

When The Ring Bells

System Design Document

V0.1

15.12.2019

Ersin ÇEBİ
Zafer KALYONCU
Serdar ŞAHİN
Çağatay DEMİRCAN

Prepared for
SE301 Software Engineering



IŞIK UNIVERSITY
COMPUTER
SCIENCE AND
ENGINEERING

Table of Contents

1. Introduction..... 1

 1.1. Purpose of the System.....1

 1.2. Design Goals.....1

 1.3. Definitions, Acronyms, and Abbreviations..... 2

 1.4. References.....2

2. Current Software Architecture..... 2

3. Proposed Software Architecture.....3

 3.1. Overview..... 3

 3.2. System Decomposition.....3

 3.3. Hardware Software Mapping.....3

 3.4. Persistent Data Management.....6

 3.5. Access Control and Security.....6

 3.6. Global Software Control..... 8

 3.7. Boundary Conditions..... 9

4. Subsystem Services.....10

5. References..... 11

1. Introduction

The purpose of this system is to design a attendance application for use in schools and in similar places. Designed system is user-friendly system. The biggest reason for making this system is that the existing polling applications used create many difficulties for students, teachers and other users. Our biggest aim is to minimize these problems in the website. We designed application to use fondly in schools and other similar place.

1.1. Purpose of the System

This system was designed with the aim of providing easy, credible for teachers who want to take attendance, users can display their classes, and participation percentages, which class their missed and detailed view of classes. It makes the whole process an easy affair without too much hassle as follow a few easy steps.

1.2. Design Goals

The design objectives signify the specified qualities of online attendance tracking system a constant set of standards that should be thought of when making design choices. Based on non-functionality requirements the next design targets must achieved as a way to qualify the system as profitable:

- **Security**

We use for save all kind of information in Firebase database. Firebase is a Google company. When we set non-security settings, Firebase warns us like "These settings are no safe". Firebase is a security platform for web application. Also, we set the rules of database in Firebase settings which kind of users can write or read data in Firebase. We can set these kind of rules in Firebase.

- **Reliability**

We keep the relate every data with users id in the tables in database. We will display the users summarized data in their main pages without any flow or mistake. Students and teachers will be able to see their classes and detail of those classes.

- **End User**

All users can reach their class data, if they login the application. However, Anonym Users can not login or sign-up the system, it is possible if and only if they are teacher or student. When they are using the application, their device has to be connected the

When The Ring Bells

internet. Our application can work on every device, they only need a web browser.

- **Performance**

Our system can sturdy enough to manage any valid input from the users. Our system can support 100 users to write and read data in firebase database at the same times. Our system has to show the elections results with 2-3 seconds.

Moreover, the other goals of our system should accept upgrades. The other thing is we have to design our app with easily understandable and useful. Also, the colors, pictures and icons in our app have to be connected each other.

1.3. Definitions, Acronyms, and Abbreviations

Teacher: One of the main actor of the system who wants to take attendance.

Student: The other main actor in the system who enrolls the lectures.

Admin: The system administrator who will manage all data system data and user controls.

Student Affair: Controls the relation between system and student, teacher.

RAD: Requirement Analysis Design

SDD: System Design Document

Firebase: Firebase is a Backend as a Service

Angular CLI: A platform for web application development

HTML: Hyper Text Markup Language

1.4. References

- 1) <https://doodle.com/free-online-voting>
- 2) <https://www.easypolls.net/>

2. Current Software Architecture

There is an existing system for online attendance tacking. The application show all class details and info gives ability to open attendance and take it. Registered user can see their class details. Students can enroll classes and teacher can display their classes student via system. Student Affair can create new class and can approve the students and teacher who wants to join in the class. User's has specific profile screen. Students can view their information even view the participation ratio results per class. The most important point is teachers can take attendance easily and safety with online attendance tracking system.

3. Proposed Software Architecture

'Whentheringbells' is a web application. This project will be very useful for schools, universities or education communities. Our system is easy to use and reliable, has a user-friendly interface.

In short, in our system, the system allows users to participate in open courses, register to courses, view his attendance, view his profiles.

In addition, the system allows users to update their profiles, change their passwords, and delete user accounts by the administrator.

3.1. Overview

During the system design modeling of 'Whentheringbells', we divided our system into subsystems. Our subsystems are; Admin Management System, Student Management System, Teacher Management System, Student Affairs Management System

3.2. System Decomposition

This subsystem is managing admin accounts. It offers perform creating an election, deleting an election, approving an election. Admin is the only actor who has permission to access add election and approves functions. This subsystem uses login services of the Registered User management subsystem.

The operations provided by this subsystem are:

- Login()
- Add Election()
- Delete User()
- Delete Election()
- Logout()

Student Management Subsystem

This subsystem is managing student users' functions, offers registered side to its functions. This subsystem manages student users' student-course relationship.. After, logging step.

The operations provided by this subsystem are:

- Login()
- forgotPassword()
- AddCourse()
- Profile()
- ListAllCourses ()
- ListMycourses()
- DropCourses()
- CourseInformation()
- Logout()

Teacher Management Subsystem

This subsystem is managing mainly attendance functions, offers registered side to its functions. This subsystem manages taking, opening attendance of a class. After, logging step;

The operations provided by this subsystem are:

- Login()
- Open Attendance()
- Edit Attendance()
- Take Attendance()
- View Student List()
- Logout()

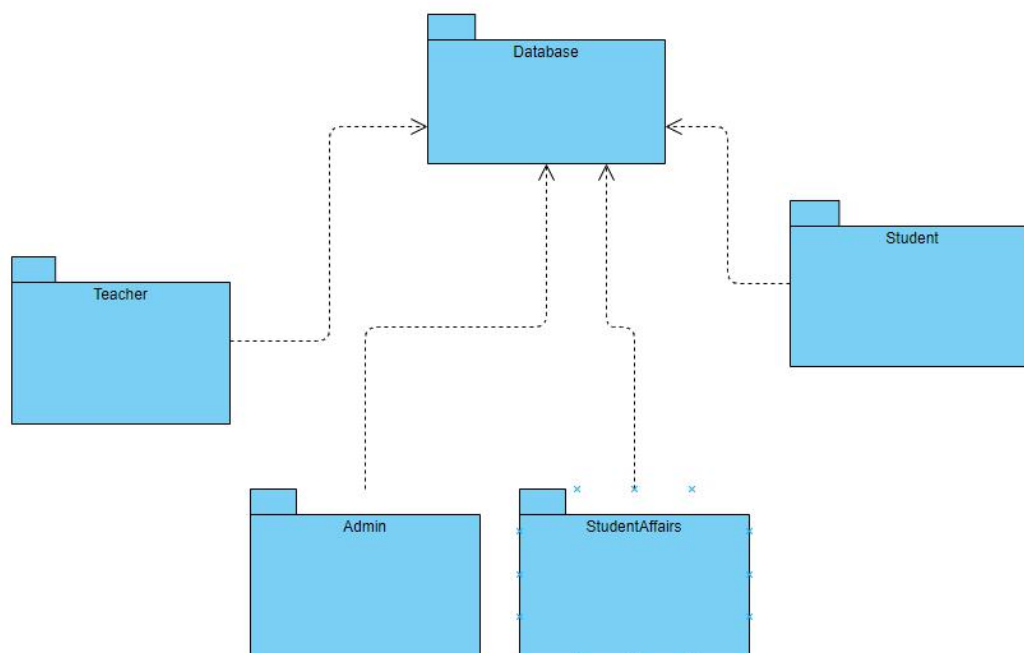
Student affairs Management Subsystem

This subsystem is managing Student affairs accounts. Student affairs, opens courses, edit student information, responds to wishes, deletes students or teachers from the course. It is the most authoritative person after admin, the only restriction is that he does not have access to database.

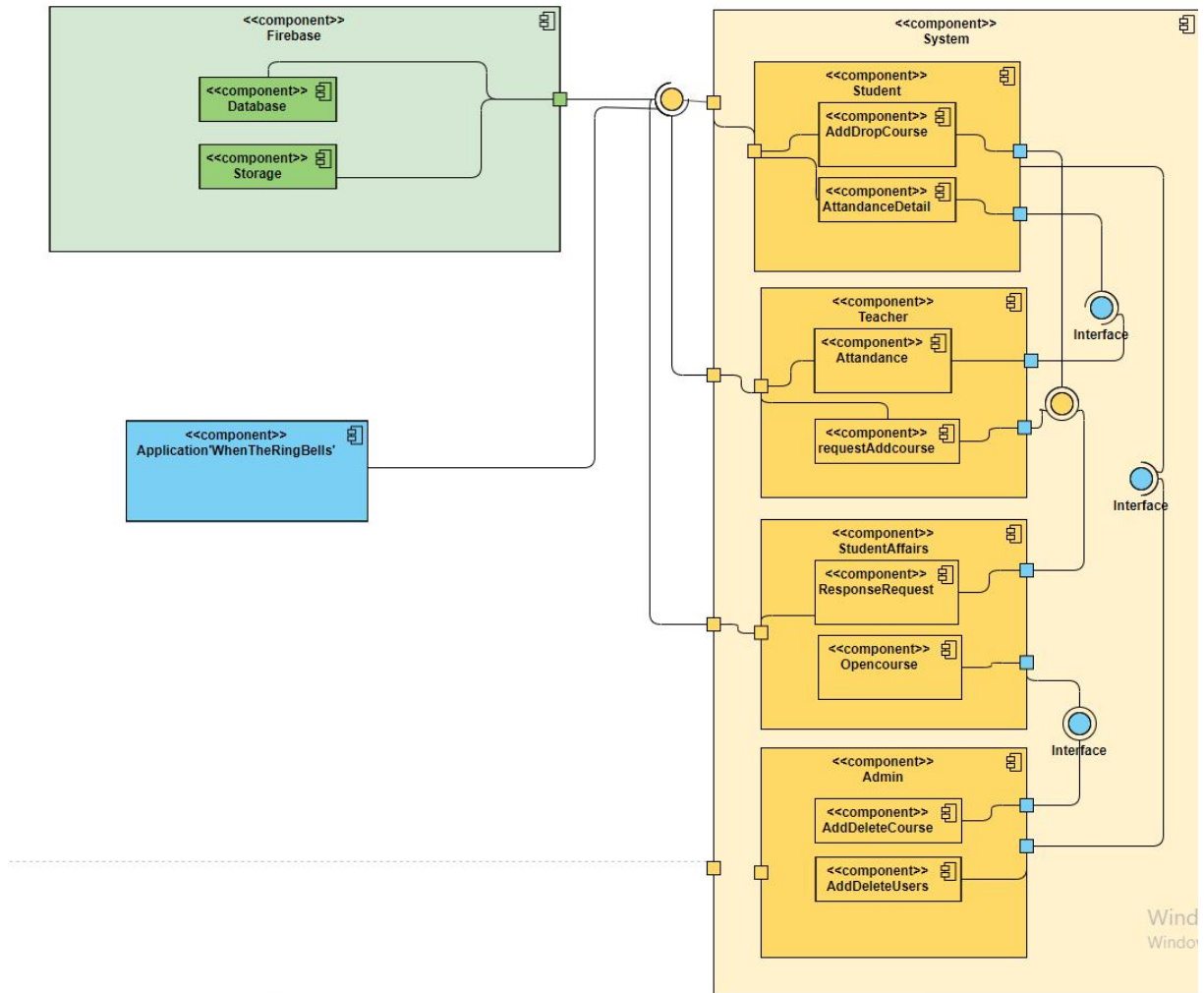
- Login()
- Adding students and teachers to course()
- Edit student information()
- Opening a course()
- Request Section()
- deletion student and teachers to course()
- Logout()

Database subsystem

This subsystem will be implemented by relational database management system which is Firebase. All subsystems are related and having service with this subsystem



3.3. Hardware Software Mapping



3.4. Persistent Data Management

Our system will use the firebase system to store data. This system will make it easier and faster to read and write data. The database will store values such as user input, user information, and attendances. Our database structure appears as follows.

3.5. Access Control and Security

The website is multi user website so it occur of 4 types of users which are registered user, these are; admin, student affairs, teacher, student. Because, web site going to offer different interfaces for each user type.

First, admin login like normal users. As he / she is logged in, he / she sees his / her special page on this page, can confirm the changes, delete students from the course and add teachers to the course. In addition, admin can directly access the database and make changes. When student affairs login to the system, they see their own pages. add students and

When The Ring Bells

teachers to the lesson, update the student's information, open the course, approve the requests from the student and teacher, and finally delete the student or teacher from the course. When the teacher enters the system, he sees the special page on this page; open polling, receive polling, send requests to student affairs. When the student enters the system, he / she sees his / her special page; can send the request to register for the course, can see the percentage of the course stay, view the profile, can be withdrawn from the course.

Only student affairs and admin will be able to register. The data will not be added to the database if they leave a blank field during registration. The system saves database changes made on the website and all recorded information is displayed in real time by other users. They will be required to complete all fields. Furthermore, if a teacher or student affairs deletes a course while the student is looking at course attendance, the website will send the student to the home page because the course has been deleted.

	System	Attendance	Course	request
Admin	login() logout() forgetPassword() resetPassword() deleteUser() addUser()		addCourses() deleteCourses() editCourse()	checkrequest()
Student affairs	login() logout() forgetPassword() resetPassword() deleteUser() addUser()		addCourses() deleteCourses() editCourse()	checkrequest()
Instructor	login() logout() forgetPassword() resetPassword()	takeAttendance()		sendrequest() checkrequest()
Student	login() logout() forgetPassword() resetPassword()	participateAttendance() trackAttendance()		sendrequest()

3.6. Global Software Control

External Control Flow

When The Ring Bells defined by the website with a simple feature. Firebase Database System requests submission of user data. System is multi-user, concurrently executions can occur. But, single user's control flow has been defined before. After user login to the system, the system has side-bar to navigate through web pages that can user see section. Some sections can only reachable by some user's role.

Internal Control

Callbacks Between Subsystems: All of the services uses asynchronous callbacks for inter-process communication. Every single service uses internal methods to notify the other subsystems of its own status, its needs and abilities.

Avoiding Deadlocks: Callbacks must not block the caller. So only status information is updated in the callee. All other internal work within a single service must be done by other worker threads.

Worker Threads For Each Service: Each service has an own thread for communication, which communicates with the service manager and other services.

Concurrent Control

Callbacks Between Subsystems: When The Ring Bells uses asynchronous callbacks for interprocess communication.

Multithreading: When The Ring Bells uses threads, so that a large number of services is able to use the service manager simultaneously. The service manager also handles asynchronous events within the system.

User Interface

User Interface has many web pages and side-bar that can user navigate through web pages. The control of next step depends on user's action and choice. In addition to this, the flow is implemented in pages. In When The Ring Bells, we have user roles. This roles can or can not be access some pages. Admin, Student, Student's Affair, Teacher have their own web pages that can do their actions defined their roles.

3.7. **Boundary Conditions**

System Start-Up

Go to website and login.

System Shutdown

Click log-out.

Error Conditions

1-Login

E-mail and password filled can not be blank.

E-mail must be in e-mail format.

E-mail must be valid.

E-mail and password can not be same.

Password must be longer than 5 characters.

2- Creating Attendance

Attendance has to have list of students and date.

Attendance date can not be date before day that adding attendance.

Attendance can not be created by Student.

Attendance has to have class and list of students that taking this class.

3- Taking Attendance

Every student's attendance must be filled.

4- Taking Class (As a Teacher)

Class teacher position has to be empty.

When The Ring Bells

4. Subsystem Services

Login Page



Giriş

Email address



Password



Giriş

© 2017-2019

Create Attendance

Name:

Name



Date of...:

DD.MM.YYYY

Explanation:

Explanation

Submit

Viewing Attendances

#	Attendance Name	Date	Attendance Ratio	
1	Attendance 1	12.12.2019	100%	Edit Delete
1	Attendance 1	12.12.2019	100%	Edit Delete
1	Attendance 1	12.12.2019	100%	Edit Delete
1	Attendance 1	12.12.2019	100%	Edit Delete
1	Attendance 1	12.12.2019	100%	Edit Delete
1	Attendance 1	12.12.2019	100%	Edit Delete
1	Attendance 1	12.12.2019	100%	Edit Delete
1	Attendance 1	12.12.2019	100%	Edit Delete
1	Attendance 1	12.12.2019	100%	Edit Delete

Taking Attendance

#	Student Name	Status
1	Ersin ÇEBİ	<input checked="" type="radio"/> Present <input type="radio"/> Absent
2	Zafer Şükrü KALYONCU	<input type="radio"/> Present <input checked="" type="radio"/> Absent
3	Çağatay DEMİRCAN	<input checked="" type="radio"/> Present <input type="radio"/> Absent
4	Serdar ŞAHİN	<input type="radio"/> Present <input checked="" type="radio"/> Absent

5. References

- 1) <https://doodle.com/free-online-voting>
- 2) <https://www.easypolls.net/>
- 3) Bruegge B. & Dutoit A.H.. (2010). *Object-Oriented Software Engineering Using UML, Patterns, and Java*, Prentice Hall, 3rd ed.
- 4) Lecture presentations of the course (the presentations were provided by the Instructor who is Emine Ekin and Deniz Yigitbasi).