Conference management system

I. The application

The process of managing information required for organising a conference is a time-consuming activity. In order to produce and manage this information, using a dedicated software application is from far the best solution.

II. Functional requirements

1. The system should allow its users (chair, reviewer and author) to sign in using their user name and password.

2. The system should allow its users (chair, reviewer and author) to upload their personal information.

3. The system should allow the chair to submit basic conference information, which includes conference name, URL, subtitles, and main organiser's contact information.

4. The system should allow the chair to set and modify the deadline for paper submission, paper review, acceptance notification, and the uploading of accepted paper (camera ready-copies).

5. The system should allow the chair to set up the topics of interest.

6. The system should allow authors to submit an abstract of a paper in addition to the paper title, authors, their emails, addresses, keywords, and the topic of interest that applies to their paper.

7. The system should allow authors to upload the full paper in a specific format (e.g. PDF file or Word file).

8. The system should allow reviewers to specify the topics that falls into his/her area of expertise.

9. The system should allow reviewers to bid for papers that they are interested in reviewing.

10. The system should allow reviewers to indicate any conflict of interest.

11. The system should be able to assign papers to reviewers automatically.

12. The system should allow a reviewer to submit an evaluation of a paper that was assigned to him/her.

13. The system should allow the reviewer to submit a special comment that can be read by other reviewers.

14. The system should allow the chair to make the final decision on accepting or rejecting a specific paper.

15. The system should allow the chair to assign an accepted paper to a specific conference session.

16. The system should allow the author to upload the camera-ready copy of an accepted paper

III. The implementation

As the SE course proposes an object-oriented approach, it is required that both the domain model and the solution model to be object oriented. There are mandatory requirements regarding the artifacts produced during the software development life cycle like:

- Use Case diagrams and documents, describing the functional model
- Class diagrams for the analysis and design model
- The application will be multilayer structured: User Interface, Business Logic and Database. Object orientation concerns mainly the Business Logic
- By its nature, this is a web application. So, it is recommended that UI provides browser forms specific for each user. The front end functionality will be responsible to gathering data, transmission, and appropriate king of validations. The database and the business logic will be resident on the server.
- A particular attention will be given to the validation process by using assertions (pre & post-conditions, invariants) also to the testing phase, including the techniques of selecting (generating) tests data.
- Also the usage of design patterns in the late design model will be appreciated. The same attitude related to capturing and accessing the rationale of the system.
- Software production is a team activity. So, specifying the different tasks to each team member, analysing and managing the risks of the development will be considered.
- The teams should use Git for version management

The system should:

- be able to display information & perform actions in less than 5 seconds
- generate the documents mentioned in the functional requirements within 20 seconds
- must use validations for usernames/emails to be unique and for password to be securely stored
- ensure the the user's personal information is confidential, and only the user can see it
- ensure the user's personal & authentication information cannot be modified by any other person
- be able to restart after a failure
- be able to back-up data and recover the data using the backups
- generate fault reports & various logs (both for user behaviour & error logging)