, organization of the code, coverage of the required technologies.
Functions

- **User Management**
  - List existing users of the system
  - Creation of a new user
  - Deletion of the existing user
  - List and modify access rights of the users - check boxes with some capabilities (min 3) – e.g.
    - A user can comment all items
    - A user can download an item with label “parties”
  - User registration and login to the system
Functions (II)

- **Items management**
  - Users add, edit or remove items
  - Users comments or reviews items
  - Administrator can manage the comments (edit, remove, add)

- **Personalization**
  - Salutation for a returning user
  - List resources that are new from the last visit
  - Customization of the layout for a class of users.
Techniques – MUST be used

- Static HTML: "natürlich"
- CSS: all the look and feel must be in CSS files
- Javascript: check input and manage menus
- Servlet
  - Reading (parameters and headers) and writing headers and resulting page
  - Session management with cookies and session object
  - Redirect the client
  - Forward to another page or servlet
- JSP
  - Expressions, scriptlets and declarations
- Beans
- DBMS access through JDBC
- Integration of JSP and Servlets (forward and include) using MVC pattern.
Software Components

- DBMS – PostgreSQL (lamj.inf.unibz.it)
- Application server – Tomcat 6.x (lamj.inf.unibz.it)
- Programming language – Java 1.6
- IDE – Eclipse or Netbeans

- Minimal Framework – we will provide you with a minimal framework:
  - Layout
  - Example CSS
  - Connection to DBMS
  - JSP and Servlet examples
Project Evaluation

- **Usability**
  - Minimum level of usability! We should be able to visit all the system functions and use them without any "manual" – the navigation through the functions should be supported

- **No errors**
  - The system should run smoothly without errors and with reasonable response times

- **All the required functions should be supported**

- **All the required techniques should be used**

- **The report**
  - must clearly illustrate the design choices: functions and the technical implementation
  - **min 2000, max 3000** words, plus images.
Timeline

- Written exam is on ???
- The projects should be uploaded not later than ??? on the web site ... (instructions will follow)
- The report should be upload together with the project as a PDF file and in the home page there must be the link to the pdf
Evolution of Project 1/10

- First you will create a static HTML web page
- You will validate your HTML using W3C validators
Evolution of Project 2/10

- Next, you will extend it with CSS

LINAS::MISC

LINKS
- kalnai.blogspot.com
  My blog about experiences in mountains. Sorry only in Lithuanian language.
- photos@megalogika.lt
  Some traveling photos.
- PhD@inf.unibz.it
  Webpage of the PhD students at the Free University of Bolzano.
Evolution of Project 3/10

- You will create HTML forms for submitting data to the server
- You will add basic Java Script functionality to your web page
  - Form fields checking
  - Fast image preloading
  - etc
Evolution of Project 4/10

- Configure Tomcat application server
- Implement initial server side application
  - Posted data managing (from HML Forms)
  - Uploading and displaying photos/music
Evolution of Project 5/10

- Create DDT and XML of trusted users database
- Check Validity
- Advanced – RSS feed of news on your page
Evolution of Project 6/10

- Add redirect function to the application
- Respond differently depending on the web-browser
- Maybe somebody wants to make WAP response...
Evolution of Project 7/10

- Add cookies to track the persons that access the page
- Maybe implement functionality to allow some context only for trusted users (from XML)
- Respond to old users with a message
Evolution of Project 8/10

- Replace starting page with a JSP page
Evolution of Project 9/10

- Add database support
  - Retrieve data using JDBC
  - We will use PostgreSQL for our projects
  - Display the data
Evolution of Project 10/10

- Implement a shopping/visiting cart
  - User can select what pictures to download in a single zip from different albums
  - Other information managing within the session
Project Evaluation

- The project will be evaluated according to:
  - complexity of the implemented functions
  - user interface usability and completeness
- Learn from bad examples:
  - http://www.webpagethatsucks.com
  - http://www.havenworks.com/
- Good practical guide:
  - http://www.usability.gov
  - http://www.w3schools.com/quality/default.asp
- organization of the code
Tools

- Java IDE for server side coding, XML editing, validating
  - Eclipse
  - NetBeans

- Web Browsers
  - Firefox firebug + codeburner
  - Firefox developer plugin

- rdesktop (only for Linux users (we suggest using Windows or Mac 😊))
  - rdesktop turing.inf.unibz.it -d UNIBZ -u _USERNAME_ -a 16 -g 1280x960
Java Reminder

- Create your first webapp
  - JSP
  - Servlet
- Install Firefox plugins
Install Tomcat on your IDE

- Requirements:
  - J2 SDK 1.6
  - Tomcat 6.0.14
  - Eclipse or NetBeans
Eclipse: Show Server View
Eclipse: Define New Server
Eclipse: Define New Server (cont.)

Define a New Server

Choose the type of server to create

Server's host name: localhost

Select the server type:

- Tomcat v6.0 Server

Publishes and runs J2EE and Java EE Web projects and server configurations to a local Tomcat server.

Server name: Tomcat v6.0 Server at localhost

Server runtime environment: Apache Tomcat v6.0

New Server Runtime Environment

Specify the installation directory

Name:
Apache Tomcat v6.0

Tomcat installation directory:
C:sers\apache-tomcat-6.0.14

JRE:
Workbench default JRE

Download and Install...
Eclipse: new project
Eclipse: new project (cont.)
NetBeans: Define New Server

Server Location: C:\{cots\apache-tomcat-6.0.14

Use Private Configuration Folder (Catalina Base)

Catalina Base: 

Enter the credentials of an existing user in the "manager" role

Username: 
Password: 

Create user if it does not exist
NetBeans: new project