CNSCC.212: Advanced Programming – Coursework

The coursework is a mini-project that assesses the students' ability to use the programming languages learned. It is a group-based work with 4-5 students per group.

The aim is to design, implement and demonstrate the functionality of a Digital Library for Computer Science Research such as arXiv, IEEE Xplore and ACM Digital Library. The basic workflow is as follows:

1. Authors affiliated to universities upload PDF papers for full peer review.
2. The system allocates 3 reviewers automatically based on a student-designed approach (e.g., based on the paper's abstract and keywords).
3. The system only allows allocated reviewers to download papers for review, input their comments and submit their decision (i.e., Accepted or Rejected). No other users can do the aforementioned actions.
4. The system automatically determines whether a specific paper is published in the repository of the universities where the authors are affiliated. This is done if and only if the paper is accepted by at least two reviewers.
5. The system allows CRUD operations for universities, authors, papers and reviewers. CRUD means Create, Read, Update and Delete.
6. The system automatically certifies a review (e.g., using a digital signature/stamp included in the paper published). This is done with the aim of guaranteeing that the paper has passed through a formal peer-review process.

Deliverable 1: System Presentation (10 slides maximum -excluding title pages- and 25 minutes maximum -including system demo-)

The task is to create a video presentation to explain the following three artifacts:

- System design (can use the same group slides but individual narration). The students can provide use cases, class diagrams, activity diagrams, etc.
- System demo (individual explanation using either a video or multiple screenshots).
- Explanation of individual contribution (contribute to design, implementation and demo).

Deliverable 2: Source code in two zip files (entire project + contributed code)

- The system must be implemented using at least Java and JavaScript.
- The use of existing frameworks/APIs is allowed.
- We encourage the use of version control systems such as Git.
- Code should be easy to read (use comments).

Assessment

- System design and implementation **[30%]**
- System demo **[20%]**
- Explanation of individual contribution **[50%]**

Deadline: 07 May 2021 EOD