“Web technologies II” course description (2020)

Requirements for successful completion of the course

Darja Solodovnikova darja.solodovnikova@lu.lv
Faculty of Computing, University of Latvia

1. Course completion and grading

All the information needed for studies – homework and test announcements, information of changes in lecture schedule etc. – is published in course news forum in the e-Studies system. The forum is available here: http://estudijas.lu.lv/mod/forum/view.php?id=71857. Posts in this forum are automatically sent out to all students registered for the course. Students are responsible for entering a valid address in their LUIS profile so that they can read these email messages.

Minimal passing grade is 35% which gets rounded up to 4. All mandatory requirements must be fulfilled to successfully complete the course. Maximum grade 10 is awarded only for completion of all i-option requirements. The final grade is calculated using the following components.

Homework – optional, 20% of final grade

Two homework assignments are given during the semester, each 10% of the final grade. Homework dates are announced in advance and approximately 1 week is given to complete the assignments. If deadline is not met, homework may still be submitted, but maximum grade is decreased by 50%.

Submitted homework solutions are compared with each other to find similar submissions. In case of plagiarism (submissions are identical or very similar), grades are canceled to all students involved in plagiarism and students must submit a statement of reasons. In case of repeated plagiarism, the statement is escalated to the Dean of the faculty.

Tests – optional, 20% of final grade

Two in-class tests take place during the semester, each 45 min long and provides 10% of the final grade. Test dates are announced in advance. Any printed and handwritten materials can be used but no electronic devices are allowed.

One exam date and specific time will be provided when tests can be taken in case the student could not attend the original test because of some justifiable reason.

Practical assignment – mandatory, 60% of final grade

Students have to develop a server-side web application which demonstrates student's understanding of web programming and MVC solutions in particular. Students may create the application all alone or in student groups no larger than three participants.

The assignment consists of two parts:

Description of practical assignment – optional, 10% of final grade

The description has to be written during the term and handed in at a specific date.

Implementation of practical assignment – mandatory, 50% of final grade

The student (or the group of students) implements the intended web application and presents it in person during the summer exam session. Practical assignment gives 50% of the overall score. Students are required to receive at least 20% for this component.

Laboratory classes – optional, bonus + 10% of final grade

There will be 4 laboratory classes during the semester. The classes will be held in the computer lab. During the laboratory classes, practical tasks will be given. For successful completion of all 4 practical tasks, 10% bonus to the final grade will be awarded.

Course questionnaire – mandatory, 0% of final grade

Students have to fill out the course evaluation questionnaire in LUIS. It is technologically impossible to submit a final grade without filled questionnaire. Students who ignore this step will automatically fail the course.

Lecture attendance – optional, 0% of final grade

2. *Approximate* course schedule

The schedule may change during the semester, all changes will be published in advance in the course forum.

Weeks 1-3: Lectures

Week 4: Lecture, Laboratory classes

Week 5: Lecture

Week 6: Lecture, 1st test, Laboratory classes

The 1st test validates students' understanding of web protocols and web servers as well as basic constructs of PHP programming language.

Week 7: No lectures, Submission of the 1st homework

This homework improves your overall PHP programming skills. It concentrates on how you perform simple tasks in PHP (reading from file, working with strings, working with HTTP headers).

Weeks 8-9: Lectures

Week 10 (after holidays): Lecture, Laboratory classes (groups 5-7 on Wednesday)

Week 11: No lectures, Laboratory classes (groups 1-4 on Monday)

Week 12: Lecture, Laboratory classes (international students), Submission of practical assignment description

Week 13: Lecture

Week 14: No lectures, Laboratory classes, Submission of the 2nd homework

2nd homework: This homework is an exercise of application of MVC principle to real-life situations.

Week 15: Lecture, 2nd test

2nd test: Application of MVC principle, working with databases, AJAX.

Week 16: Lecture

3. Details of practical assignment

3.1. Requirements of practical assignment

The assignment has to be created in PHP programming language using an MVC framework (Laravel used in lectures or any other PHP MVC framework, e.g., FuelPHP, CodeIgniter, Kohana, Zend Framework, CakePHP, Symfony, Phalcon). Alternatively, you can use any other programming language. The only requirement is that it supports MVC and you utilize it. Examples of acceptable choices - Django, Ruby on Rails, ASP.NET MVC.

The assignment has to be created all on your own or in a 2-3 student team. Team work is welcome but has additional requirements. First, there must be **a complete git history** with all commits made by **all team members**. Second, only non-trivial and potentially viable web projects are accepted. Third, size of the project in terms of LOC and invested man-hours should exceed size of n*individual projects where n is number of members in the team.

3.2. Exam procedure

There will be 4 exam dates during the session. No more than 35 students (or student groups) are admitted per examination day. Students have to apply to a particular examination date in e-Studies system prior to the exam.

Application to a particular examination date is binding for the student - applying and not appearing without justifiable reasons on the date of exam is graded as N “neieradās”. If the student cannot take the exam on the date he has applied to it is possible to switch places with some other student, who has applied for a “better” date.

Practical assignment can be presented on one's own portable computer or on desktop computers available in examination room. Approximately 20 minutes are given for the presentation, this includes both demonstration of the functionality developed and source code review.

3.3. Grading

Basic requirements - 30% of final grade

- The system developed conforms to the description.
- The system has to be developed by the student (or group of students) and not downloaded from GitHub or any other source.
- System data is stored in a database (typically a SQL or NoSQL storage). If the database is relational, it must contain not less than 4 related tables and must be designed in 3rd normal form.
- The system conforms to MVC approach. User interface layer is separated from business logic and from data storage using appropriate MVC means.
- The system developed is non-trivial (not blog with 1 form), by adding some additional effort, one could create a usable web application. All 4 CRUD functions (create, read, update, and delete) must be implemented.
- Localization pattern is applied in the system, users can switch languages.
- The system has user authentication mechanism. Each registered user is included in some role, permissions are assigned to user roles.
- User passwords are not stored in plain text.
- HTML and CSS validates. It is requirement from “Web technologies I” course.
- UTF-8 text encoding is used throughout the system.

- ”Nice URLs” are used in the system (addresses do not contain script names, e.g., index.php, GET parameters are not used when not appropriate).
- Version control is used. There are at least 3 commits with different timestamps and different functioning versions of the developed system. Git is expected to be used if you don't have previous experience with version control. Indicates that development took at least 3 different sessions and was not committed only after completion.

Advanced requirements – 10% of final grade

Fulfilling any single requirement provides no more than 5% of the final grade:

- The system provides a personalization mechanism (e.g., in a social network system user may choose what's shown on the first page – most recent items or most interesting items).
- The system adapts to user actions by providing content suitable to current user (e.g., views like “my recent requests”, “other users who were looking for this book, opened this item as well...”)
- The system stores audit logs of user actions. Third party plug-ins are not evaluated for this requirement.
- The system adjusts to user's “accept-language” and automatically provides content in the most appropriate language.
- The system includes communication interfaces other than standard web / HTML (for instance, email notifications, RSS or ATOM feeds, RESTful interface, SOAP web services).
- The system utilizes file manipulation (upload, processing, display / download, e.g., image gallery, data import, file format conversion).

i-option requirements – 10% of final grade

In addition to the basic and advanced requirements, the following 3 requirements must be fulfilled:

- The system is deployed to an Internet-connected host and demonstrated using delegated domain / subdomain in the browser on a separate computer.
- Protection against CSRF / XSRF attacks must be built into the system.
- The algorithm or method used for protection must be described in the description students submit in mid-April.