Instructions for practical tasks and coursework from "WEB design technology"

This document describes the "WEB design technology" list of tasks, instructions, individual subtasks and coursework details. Tasks estimation is 40 points from 100. Coursework has its own estimation – 100 points.

TBD

REFERENCES

TBD

IMPORTANT:

1) **ALL TASKS ARE ONLY FOR STUDYING PURPOSE. DO NOT USE IT IN REAL LIFE. IT IS ILLEGAL.**
2) All software must be under free license.
3) Estimation will be done in the rate system: whoever delivers the task first will get the biggest point (students number in group), the last one will get 1 point. And it will be maximum points. Each mistake in work will decrease these points. If two or more work will be presented in one day then points will be decreased as well (I work - maximum points, II work - maximum point minus 1, III work - maximum point minus 2 and so on). Task and report will have different estimations.

Prohibited:

- jQuery;
- Wordpress;
- Joomla;
- Other CMS;

General requirements:

XAMPP

MySQL

PHP

Dictionary:

Author: andrii.s.spitzer@lpnu.ua

<mark> - must be changed to specific student data without “<>”

DB – database

Test report – list of tests, tests details (steps, input data, and expected data), status, and coverage

AI – Artificial Intelligence

BEF – both education form

EEF – external education form

FEF – full-time education form

UT – unit tests

i-face – interface

.

Table 1. Tasks by students:

Student number	fraud
1	fishing
2	DDOS

Tasks:

Required steps for all tasks:

1. Add/modify UTs
2. Create/Modify the **ci** script to build, test and run project.
3. Create/Modify the **YML** file.
4. Commit changes to **feature/develop/<task number>**.
5. Update **README** with details:
   - how to build project;
   - how to run project;
   - Version number.
6. Create **GIT TAG**:

   <PROJECT NAME>_<task number>_<VERSION>_ww<YYWWD>

   - YY – current year;
   - WW – work week;
   - D – current day number of weak.
7. Create pull requests with name **<task number>** and submit **lecturer** as reviewer.
8. After the reviewer approved – merge into the develop branch.

   Acceptance criteria:

   Project in the develop branch.

Task 1. Initiate GIT repository:

1. Create a github repo with the name "**web<*YY_1YY_2*><*group*><*student's full name*><*student's number*>**", main branch develop.

   YY_1 – Start studying year

   YY_2 – End studding year

   For example: 2019-2020 -> **1920**

   Full example **web1920ki47spitzeras03**

2. Add access for the author.
3. Create a branch **feature/develop/<task number>**. For example **feature/develop/task1.**
4. Create **README** file with:
   a. details about repo;
   b. task details;
   c. student number and details from **Table 1. Tasks by students**;
   d. add details about technology, program language.
5. Create **GIT TAG**: *<PROJECT NAME>_<VERSION>_WW<YYWWD>*
   - YY – current year;
   - WW – work week;
   - D – current day number of weak.
6. Create pull requests with name **task1** and submit **lecturer** as reviewer.
7. After the reviewer approved – merge request into **develop** branch.

Task 2. Create website business card:

1. Create a simple website business card with your personal data or information about your project. Use POST and GET methods to navigate pages or send forms to the server. Use AJAX method to load parts of the page depending on user inputs.
2. Required steps.

Task 3. Deploy:

1. Deploy site to hosting.
2. Required steps.

Task 4. Authentication:

1. Add authentication using:
    a. GOOGLE api.
    b. login and password
2. Required steps.

Task 5. Create fraud site:

1. Create fraud site according to table#1.
2. Required steps.

Reports:

1. title (see below)
2. task description and details
3. theory
4. implementation details
5. summary
6. referencies

МІНІСТЕРСТВО ОСВІТИ ТА НАУКИ

НАЦІОНАЛЬНИЙ УНІВЕРСИТЕТ ЛЬВІВСЬКА ПОЛІТЕХНІКА

ТЕХНОЛОГІЇ WEB ПРОГРАМУВАННЯ

<task#><task name>

Виконав:

<group number>

<student name>

Прийняв: <mentor name>

<YEAR>