



CS209 SOFTWARE ENGINEERING

# CONFERENCE MANAGEMENT SYSTEM

**SUBMITTED TO:**

MR. ROHIT BENIWAL  
Department of Computer  
Science and Engineering

**SUBMITTED BY:**

ARIHANT JAIN (23/CS/075)  
DIVYAM GOYAL (23/CS/145)

# TABLE OF CONTENTS

Introduction

Requirements

Use Case Diagram

Context Diagram

ER Diagram

Class Diagram

Reference





# INTRODUCTION

Conference Management System is a web-based software product catering to the arrangement and coordination of academic and professional conferences on a large scale (like EasyChair or Microsoft CMT). The system is dedicated to providing end-to-end solutions for the management of the complete lifecycle of a conference, particularly paper submission, review processes, scheduling, and participant management.

This application will allow conference organizers to set up details of their events, such as deadlines, tracks, and themes. It allows participants (authors) to submit a paper, or proposal directly in the system and reviewers to evaluate submissions efficiently.



# REQUIREMENTS

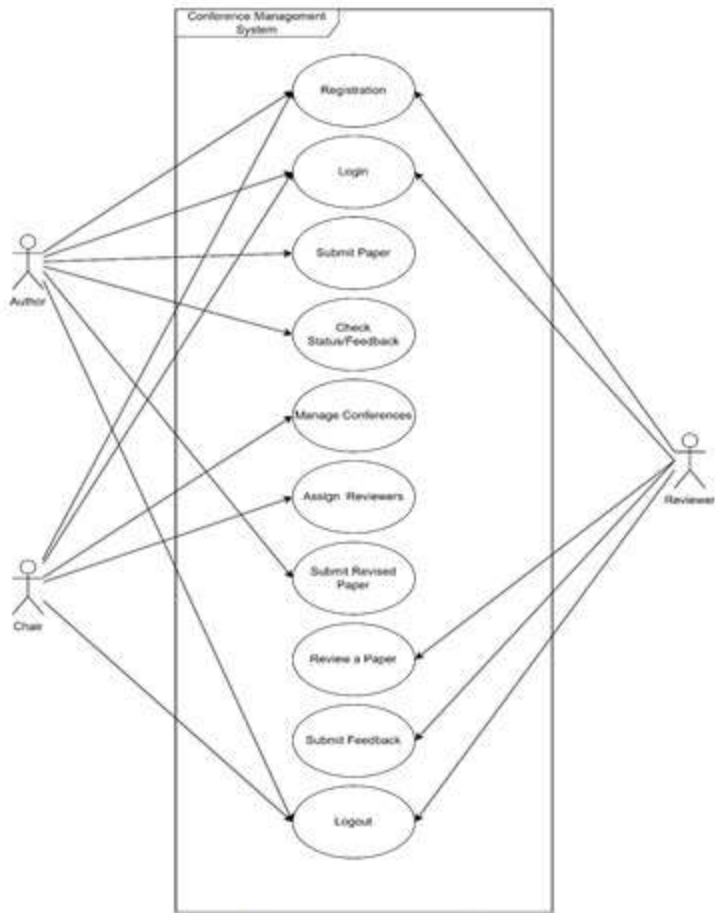
The Software Requirements Specification (SRS) for a Conference Management System outlines the functional and non-functional requirements necessary for developing the system. This document serves as a blueprint, ensuring all stakeholders understand the expectations and functionalities of the system.

Functional Requirements are Login/ Logout , Download Paper, Submit Paper, Manage Conference, Automated notifications for submission deadlines , review deadlines.

Non Functional Requirements are Performance, Reliability, Scalability, Adaptability, Portability etc.

# USE CASE DIAGRAM

The Conference Management System use case Diagram illustrates the interactions between the system's actor— Author, Chair, and Reviewer and the core functionalities provided by the Software. Each actor has specific access to modules based on their role, ensuring that they can platform actions relevant to their purpose within the portal. This also ensures data protection and data encapsulation in this Software.



Use Case Diagram



# USE CASE DIAGRAM

## ACTORS:

- **Author:** Interested in submitting a paper, by completing a form asking for details about it, in order to obtain the review of qualified people. After the review period has passed .If it is positive, then paper will be presented to the public .
- **Program chair:** Interested in having the papers review by qualified peoples.
- **Reviewer:** Interested in analyzing and review new ideas.





## FEATURES:

- **Registration:** The User (Author, Reviewer, Chair) information is put into the database. Once registered, the user can log in to the Conference Management System and access its other functionalities.
- **Login:** The login process ensures secure access to the system and requires user credentials.
- **Submit Paper:** Authors invited to a conference will be able to submit their papers.
- **Check Status/Feedback:** It enables authors to monitor the status of their submitted papers and access feedback provided by reviewers .
- **Manage Conference:** It enables organizers to oversee and manage all aspects of a conference.



## FEATURES:

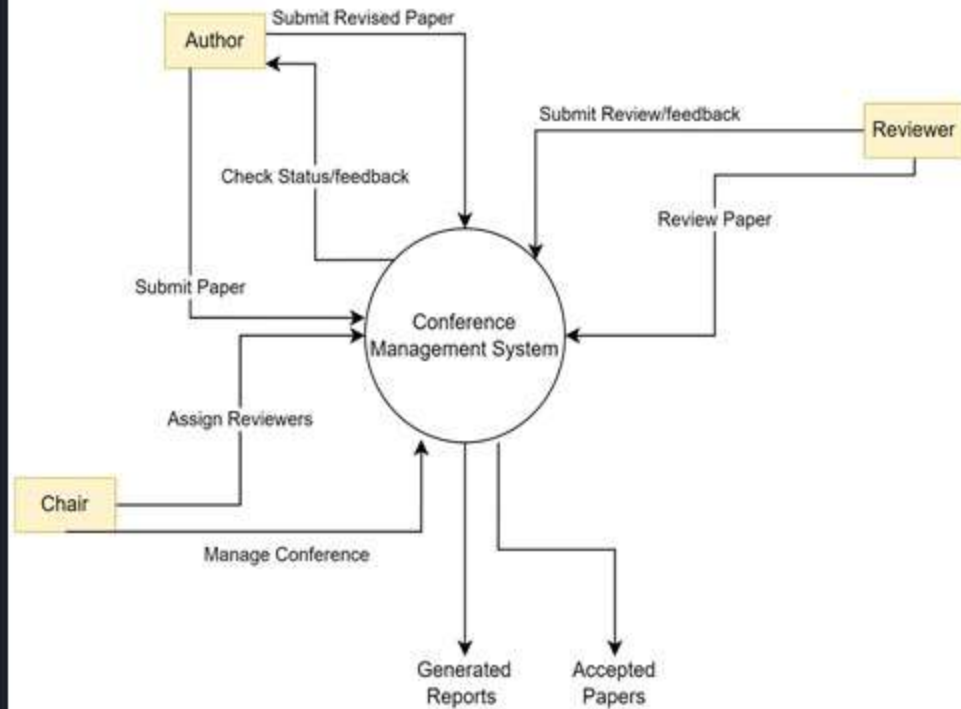
- **Assign Reviewers:** It enables program chairs to allocate submitted papers to suitable reviewers based on their expertise.
- **Submit Revised Paper:** The "Submit Revised Paper" feature in Conference Management System allows authors to upload updated versions of their papers in response to reviewer feedback.
- **Review a Paper:** It allows assigned reviewers to evaluate and provide detailed feedback on submitted papers
- **Submit Feedback:** It allows reviewers to provide detailed evaluations of submitted papers.
- **Logout:** It enables users to safely exit the platform once their tasks are completed. It ensures that their session is fully terminated, safeguarding sensitive data from unauthorized access.



# CONTEXT DIAGRAM (LEVEL-0 DFD)

This is a Level-0 Data Flow Diagram (DFD) for an Conference Management System (CMS), depicting the overall interactions between external entities and the system and the data processing by the system.

This provides an abstract overview of the data flow of the software project as a whole.

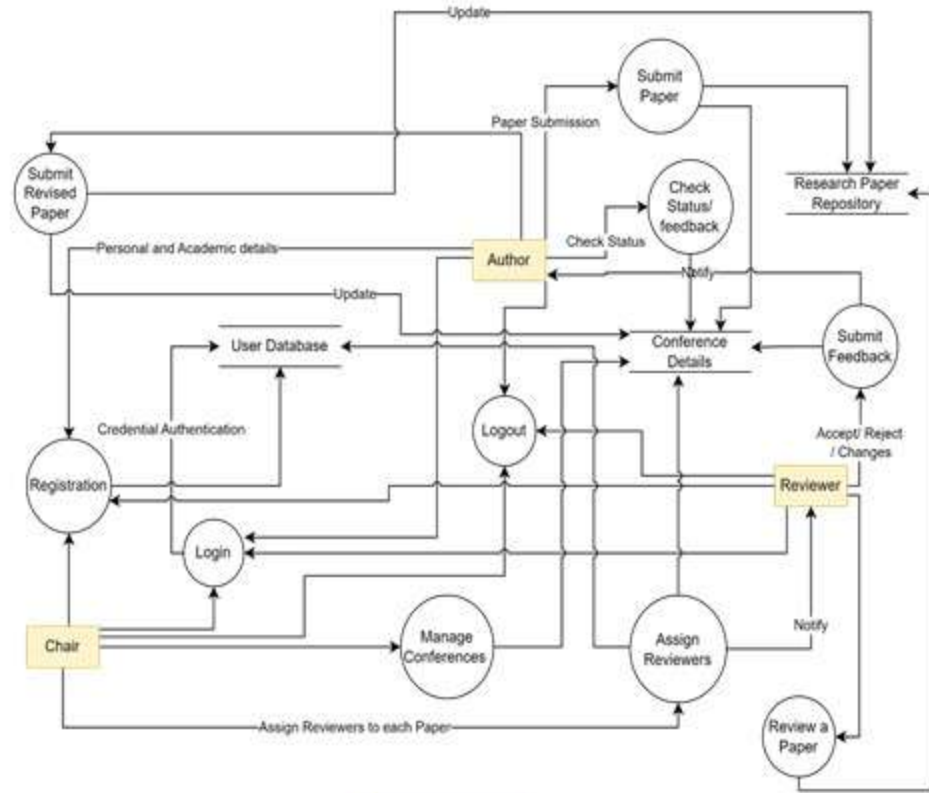


Level 0 DFD

# CONTEXT DIAGRAM (LEVEL-1 DFD)

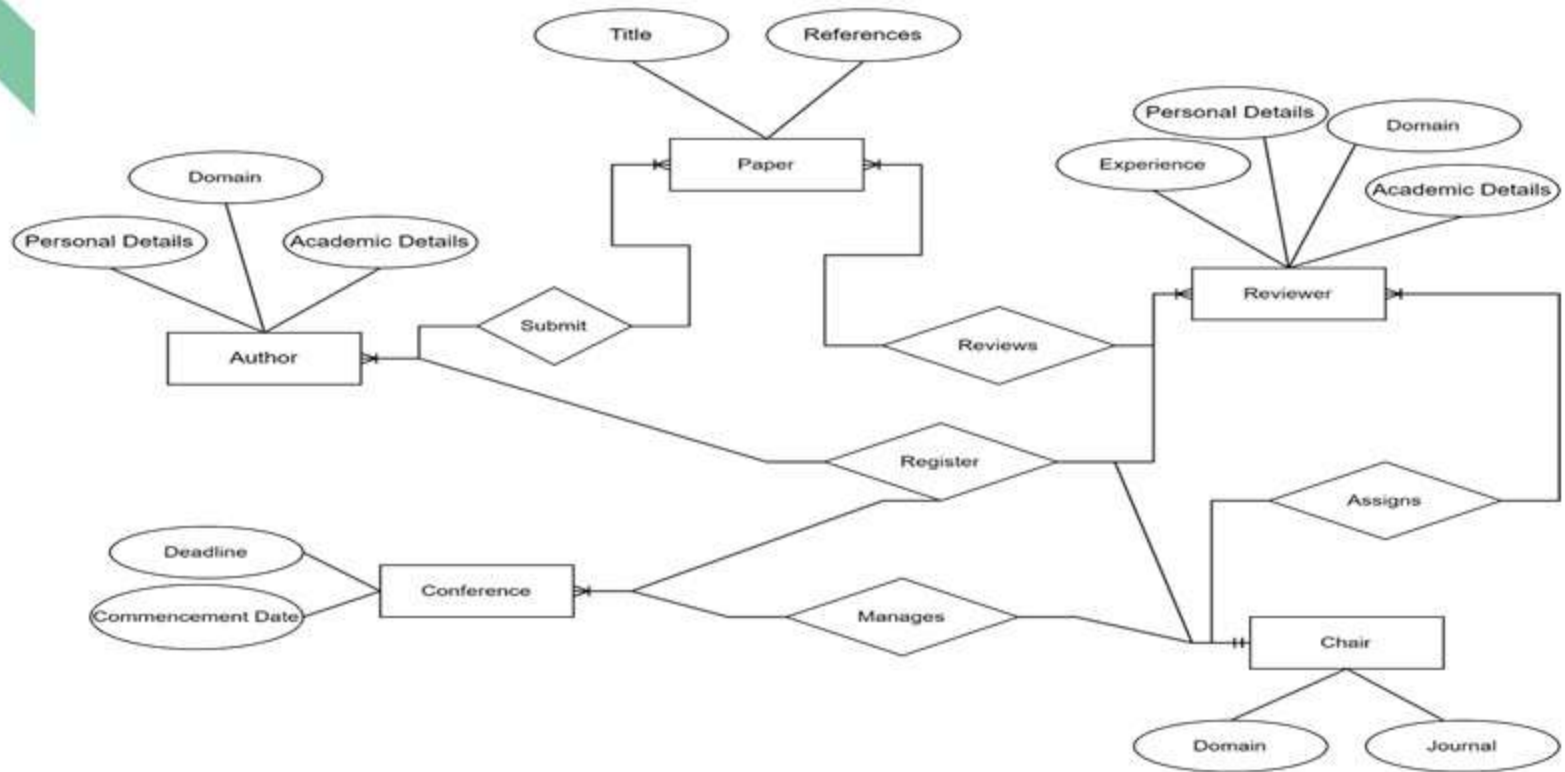
This is a Level-1 Data Flow Diagram (DFD) for an Conference Management System (CMS), depicting the overall interactions between external entities and the system and the data processing by the system.

This is a more in-depth view of the system which elaborates on the various modules and their functionality, it is an extension to the Context Diagram described previously.



Level 1 DFD

# ER Diagram

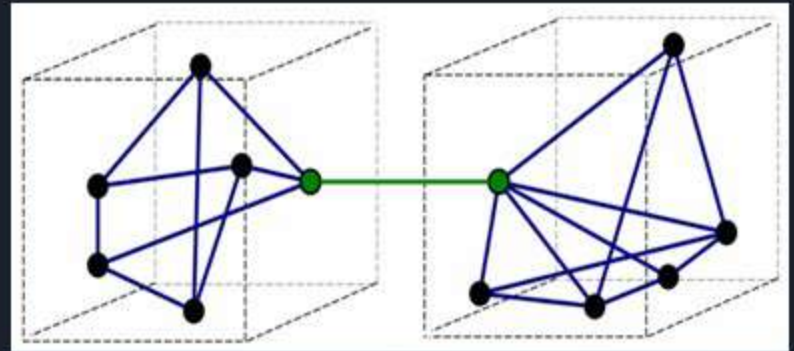


E-R Diagram

# DESIGN

The SRS created as product of the requirement phase, acts as input for the design phase, in which we try to accomodate following characteristics in order to make good quality software:

- Low Coupling: Components and modules are designed to minimize dependencies, ensuring that changes in one module have minimal impact on others.
- High Cohesion: Each module has a clear, focused responsibility, which enhances readability, maintainability, and reusability of the code.

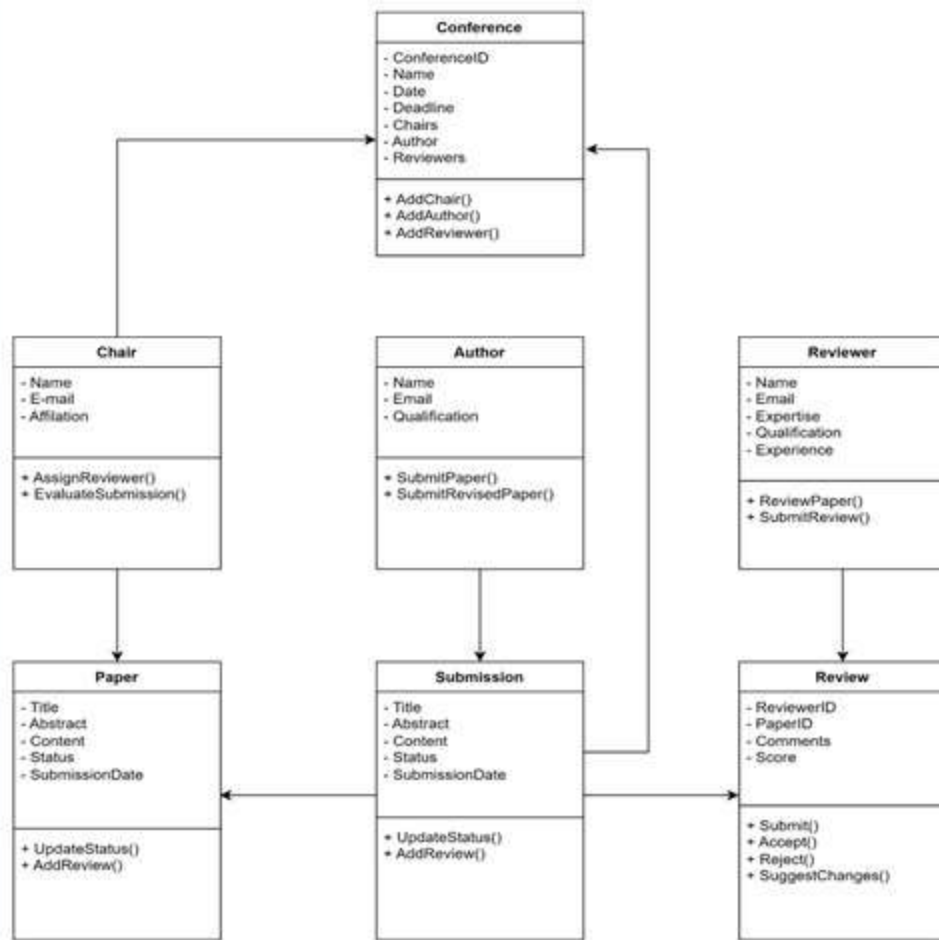


*Low Coupling and High Cohesion*

# Class Diagram

The class diagram for a Conference Management System represents the structure of the system, defining the key entities, their attributes, and relationships. It provides a blueprint for how the system's objects interact and encapsulate data and behavior.

The system includes classes like **Author**, **Reviewer**, **Chair**, **Paper**, **Submission**, **Review** and **Conference**.



Class Diagram



# REFERENCES

- EasyChair - <https://easychair.org/>
- Microsoft CMT - <https://amtint.research.microsoft.com/>
- IEEE Recommended Practice of Software Specification- IEEE std 830-1993.





Thank you!