Confy

System Design

1.0

27.12.2020

Ali Albaali

Uğur Kaya Küçükkaptan

Bekir Malik Özkuran

Atila Ersun

Prepared for

SE301 Software Engineering



Table of Contents

[1. Introduction 1](#_Toc433996772)

[1.1. Purpose of the System 1](#_Toc433996773)

[1.2. Design Goals 1](#_Toc433996774)

[1.3. Definitions, Acronyms, and Abbreviations 1](#_Toc433996775)

[1.4. References 1](#_Toc433996776)

[2. Current Software Architecture 1](#_Toc433996777)

[3. Proposed Software Architecture 1](#_Toc433996778)

[3.1. Overview 1](#_Toc433996779)

[3.2. System Decomposition 1](#_Toc433996780)

[3.3. Hardware Software Mapping 2](#_Toc433996781)

[3.4. Persistent Data Management 2](#_Toc433996782)

[3.5. Access Control and Security 2](#_Toc433996783)

[3.6. Global Software Control 2](#_Toc433996784)

[3.7. Boundary Conditions 2](#_Toc433996785)

[4. Subsystem Services 2](#_Toc433996786)

[5. References 2](#_Toc433996787)

SYSTEM DESIGN DOCUMENT

# Introduction

Confy is a conference management application that is coded using the Java programming language and runs on the Android Platform. Its design purpose is to be an easy to use and quick to work application that can be used in various ways. Confy users should be able to create meetings and share, review, and grade papers.

## Purpose of the System

Confy is a conference management application which users can post and review articles that are relevant to the conference topic. The main draw of Confy is it is easy to use system that allows authors to apply and upload their articles that needs approval and review, and easy to review and grade for conference reviewers. During the conference date, authors can upload their file and reviewers can review and assign their grades easily, then it can be decided if the article uploaded is approved or not. Overall, it is an easy process that simplifies the review of an otherwise complex topics about articles.

## Design Goals

**Usability**

* The users should be able to reach their profiles with one click.
* The users should be able to create conferences very easily.
* The users should be able to use the system without any training.
* The system should be easy to navigate and understand such that a user should be able to successfully complete their learning phase within 10 minutes on their first time.

**Performance**

* The system should be able to create a new user profile within 5 seconds.
* The system should be able to create a new conference less than 10 seconds.
* The feed page load time of the system should be no more than 6 seconds.
* The system should log in a user within 8 seconds.
* The system should not lose any user data.
* The system should be available 95% of the time.

**Reliability**

* Users’ password is not displayable for everyone except the owner.
* When the system fails to create a conference, the user should be able to retry creating it within a very short time.
* The system should be able to prevent restarting on failures 90% of the time.

**Supportability**

* The system should support the English language.
* The system should support the Turkish language.
* The system should run on the Android system.
* The database of the system should be able to support 15% of growth without losing any performance.

**Implementation**

* All software codes associated with Confy will be written by the usage of Java.
* Database management will be done by Firebase Realtime Database.
* The system should be written in Android Studio.
* The system should run on at least 90% of Android devices.

## Definitions, Acronyms, and Abbreviations

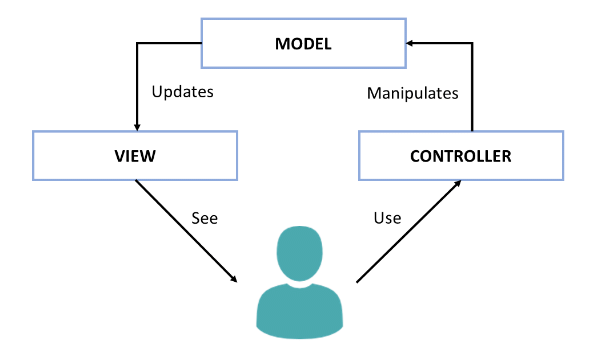
* ***SDD:*** System Design Document
* ***Confy:*** A conference management application which users can upload and review articles and get an approval grade relevant to the conference topic.
* ***Visitor:*** A user who is not registered into system.
* ***Java:*** Java is a programming language.
* ***Firebase Realtime Database:*** A cloud-hosted database provided by Google which allows to store and sync of data.
* ***Subsystem****:* Subsystem is a collection of classes of the system that are closely related to each other.
* ***Article:*** A scientifical writing document that is relevant to the topic of conference that is created.
* ***Conference:*** A meeting of several reviewers that came together to discuss and review an article that is relevant to the conference created.
* ***Reviewers:*** A role during the conference that has responsibility to review and assign grade to the article that is assigned to the conference.

## References

<https://www.exordo.com/>

# Current Software Architecture

There are numbered conference management tools and applications used in the world. These applications main purpose diverge in themselves. One can consider Discord, Zoom or Skype as a tool for conference. But main purpose of these applications in the academic world is to an event that is about the review process of academic articles about specified topics. Confy aims to be one of these applications under that subject. This current system is based on MVC architecture and subsystems inside of these defined three subsystems are called Model, View and Controller subsystems. The Model subsystem includes only application data as “User Data”, “Conference Storage”, “Paper Storage”. The View System presents the models data to the user. The Controller Systems acts as a bridge between Model and Controller Subsystems and takes events triggered by View and acts a reaction according to it.



# Proposed Software Architecture

The proposed software architecture in Confy is applying of MVC model to the Android platform along with Java. The Activities, Fragments and Views will be the Views. By this way, it is easier to extend and allowing easier new implementations. Because of the Model classes not having any references to Android classes, the Model classes will be straightforward for the unit test.

## Overview

MVC(Model-View-Controller) architecture are used for decomposing the system. There are 3 main subsystems in Confy: The Model Subsystems, The View Subsystems and The Controller Subsystems. The View Subsystems describes how the data presented visually to the users. The Model Subsystems include data of applications only. The View Subsystems executes its reactions according to the events coming from View subsystems’ activities. System Decomposition, Hardware Software Mapping, Persistent Data Management, Access Control Security, Global Software Control and Boundary Conditions of the Confy System takes care of the application in the next parts.

## System Decomposition

To decompose the system, MVC(Model-View-Controller) as architectural style. The decomposed system has 3 Subsystems: Model Subsystems, View Subsystem and Controller Subsystems.

**1) Model Subsystem**

Model Subsystem composes the entity objects of the system and responsible of keeping the data that are presented in conferences.

* **User Storage**

User storage takes the user information and stores it.

* **Conference Storage**

Conference Storage takes the information of conferences and review process and stores it.

* **Paper Storage**

Paper Storage stores the information about the grade of assigned article.

**2) View Subsystem**

View Subsystem is responsibility is to show events triggered, coming from Controller Subsystem by the user interface. Besides that, it is also responsible of presenting the models data to the user.

**3) Controller Subsystem**

Controller Subsystem includes User Management, Conference Management, Review Management and Paper Management components. These subsystems main responsibility is to manage data and events triggered by the user and be the bridge between Model and View Subsystems.

* **User Management**

User management is related with the Account Management. It processes the data related with the users by the Model Subsystem such as login and registration.

* **Conference Management**

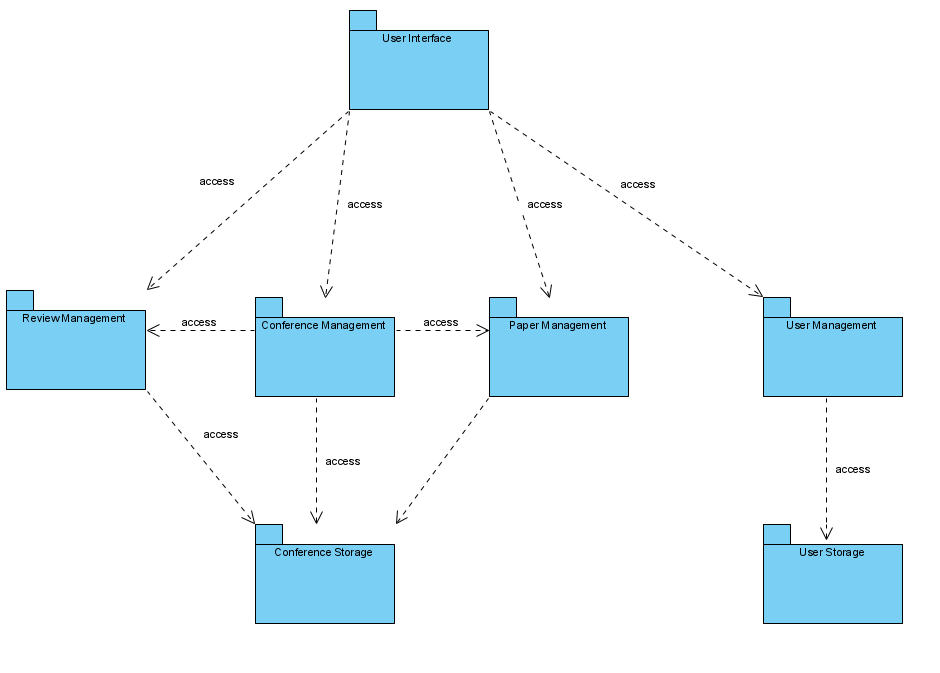
Conference management is related with the management and creation of a conference. It processes the data related with the conference by the Model Subsystem. It has functions such as “Create Conference” and “Edit Conference”.

* **Review Management**

Review management is mainly related with data that is associated with review process of the article that is sent to the conference. It has chat functionality and review services.

* **Paper Management**

Lastly, the paper management data is related with the article that is in the appropriate conference. It has the upload and edit functionalities of the post.



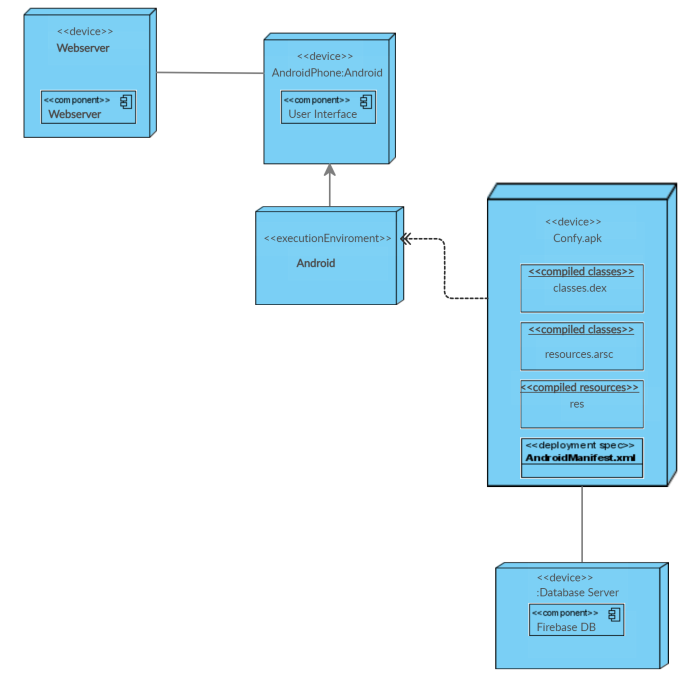
## Hardware Software Mapping

An Android Package Kit (APK for short) is the package file format used by the Android operating system for distribution and installation of mobile apps.

The AndroidManifest.xml file describes essential information about the app to the Android build tools, the Android operating system, and Google Play, including but not limited to, the app's package name, the hardware and software features the app requires, the components of the app and the minimum [API Level](https://developer.android.com/guide/topics/manifest/uses-sdk-element#ApiLevels) required by the app.

Confy.apk layer corresponds to the Controller layer, it is a deployable execution environment specified by Android, receives events from View layer, and updates the data at the Model layer.

Firebase Realtime Database corresponds to the Data layer, it handles cloud storage for the app.

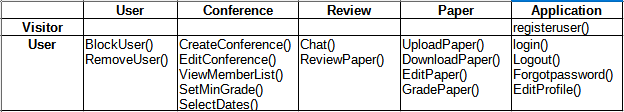


## Persistent Data Management

Confy is a conference management system application in which users can share their documents and papers. The system stores data such as user’s name, email, password, conferences, other blocked users, papers and reviews into a SQLite database by using an ORM library called [Room](https://developer.android.com/topic/libraries/architecture/room) as an abstraction over the database.

## Access Control and Security

In the system, there are two kinds of actor, Visitor and User (Registered User), the information of users, their conferences and their reviews are stored in the system. To gain security, crucial information such as users’ passwords are encrypted. Encryption is provided by Firebase Security& Rules. Firebase is hosted on Secure Sockets Layer (SSL in short) t's the standard technology for keeping an internet connection secure and safeguarding any sensitive data that is being sent between two systems. Firebase has two options for encryption, one of these is default one and it is not recommended but the second one is more secured and has tightened default security rules because the visibility of the data is not possible. With these two ways the database is secure in addition to the hashing of the data before storing it in the database. provides more secured way. When importing a user “UserImportHash”, SHA-1 password hashing algorithm or HMAC SHA256 password hashing algorithm can be used. By this way, accessing the database and the critical information of the user is getting harder.



## Global Software Control

User Interface subsystem positioned at the View Layer that initializes subsystems in the Controller Layer. User Interface initializes the login, registration, forgot password etc. The sent requests that are coming from users in User Interface Subsystem, at the View Layer, the Controller mechanism starts to manage requests to complete the events triggered by the users. Subsystems at the Controller Layer manages the processes and updates the data stored in the Model Layer Subsystem. User starts Login services at the View Layer by the User Interface, controller mechanism becomes a part of authentication the data coming from user. The View Subsystem provides required information to the Controller Subsystem and the Controller Subsystem works according to info that is presented to complete its services, and gives feedback to the user.

## Boundary Conditions

**Initialization**

To useConfyusers should download and install the app to their mobile device. An internet connection is required to download and utilize the application. Users must either register or login to the system after installing the app to make use and activate the app functionalities. They should be directed to the Main screen of the app. Accordingly they should be able to create and start their own conferences or review papers during their participation of other conferences.

**Termination**

For the system to shut down, the Controller Subsystem and the Management Subsystem are required. Controller subsystem manages the process and notification subsystem under the View Layer. Management Subsystem plays a role to update the data between the Model Subsystem and the View Subsystem. The termination of the system is completely in user’s control.

**Failure**

When a user fails to login, register, or has lost internet connection, the system should inform the user about the failure and the services should operate and require the least amount of resources to handle the user’s case.

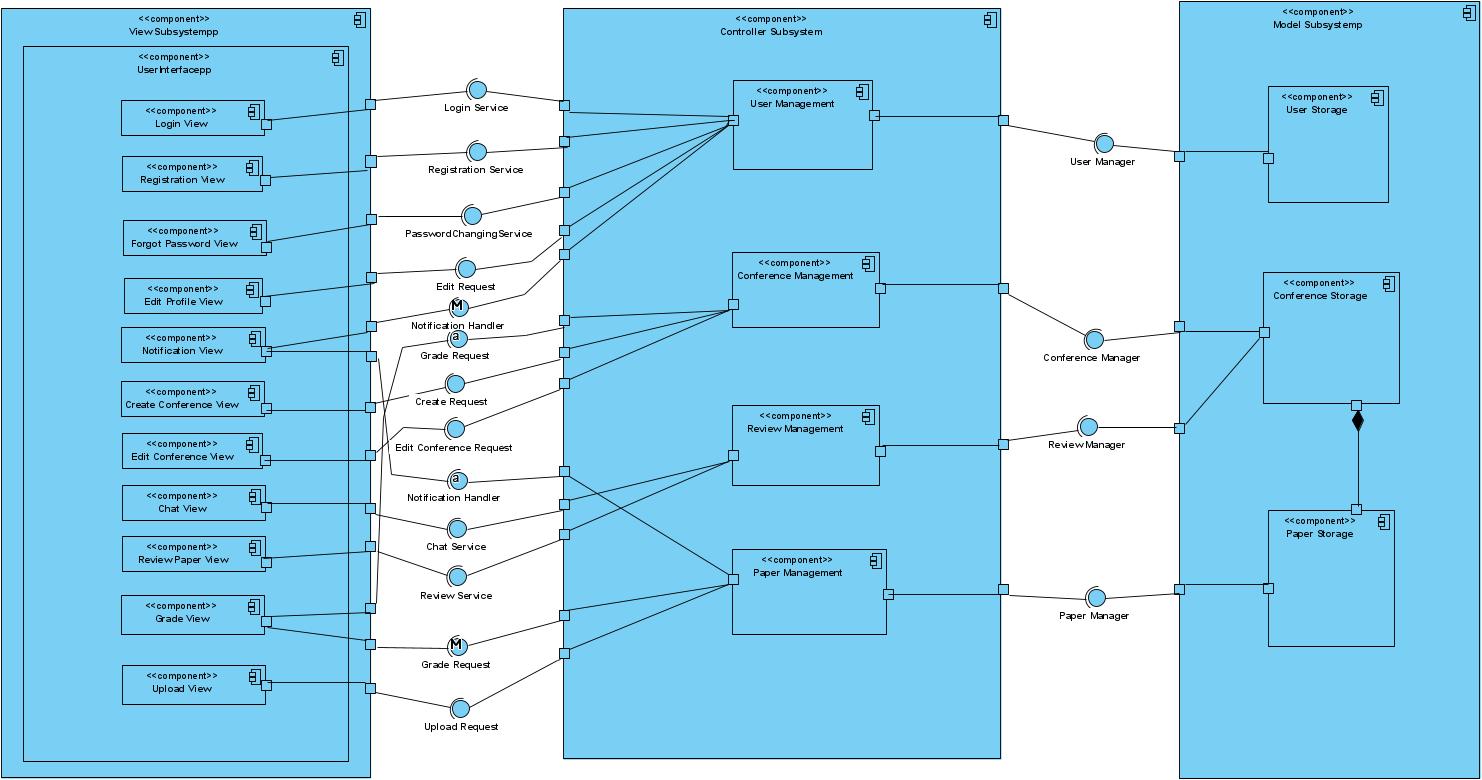
# Subsystem Services

View Subsystem’s responsibility is to show reactions coming from Controller Subsystem by the user trigger. Besides this, it is responsible of presenting the model’s data to the user. View Subsystem has Login View, Registration View, Forgot Password View, Edit Profile View, Notification View, Create Conference View, Edit Conference View, Chat View, Review Paper View, Grade View and Upload View components.

When a request comes to the Controller components, after the execution, controller object updates the data in the Model Subsystem components. Model subsystem stores information about users, posts and comments. When a user registered into the systems, their information such as name, password, e-mail etc. will be directed to Model Subsystem components and if is needed, that information can be accessed by the Controller components and can be updated.

The Controller Subsystem acts as a bridge down between the View Subsystem and the Model Subsystem. It processes at the server side of the system and provides main functionality of the system. In the system, there are four components in the Controller Subsystem: User Management, Conference Management, Review Management and Paper Management. User Management manages as an authentication tool for the Login, Registration , Forgot Password, Edit Profile and Notification View services. Conference Management creates and edits the processes of conference creation as well as takes the account of Grade assigned. Review Management manages the Review Part of the Conference process. Lastly Paper Management takes care of the grade process of the review process. In a way three of four management systems holds a keen link between themselves, being that all information and process being dependent of the conference created, review and conference should act according to it.

After the provided service “Login” by the Login View component user logs into the system. User Management component at the Controller layer manages the login, registration, forgot password, edit profile and notification view services so that when a request comes from the Login View, User Management checks the request’s data and process the data, authentication, and directs the user next related view. At the same way, registration service and forgot password service are managed by User Management. After User Management handles with the request sent by Login View, Registration View, Forgot Password View, Edit Profile View or Notification View components, it updates the corresponding model according to it. After the updating the model with the Controller, updated data is shown on the View Subsystem components. Conference Management handles creation and edit of conferences, events triggered by Create Conference View, Edit Conference View and Grade View. Along with the Review Management, Conference Management sends its data to Conference Storage and updates the data at the Paper Storage. Chat System responds to Chat View and sends the data to Review Management as well as Review Paper View. Grade Paper also sends data to Paper management as well as Conference Management and acts as a bridge between them. A user who wants to upload his/her file also uses the Upload View which sends its data to Paper Management. All data sent to Paper Management is stored at Paper Storage.



# References

The following is an example of listing a book in this section. Check the text to see how it is cross referenced (The whole document is based on [1]).

1. Bruegge B. & Dutoit A.H.. (2010). *Object-Oriented Software Engineering Using UML, Patterns, and Java*, Prentice Hall, 3rd ed.
2. <https://www.uml-diagrams.org/component-diagrams.html>
3. Huang, M., Y. Feng, B. C. Desai. CONFSYS: A Web based Academic Conference Management System. Proceedings of 2008 C3S2E conference. ACM Press, 2008
4. Timothy K. Shih, Jiung-Yao Huang, Jason C. Hung, and TeHua Wang, Wen-Chang Pai“ The design and implementation of a virtual conference system “, COMPSAC 2000.
5. Cheng Zheng, Weiming Shen, Qinghua Zheng, Feng Tian, ”Design and implementation of a collaborative conference management system”. CSCWD 2008.