(1) GENERAL

<table>
<thead>
<tr>
<th>SCHOOL</th>
<th>Engineering</th>
</tr>
</thead>
<tbody>
<tr>
<td>ACADEMIC UNIT</td>
<td>Department of Computer Engineering and Informatics</td>
</tr>
<tr>
<td>LEVEL OF STUDIES</td>
<td>Undergraduate</td>
</tr>
<tr>
<td>COURSE CODE</td>
<td>CEID_NY538</td>
</tr>
<tr>
<td>SEMESTER</td>
<td>FALL</td>
</tr>
<tr>
<td>COURSE TITLE</td>
<td>Web Programming and Systems</td>
</tr>
</tbody>
</table>

INDEPENDENT TEACHING ACTIVITIES

If credits are awarded for separate components of the course, e.g. lectures, laboratory exercises, etc. If the credits are awarded for the whole of the course, give the weekly teaching hours and the total credits.

<table>
<thead>
<tr>
<th>Lectures, Tutorials, Laboratory</th>
<th>WEEKLY TEACHING HOURS</th>
<th>CREDITS</th>
</tr>
</thead>
<tbody>
<tr>
<td>3 (L), 1 (T), 2 (L)</td>
<td></td>
<td>6</td>
</tr>
</tbody>
</table>

Add rows if necessary. The organisation of teaching and the teaching methods used are described in detail at (d).

COURSE TYPE

Skills development

PREREQUISITE COURSES:


LANGUAGE OF INSTRUCTION and EXAMINATIONS:

Greek. The course may be offered in English for Erasmus students.

IS THE COURSE OFFERED TO ERASMUS STUDENTS

Yes

COURSE WEBSITE (URL)

https://eclass.upatras.gr/courses/CEID1092/

(2) LEARNING OUTCOMES

Learning outcomes

The course learning outcomes, specific knowledge, skills and competences of an appropriate level, which the students will acquire with the successful completion of the course are described.

Consult Appendix A

- Description of the level of learning outcomes for each qualifications cycle, according to the Qualifications Framework of the European Higher Education Area
- Descriptors for Levels 6, 7 & 8 of the European Qualifications Framework for Lifelong Learning and Appendix B
- Guidelines for writing Learning Outcomes

Upon conclusion of the course, the students ought to be able to:

1. Know the basic operational principles of the WWW (Client-server architectures, Peer-to-Peer, use of webpages and websites).
2. Understand the basic structural protocols of the WWW (HTTP(s), DNS, TCP/IP).
3. Configure the structure and operation of web proxies in the WWW (shared caches, effects on the quality of service, data pre-fetching, cache consistency and replacement policies).
4. Understand the structure and operation of web replication systems (mirroring, architectures, metrics and algorithms for request distribution, content delivery networks).
5. Use DNS and HTTP support to operate web caching and replication systems.
6. Understand the operation principles of web browsers.
7. Construct the basic elements of a webpage using HTML.
8. Format HTML elements using CSS.
10. Understand the differences between client and server-side scripting technologies.
11. Configure and operate a web server.
12. Use web sessions and web cookies.
13. Develop integrated web applications using server-side technologies.
14. Employ basic principles and techniques for secure web applications.
15. Interconnect web applications with database management systems.
16. Use asynchronous client-server communication in web applications (e.g. AJAX).
17. Develop web applications using standards for describing, exchanging and structuring data (XML/XSL).
18. Use modern technologies for the development of mobile web applications, online geospatial services and online social networks.

Upon conclusion of the course, the students are expected to have the following skills:

1. Be able to exploit:
   - The methods offered by the HTTP protocol.
   - The effects of web caching in proxies and mirroring in content delivery networks to improve the quality of service over the WWW.

2. Be able to develop novel integrated web applications which:
   - Provide a full, usable and interactive interface with users, using HTML5, CSS3 and JavaScript.
   - Connect to a database management system to save and retrieve data (e.g. user credentials, application data etc).
   - Use algorithms to solve complex problems (e.g. shortest path finding in logistics or transport systems).
   - Communicate with other web systems whose operation might be unpredictable and which are not controlled by the developers (e.g. Google Maps, Facebook API etc), using the available APIs and data exchange standards (XML, JSON etc).

3. Configure the environment for the installation and operation of their developed web applications, indicatively:
   - Install and configure a web server (e.g. Apache)
   - Install and configure server-side programming languages (e.g. PHP)
   - Install and configure database management systems (e.g. MySQL)

4. Be able to examine the conformation of their developed web applications to basic principles of information systems security and the robustness of the developed system in cases of unpredicted or malicious use (e.g. using sessions to ensure controlled authorised access to parts of the web application).

5. Be able to recognise problems and design and develop complex integrated applications based on the pertinent social, scientific and ethical issues that apply to the development scenario.

6. Manage the collaboration and communication with other team members, develop organisational plans for the development and progress monitoring of team work and distribute workloads amongst team members.

7. Undertake personal initiative, responsibility and decisions that affect the progress and outcomes of team project assignments.

General Competences

Taking into consideration the general competences that the degree-holder must acquire (as these appear in the Diploma Supplement and appear below), at which of the following does the course aim?

- Search for, analysis and synthesis of data and information, with the use of the necessary technology
- Project planning and management
- Adapting to new situations
- Respect for difference and multiculturalism
- Decision-making
- Respect for the natural environment
- Working independently
- Showing social, professional and ethical responsibility and sensitivity to gender issues
- Team work
- Criticism and self-criticism
Working in an international environment
Working in an interdisciplinary environment
Production of new research ideas

- Search for, analysis and synthesis of data and information, with the use of the necessary technology
- Adapting to new situations
- Decision-making
- Team work
- Project planning and management
- Production of free, creative and inductive thinking

(3) SYLLABUS

The course is divided in two main parts, Web Programming and Web Systems.

Web Programming
1. Introduction to the World Wide Web.
2. HTML4, HTML5 and CSS.
3. JavaScript, Bootstrap, introduction to AngularJS.
4. Introduction to server-side technologies, the PHP web programming language.
5. Connecting a web application with a database management system (PHP and MySQL)
6. Store and exchange data on the web, introduction to XML. Formatting of XML, introduction to XSL.
7. Client – Server asynchronous communication. Introduction to AJAX, JSON, jQuery and NodeJS.
8. Introduction to mobile web. Design principles for mobile web applications.
9. Introduction to online social networks and communication with third party web systems (web services and web APIs).

Web Systems
1. Introduction to web caching and web replication.
2. Internet protocols. URL, DNS and HTTP (simple, persistent & pipelining).
3. HTTP support for web caching and replication. Security and cryptography during data transmission over the WWW (SSL/TLS, HTTPS).
4. Performance evaluation of web caching, TCP connection splitting, TCP connection caching.
5. Algorithms and cache consistency management techniques, validation protocols (leases, subscriptions, subscribed leases), validation protocols (chaining, TTL adaptation, piggyback cache validation)
6. Content replication systems (static/transparent, full/partial mirroring and combinations, request distribution techniques with or without content awareness)

(4) TEACHING and LEARNING METHODS - EVALUATION

<table>
<thead>
<tr>
<th>DELIVERY</th>
<th>Ex cathedra.</th>
</tr>
</thead>
</table>

| USE OF INFORMATION AND COMMUNICATIONS TECHNOLOGY | The slides of the course and additional supplementary material are freely available from the course’s website. Communication with students is done through a dedicated e-forum. |

<table>
<thead>
<tr>
<th>TEACHING METHODS</th>
<th>Activity</th>
<th>Semester workload</th>
</tr>
</thead>
<tbody>
<tr>
<td>Lectures</td>
<td>3x13 = 39</td>
<td></td>
</tr>
</tbody>
</table>
described in detail. Lectures, seminars, laboratory practice, fieldwork, study and analysis of bibliography, tutorials, placements, clinical practice, art workshop, interactive teaching, educational visits, project, essay writing, artistic creativity, etc.

The student's study hours for each learning activity are given as well as the hours of non-directed study according to the principles of the ECTS.

<table>
<thead>
<tr>
<th>Tutorials (exercises)</th>
<th>1x13 = 13</th>
</tr>
</thead>
<tbody>
<tr>
<td>Laboratory exercises</td>
<td>2x13 = 26</td>
</tr>
<tr>
<td>Individual study, preparation and problem solving</td>
<td>3x13 = 39</td>
</tr>
<tr>
<td>Weekend study</td>
<td>3x13 = 39</td>
</tr>
<tr>
<td>Study during the 3 “empty weeks” (2 weeks of vacation and 1 week of exam preparation)</td>
<td>4x3 = 12</td>
</tr>
<tr>
<td>Course total</td>
<td>168</td>
</tr>
</tbody>
</table>

**STUDENT PERFORMANCE EVALUATION**

*Description of the evaluation procedure*

Language of evaluation, methods of evaluation, summative or conclusive, multiple choice questionnaires, short-answer questions, open-ended questions, problem solving, written work, essay/report, oral examination, public presentation, laboratory work, clinical examination of patient, art interpretation, other

Specifically-defined evaluation criteria are given, and if and where they are accessible to students.

- The language of instruction and examination is Greek. Special provisions (lecture notes and examinations in English) can be made for foreign students.
- The final grade is based on performance in the final examination (30% of the total grade) and course project (70% of the total grade). The evaluation criteria are posted on the course website.
- The final examination is written, of graded difficulty, and includes multiple choice questionnaires and short-answer questions (30%).
- During the course, students are assigned a web development project, requiring the application of multiple web programming skills (70%).

**(5) ATTACHED BIBLIOGRAPHY**

- Suggested bibliography:
  - Larry Ullman, “Illustrated Introduction to PHP 6 & MySQL 5”, Greek edition, 2009, KLIDARITHMOS
  - Freely accessible web resources and content, e.g. https://www.w3schools.com/

- Related academic journals:

This is an introductory course, hence there is no systematic use of articles from the scientific literature.