Table of Contents

1. Introduction

1.1 Purpose

The document provides the layout for identifying the key architecture structures making up the system (SAM2023). The goal of the document is to establish the overall structure of the software system. The document intended to represent the link between design specification and actual design process.

1.2 Scope

This document serves as blueprint for both the system and the project development team and covers complete design description of SAM2023 System. Using this document even the stakeholders can take active part in the software development process. This helps in clear specification/understanding of requirements.

1.3 Definitions, Acronyms, and Abbreviations

SAM2023 - The system, hereafter referred to as SAM2023.
SAM - Software Architecture Mining.
PCC - Program Committee Chair.
PCM - Program Committee Member.
Admin - Administrator of SAM2023
User - Any entity that uses the SAM2023 system.
MVC - Model view controller

1.4 Overview

This document contains different architectural views including the logical view, implementation view and use case view for understanding the key structures of the application.

2. Architectural Representation

All the diagrams and descriptions use standard methods of representing design architectures which is Unified Modeling Language(UML).

3. Architectural Goals and Constraints

The SAM2023 provides a platform to submit and review the papers for the Software Engineering Conference. SAM2023 has different users and they perform different roles. Like the role of the Submitter/Author is to submit the papers and the role of the PCM (Program Committee Manager) is to review the papers submitted by the Submitter. Also the Administrator has all the control of the SAM2023.

The admin manages all the deadlines of SAM2023 and also manages the notifications to notify an user about a specific event. The PCCs (Program Committee Chair) manages the allotment of paper review process and also resolves the conflict between two reviews. The PCMs are not able to choose which document they would review, that’s one of the constraints of the system.

The system is going to be developed using a structured set of activities like specification, design and implementation, validation and evaluation. System will be developed using Visual Studio Code for the overall development process.

4. Use-Case View

4.1 Use-Case Realizations

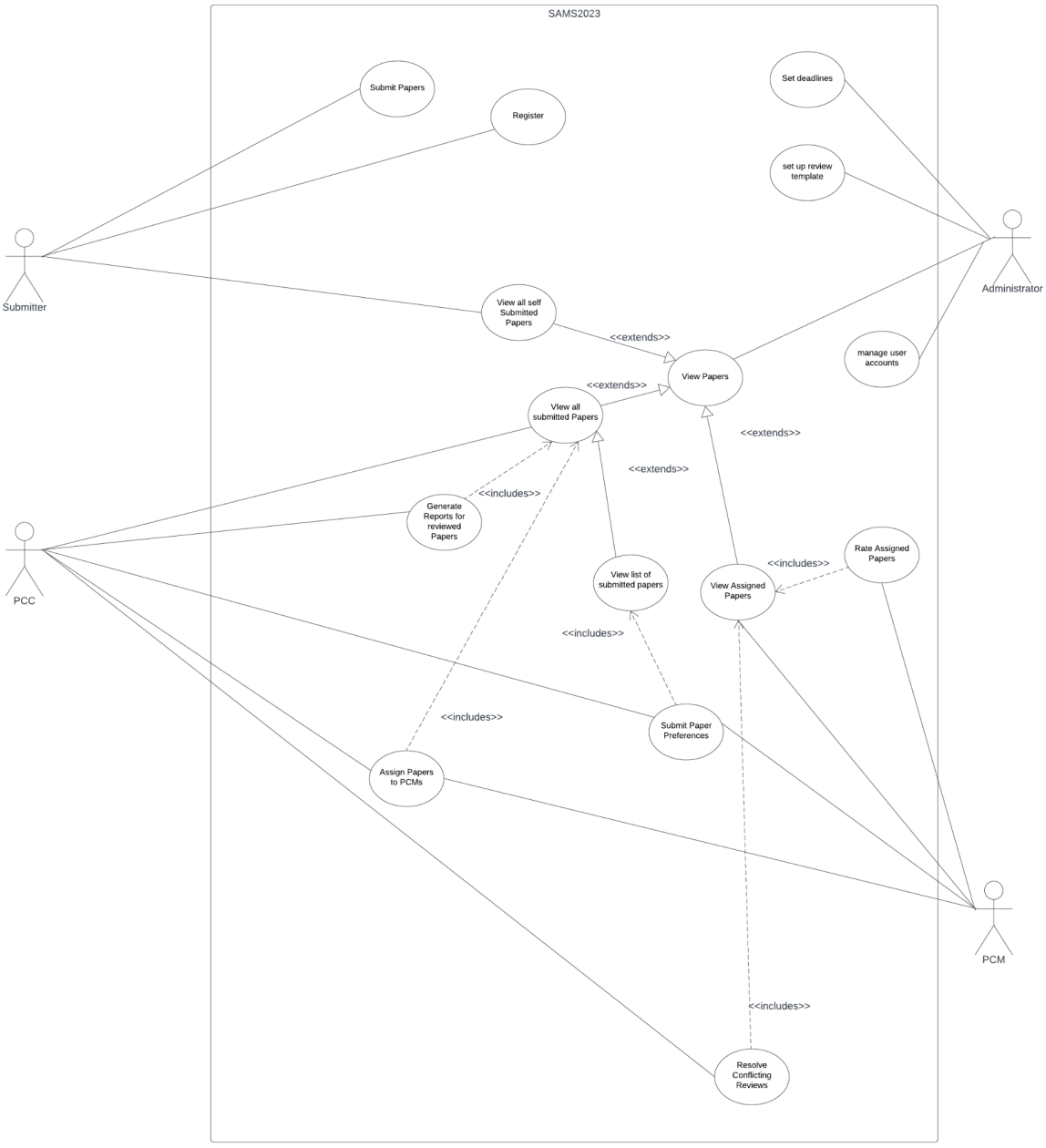


Figure 1 Use Case model

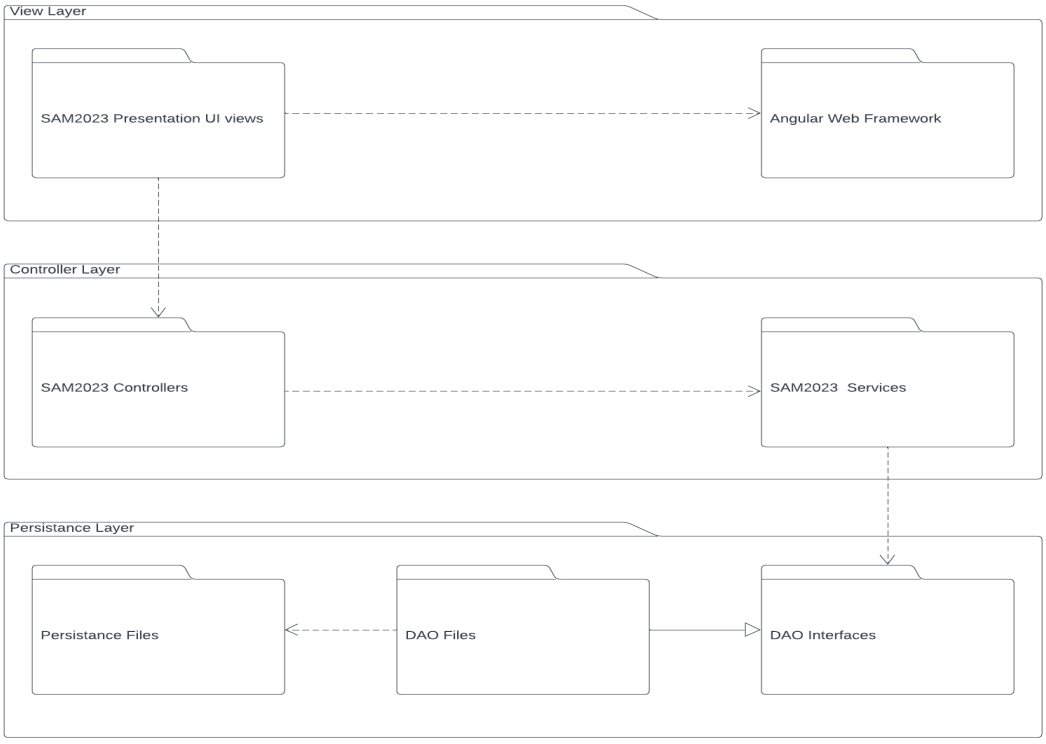
As seen in Figure 1 the system has 6 major components that interact with 4 different kinds of users. Each user has some functional limitations. The first object is the Paper which is submitted by the submitter. The PCM can also be a submitter and can submit the paper. The second object is the review submitted by the PCM on the submitted papers. The PCM can only review papers that are assigned to them by the PCC. Also PCMs cannot review their own submitted papers. The next object is the PCC generated report. Once the PCMs review the assigned papers, PCC generates the report based on the reviews. In case of any conflicting views the PCC can resolve by having PCMs discuss and modify the review. The next component is the templates that are used for review, report and all the alert notifications. This view is managed by the Administrator. The Administrator also manages the User accounts and has access to all materials in the SAMS2023 system.

5. Logical View

5.1 Overview

The logical view seeks to provide an understanding of the structure and organization of SAM2023. SAM2023 comprises multiple components and subsystems.

The figure below shows the high level structure of SAM2023 in a package diagram. The package diagram follows the MVC architecture. The package diagram is further realized into a component diagram that is presented in the implementation view.



The presentation layer is to be developed using the Angular Framework. The controller layer serves as an interface between the presentation layer and the persistence layer. The controller objects dictate the data the presentation layer receives based on the REST request response cycle. The controller objects make use of controller services.

These services comprise of the business logic that is required to run the application. The services implement the main flow of all use cases in the application. The persistence layer serves to store and access data relevant to SAM2023 in the form of a relational database.

6. Process View

The process view is not applicable for this project since the entire application runs on one single process and there does not exist process interactions

7. Implementation View

For implementation of our system a MVC architecture will be implemented. The backend api that contains the Models and the Controllers will be implemented using the java Spring MVC framework. The user may make curl commands to access the API directly and get data from and within the system. However, the user will usually access a HTML webpage which will be generated using the Angular framework. The HTML pages will implement all the user portals which will act as the interface between user and system. Figure 3 shows a component diagram that illustrates the MVC and several subsystems and models that work in the system.

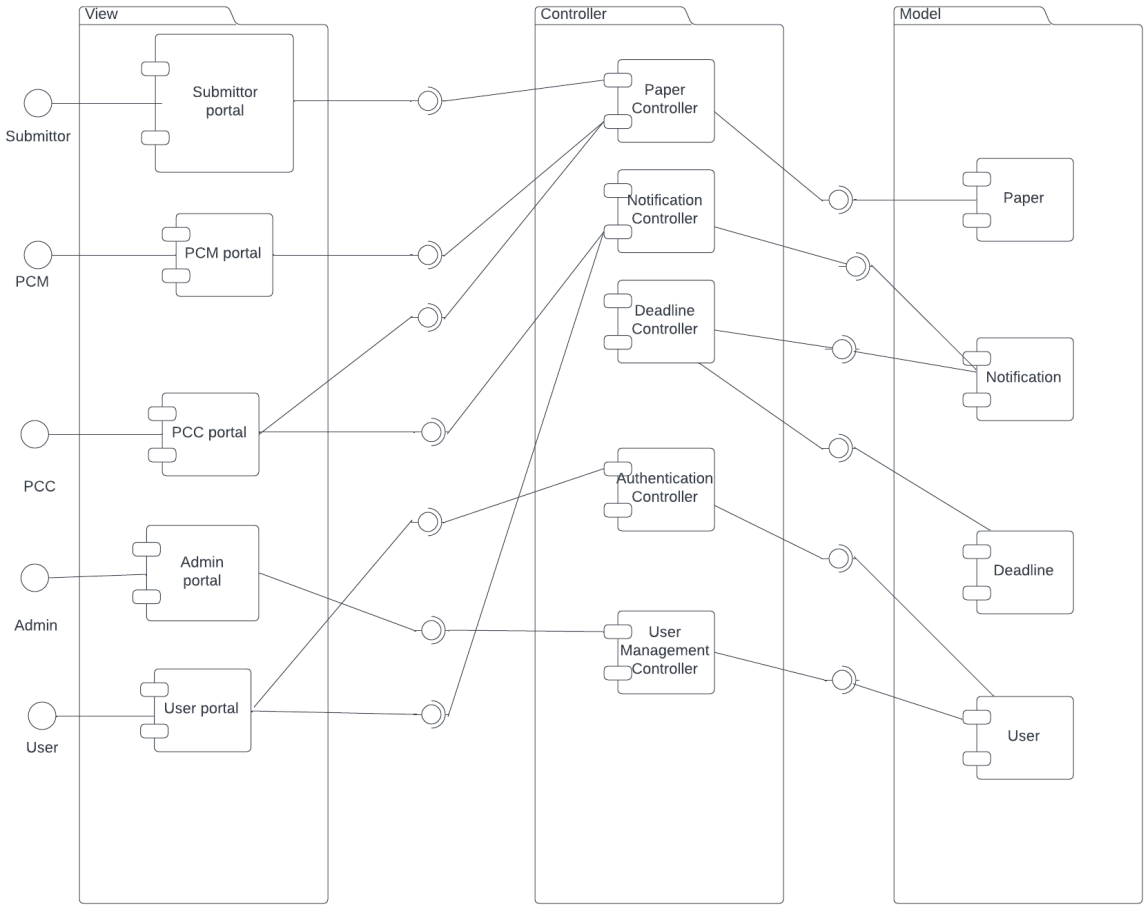


Figure 3 MVC component diagram

8. Deployment View

The system is designed to be deployed on a single server. The same server will be used by a user of the system to access and use the system. There may be future opportunities of deploying the system on separate servers and providing access to the end user globally.

9. Size and Performance

The initial system will be designed to be able to serve one user at a time. Scalability of the system has not been taken into consideration. However, the system will persist all user data which will be available to users for future use.

10. Quality

Quality of the system will be verified with the product owner on completion and improved id necessary.