COMP3207-1516-submit

From NotesWiki

Contents

COMP3207 2015/16 Group Coursework - Submission

Team Leader

The 'Team Leader' responsible for the ECS Handin submission and the organization of the demonstration is the first member of the team as specified here (https://secure.ecs.soton.ac.uk/student/wiki/w/COMP3207-1516-cw2teams) . (Teams may adjust the order of names if they decide on a different submission and demonstration Team Leader.)

Marks Distribution

Team members should agree between themselves as to how the marks awarded for the team submission will be divided between the team members (see below for instructions on how to proceed if this is not possible). The Team Leader should print out the form here (https://secure.ecs.soton.ac.uk/notes/comp3207/1516/ejz/mdr7.pdf) , complete it as agreed and arrange for every member of the team to sign and date it. The completed signed form must be submitted via the ECS Handin system (see below). An incomplete form (e.g. with missing signatures) means that the entire ECS Handin submission is incomplete and therefore subject to penalties.

Teams are encouraged to split the work evenly between all team members (in which case the marks split evenly). They are advised to consider any proposed non-linear distribution very carefully before submission. Note that an individual contribution of zero is acceptable and will result in that team member being effectively removed from the team. One or more individual contributions of 10% or less may result in an *ad hoc* reduction in the effective team size. Any proposed non-linear distribution will be discussed with the team during the presentation and may be subject to modification by the Module Leader at that stage.

ECS Handin Submission

This is a combined paper and electronic handin. One submission only from each team is required - this should be submitted by the Team Leader. The paper submission consists of the completed Marks Distribution Form and a printed copy of the five-page report. The electronic submission consists of a zipped archive file ('comp3207.zip') containing:

- 'team_x.pdf' (the five-page report)
- 'source.zip' (see below)
- 'notes.txt' (text file) (optional: include only if there is any information affecting the assessment of the team submission that the Module Leader needs to know - this replaces the similar ECS Handin facility, which should not be used. Note that this file does not replace the right of individual students to make a Special Considerations Board submission if they believe that any non-academic factors (e.g. illness) have affected their work)

'source.zip' must contain **all** of the source code used for the submission and *sufficient additional information* for any of the other teams to implement your demonstration using the files as submitted, as defined below.

Teams are advised that it would be good practice for all team members to check the 'comp3207.zip' file carefully before submission as any missing files can only be added by another submission via the ECS Handin system (and the possibility of late penalties if past the deadline).

Source Code

The files to be supplied will either be code written entirely by your team from scratch ('SourceA'), code that was written originally by somebody else and modified or adapted by your team ('SourceB') or code that has been written by somebody else and has not been modified by you at all ('SourceC'). The directory structure of the zipped archive must be as follows:

- SourceA/python
- SourceA/javascript
- SourceA/misc
- SourceB/python
- SourceB/javascript
- SourceB/misc
- SourceC/python
- SourceC/javascript
- SourceC/misc
- readme

The 'misc' directories should include code in any language other than python or javascript (e.g. HTML). The 'SourceB' files must be commented to indicate the modified lines and an estimate of the percentage of the files that have been modified included in the 'readme' directory. Other information (sufficient to enable any of the other teams to implement your demonstration on the GAE) must be included in the 'readme' directory. Where 'SourceC' material has been reliably archieved on the web, a URL can be supplied **provided** that this uniquely identifies the exact version of the code that has been used.

Please ensure that **all** unnecessary files and directories are deleted from your directory structure before submission (for example, .pyc files and .svn directories). File extensions should be the normal ones used (e.g. .py, .js, .html, .css, .yaml etc.) and in addition please ensure as far as possible that the contents of the file reflect the extension (e.g. try to avoid having a .py file that only outputs HTML code).

Source Code Size Restriction

This restriction is in terms of LOC ('Lines Of Code')- see Wikipedia (http://en.wikipedia.org/wiki/Source_lines_of_code) . The definition adopted here is physical LOC excluding comment and blank lines. A tool such as CLOC (https://github.com/AlDanial/cloc) will be used to measure the source code size.

The target size for this coursework is 2000 LOC. This is to be calculated using all of the 'SourceA' files, together with an appropriate proportion of the 'SourceB' files (using the 'modification percentage' mentioned above). The relevant statistics should be provided on page 3 of the report. There is no penalty for submitting code that is smaller than the target size, or larger up to a maximum of a total of 4400 LOC (an additional 120%). Submissions that are larger than this will attract a penalty *pro rata* to the LOC in excess of 4400 up to 6000 LOC, where a 100% penalty will be applied. While there is a long tradition of 'obfuscated code' in computer science, any use of such techniques to reduce the number of lines of code measured will affect the mark awarded for the quality of the code submitted.

Monday 11 and Tuesday 12 January 2016 Demonstration Structure and Schedule

To be supplied later.

'Failure to Agree'

Teams are advised to make every effort to agree on the marks distribution because failure to agree will be interpreted as demonstrating a general lack of competence. However, the procedure to follow if there is no agreement is set out below:

The team should divide into two or more subteams (in the worst case, a team of size 'N' could have 'N' subteams). Each team should elect a subteam leader, who should make a full submission as detailed above (i.e. including all source code). Each marks distribution form submitted should indicate proposed percentages of the overall team marks to be allocated to the members of that subteam, with a written one-page explanation of why such an allocation would be appropriate. It should be noted that any attempt by a team member to exploit the advice above (that teams should make every effort to agree) by, for example, refusing to sign the marks distribution form will not be successful (in the unlikely event that this happens, each individual should make a brief signed statement as to the facts of the case and submit this with the other documentation).

The final allocation of marks for a team that fails to agree a marks distribution will be determined by the Module Leader, taking all relevant factors into account. The decision of the Module Leader will be final.

Changes to the software after submission

No change to the software can be made after submission. Not even 'cosmetic' changes (e.g. to comments) can be made (i.e. all filestore timestamps must be before the submission time). The code demonstrated must be identical to the code submitted and teams will be asked to certify this.

Questions

Please address any questions to EJZ

- This page was last modified on 18 November 2015, at 17:42.