1. Introduction

1.1 Document purpose

The purpose of this document is to serve as a guide by which a scientific conference management system will be built.

1.2 Project scope

The purpose of the Conference Management System is to support the automatic management of information related to scientific conferences. The system is based on a relational database with its paper submission, management and evaluation functions and conference planning features. This database will be server-based. We also aim to provide a pleasant user experience with an easy-to-use GUI.

2. Overall description

2.1 Product perspective

- Participant details

These include the participant's name, email, affiliation for regular participants. The committee members (chairs and co-chairs) are required to also give the following information: personal website, CMS username and password.

- Paper details

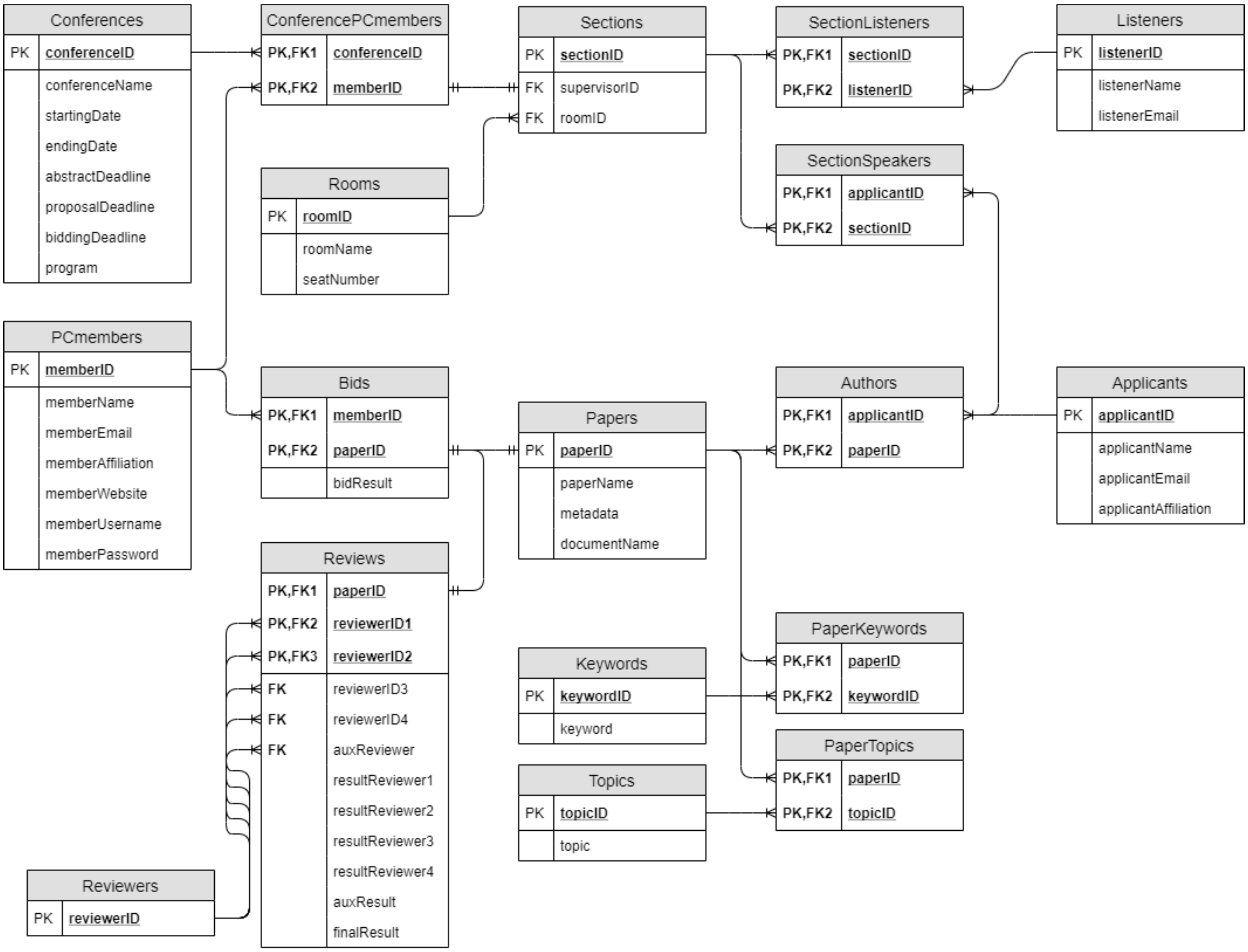
These include name, keywords, topics, and co-authors (if any) with their metadata. This information will be required for the reviewing process. The applicant may also upload a PDF or MSWORD document if he/she so desires.

- Conference details

These include name, starting date, ending date, the call for papers, the deadlines, the committee members, the sections and the program.

2.2 Product features

The major features of the conference management system will be implemented respecting the structure of the below entity-relationship model.



2.3 Functionalities available to users

The amount of control given to a user over the conference’s management depends on his/her role in it.

Authors have the following functionalities:
- submit papers during the “Call for Papers”
- upload a PDF or MSWORD document of their paper
- log into their account
- create an account

Speakers will be able to:
- view the reviewers’ opinion
- improve their paper
- upload a presentation in PPT(X) format
- choose a section to present in

Reviewers are able to:
- view all papers that have been assigned to them
- review a given paper
- view the other reviewer’s evaluation of a paper he/she finished rating

Functionalities available to PC members:
- modify deadlines
- view all papers
- bid on each paper
- assign a paper to each reviewer
- approve/disapprove/send back to review all papers with mixed review results
- view all sections he/she can supervise and choose one
    - options include only the sections they do not present in (if such is the case)
- every functionality available to authors

Listeners will have the following options:
- view all sections
- choose sections to supervise

2.4 Other functionalities

- the system will send an email to all authors informing them of their paper's review
- the system will automatically select the best room for a section to take place in based on its number of attendants

2.5 Operating environment

- distributed database
- client/server system
- operating system: Windows
- database: PostgreSQL, SQLAlchemy
- platform: Python

3. System features

Description

The Conference Management System maintains information on the user's personal data, papers review status, conference data and structure. Aside from the user functionalities mentioned above, the application will also contain a logic system for assigning a room to each section.

Client/Server system

The application will adopt such a system in order to properly manage the interaction between its users and the database. Of course, all the data will reside on the server-side (back-end) and will be managed via a DBMS. The client-side (front-end) is represented by a desktop client with a GUI and that is responsible with running the functionalities corresponding to each type of user.

4. External Interface Requirements

4.1 User interfaces

- front-end software: Python
- back-end software: Python, SQLAlchemy, PostgreSQL

4.2 Hardware Interfaces

- some version of Windows.

4.3 Software Interfaces

- Operating system: Windows – because it is ubiquitous, user-friendly and offers support for a wide range of technologies.
- Database: PostgreSQL – due to our team's already-existing experience with it, SQLAlchemy
- Python: To implement the front-end. It was chosen because it is widespread and flexible.