
Introduction

Denis Helic

- ▶ Multimediale Informationssysteme 2 (VU 706.154)
 - ▶ Elective (optional) course for Telematics
 - ▶ Elective (optional) course for Computer Science
 - ▶ Subject area catalogue: Information Systems, Multimedia

▶ Who is talking?

Name: Denis Helic

▶ How can I reach him?

Office: IICM, Inffeldgasse 16c, 1st Floor, Room
D.2.05

Office hours: Wednesday from 11 til 12

Phone: +43-316/873-5617

email: dhelic@iicm.edu

- ▶ Language
 - ▶ Lectures in English
 - ▶ Communication in German/English
 - ▶ If in German: please informally (Du)!
 - ▶ Constructional Example: German/English
 - ▶ Examination: German/English

Organization of the course

- ▶ Lectures: Thursday, 11:15 – 12:45, HS i12
- ▶ Registraton for the course TU Online System until 26.03.2008
- ▶ Lecture Slides, Links to external resources
(<http://coronet.iicm.edu/lectures/mmis2>).
- ▶ The Web site will be updated a couple of times during the term
- ▶ Newsgroup `tu-graz.lv.mmis2` (`news.tu-graz.ac.at`)

- ▶ In MMIS2 we deal with the Web as an application platform
- ▶ **Goal:** To learn and understand the **specifics of Web as an application platform**
 - ▶ Client/Server, Data Formats, ...
- ▶ Further, we deal with the Web as a huge database offering data and services
- ▶ **Goal:** To learn and understand the **specifics of Web as an information system**
 - ▶ HTTP, URL, resources

- ▶ **Goal:** To learn about the **technological trends**
 - ▶ What technology is currently used? (XML, (X)HTML, Web App Frameworks, Web Services, AJAX, ...)
- ▶ **Goal:** To learn about the **architectural trends**
 - ▶ REST, addressability, statelessness, ...

How we will achieve these goals?(1/2)

- ▶ The theoretical background and overview of the current trends
 - ▶ Lectures, slides, links to articles, ...
- ▶ Practical implementation of a Web application or a Web-based information system
 - ▶ Following a software (Web) engineering method
 - ▶ Using some of the technologies that we have discussed
 - ▶ Applying architectural and design patterns

How we will achieve these goals?(2/2)

- ▶ Presentation of the achieved results (this term not needed)
- ▶ I'm in the USA starting May 2008
- ▶ Submission by e-mail, discussion of the results by e-mail

- ▶ Traditional Page-oriented Web applications
 - ▶ Server-side programming with Java (Ruby)
- ▶ Design Patterns
 - ▶ Model-View-Controller Pattern
 - ▶ Web Application frameworks: Apache Struts, Ruby on Rails, Stripes,...

- ▶ Data Management in Web applications
 - ▶ Data in RDBMS, App logic in an OO programming language
- ▶ Object/relational mappings
 - ▶ Frameworks: Hibernate, ActiveRecord, ...
 - ▶ Patterns: Data Access Object, ...

- ▶ Service oriented Web applications
- ▶ Web Services
 - ▶ SOAP, WSDL
 - ▶ Web Services Frameworks: Apache Axis

- ▶ Web as a (hypermedia) database
 - ▶ REST architectural style
- ▶ Advanatages, scalability, addressability
- ▶ Programmable Web, Mashups





- ▶ Novel user interface techniques
- ▶ Rich Web Clients
 - ▶ AJAX, AJAX libraries, ...
 - ▶ XMLHttpRequest Object
- ▶ Web 2.0
- ▶ Combining REST and AJAX

- ▶ 06.03. Web Eng. Intro/Intro on frameworks for practical work
- ▶ 13.03. Data Management in Web Applications (O/R Mapping, Hibernate)
- ▶ 10.04. Traditional Web applications (Model-View-Controller) (Struts, RoR, Stripes, ...)
- ▶ 17.04. Big Web Services (SOAP, WSDL) and problems

- ▶ 24.04. REST Architectural style
- ▶ 08.05. Web 2.0 (AJAX, XMLHttpRequest, ...), combining AJAX and REST
- ▶ June Project submissions via e-mail

Organization of the course: Practical part(1/6)

Goals:

-  Implementing a Web application or a Web database
-  Learning about different technologies, methods
-  Working in groups
-  To try out something new

Organization of the course: Practical part(2/6)

- ▶ Prerequisites
 - ▶ (X)HTML, CSS
 - ▶ XML
 - ▶ HTTP, URI
 - ▶ Java or another OO language
 - ▶ Optionally JavaScript

Organization of the course: Practical part(3/6)

- ▶ Group work: groups of 3-5 people
- ▶ Group registration with a short description
<http://coronet.iicm.edu/denis/students/mmis/group.txt>
- ▶ Project plan (Time estimation, group members responsibilities, ...)
<http://coronet.iicm.edu/denis/students/mmis/plan.txt>

Organization of the course: Practical part(4/6)

- ▶ Documentation of Code and Design!
<http://coronet.iicm.edu/denis/students/mm1s/doc.pdf>
- ▶ KU example points to MMIS 1 (documents are same, the content is different)
- ▶ Documentation is not equal to the source code
- ▶ Documentation is a description of the system
 - ▶ Architecture, Installation, User Guide, etc.

Organization of the course: Practical part(5/6)

- ▶ Presentation (not needed in this term)
 - ▶ Submit the project via e-mail, answer a couple of questions via e-mail
- ▶ Argue about your development decisions
 - ▶ Why did you take MVC architecture?
- ▶ Point advantages, disadvantages

Organization of the course: Practical part(6/6)

- ▶ Deadlines:
 - ▶ Group building and preliminary description of the project (what we are going to do): 26.03.2008
 - ▶ Project plan: 09.05.2008
 - ▶ Completed project: 20.06.2008

- ▶ Grading of the course:
 - ▶ Project plan: 12,5%
 - ▶ Documentation: 12,5%
 - ▶ Implementation: 25%
 - ▶ Presentation 25%
 - ▶ Answering questions 25%
- ▶ Project submission via e-mail: Denis Helic (dhelic@iicm.edu)

- ▶ Important for a good mark!!!
- ▶ Show how you applied design patterns
- ▶ Show how you applied a method
- ▶ Argue why you did something!!!
- ▶ Advanatages/disadvantages!
- ▶ Note that this is very similar to Sem/Proj or Master Exam

- ▶ Students can implement whatever application/database they like!
 - ▶ Web shops
 - ▶ Content managment systems
 - ▶ Collaborative systems
 - ▶ Digital libraries

- ▶ Implementation of each project has to have these three layers
- ▶ 1. Data Management Layer
 - ▶ Implemented either with a framework or with a design pattern
- ▶ 2. Middleware Layer (Server side)
 - ▶ Implemented either with an MVC framework or as REST Web services
- ▶ 3. Presentation Layer (Client side)
 - ▶ Implemented either as XHTML or as AJAX