TALLINN UNIVERSITY OF TECHNOLOGY
School of Information Technologies
Thomas Johann Seebeck Department of Electronics

Safwan Talukder Fayad, 194522IVEM

Development of smart entry/access solution for academic office

Master's Thesis

Supervisor: Ants Koel
PhD

TALLINNA TEHNIKAÜLIKOOL
Infotehnoloogia teaduskond
Thomas Johann Seebecki elektroonikainstituut

Safwan Talukder Fayad, 194522IVEM

Nutika sisenemis/juurdepääsulahenduse väljatöötamine akadeemiliseks kontoriks

Magistritöö

Juhendaja: Ants Koel
PhD

Abstract

In the contemporary era, management of work and time has become a major issue as we go more and more into multitasking. Coping with a wide window of different tasks and responsibilities is never that easy and it requires help from an external source. In the case of office-seated working individuals dealing with the disturbance from visitors is a common challenge during their daily life. This is where this thesis comes into play with its research. It focuses on developing a system that would basically make visitor entry and management easy for university staff seated in offices. Displaying updated schedule and current status of staff and employing a means to communicate with the target person inside office room beforehand, the proposed system makes the management of visitors very smooth in such a work environment. There will be no interruption of work or meetings taking place inside office and no time wasting on the staff's end as well as that of the visitor. This system is basically composed of an affordable tablet in the corridor, a central server where a web application stands and a user's smart device to access the software. The number of users can go from one to basically any number. This system, called Smart Entry System, abbreviated in SES, is a simple, ease-of-implementation solution based on smart devices, a web application and Internet connectivity. Integration with the staff's MS Outlook Calendar is performed and then data is taken into a central web application standing on a central server. On this server, a web application is placed and hosted to two groups of people; visitors and users themselves. The visitors would be able to basically check for availability of users and the users would be able to update their information automatically through MS Outlook Calendar. The coding languages that have been used are PHP (Laravel framework), HTLM, CSS (bootstrap framework), javascript (jquery library), structure query lan.

Annotatsioon

Kaasaegsel ajastul on töö ja aja haldamine muutunud oluliseks küsimuseks, kui me läheme üha enam mitme ülesande juurde. Erinevate ülesannete ja kohustuste laia aknaga toimetulek pole kunagi nii lihtne ja see nõuab abi väljastpoolt. Kontoris töötavate isikute puhul, kes tegelevad külastajate häirimisega, on nende igapäevases elus tavaline väljakutse. Siin tuleb see tees oma uurimistööga mängu. See keskendub süsteemi väljatöötamisele, mis põhimõtteliselt muudaks kontorites istuvate ülikooli töötajate külastajate sisenemise ja haldamise lihtsaks. Kavandatav süsteem muudab külastajate haldamise sellises töökeskkonnas väga sujuvaks, kuvades ajakohastatud ajakava ja personali hetkeseisundi ning kasutades selleks vahendeid kontoriruumis sihtrühmaga suhtlemiseks. Kontoris toimuvat tööd ega koosolekuid ei katkestata ning aega ei raisata nii töötajate kui ka külastajate jaoks. See süsteem koosneb põhimõtteliselt koridoris asuvast taskukohasest tahvelarvutist, keskserverist, kus veebirakendus seisab, ja kasutaja nutiseadmest tarkvarale juurdepääsemiseks. Kasutajate arv võib minna ühelt põhimõtteliselt ükskõik millisele numbrile. See süsteem, mida nimetatakse SES-is lühendatuks nutikaks sisenemissüsteemiks, on lihtne, hõlpsasti rakendatav lahendus, mis põhineb nutiseadmetel, veebirakendusel ja Interneti-ühenduvusel. Integreeritakse personali MS Outlooki kalendriga ja seejärel võetakse andmed keskserveris seisvasse kesksesse veebirakendusse. Selles serveris on veebirakendus paigutatud ja majutatud kahele inimrühmale; külastajad ja kasutajad ise. Külastajad saavad põhimõtteliselt kontrollida kasutajate kättesaadavust ja kasutajad saavad oma teavet automaatselt värskendada MS Outlooki kalendri kaudu. Kasutatud kodeerimiskeeled on PHP (Laraveli raamistik), HTLM, CSS (alglaadimisraamistik), javascript (jquery raamatukogu), struktuuripäringu lan.

Author's declaration of originality

I hereby certify that I am the sole author of this thesis. All the used materials, references to the literature and the work of others have been referred to. This thesis has not been presented for examination anywhere else.

Author: Safwan Talukder Fayad

10.05.2021

List of abbreviations and terms

SES	Smart entry system
FE	Front end
BE	Back end
HW	Hardware
SW	Software
WA	Web application
RT	Real time
DB	Data base
CS	Central server
MS	Microsoft

Table of contents

List of figures

1. Introduction

In physical working environments, it is almost never easy to maintain a constant work-favoring conditions. There is always some sort of crisis or interruption, be it power failure, distraction from our co-workers, knockings on the door, or incoming call-ringing on our phone. To a moderately large extent, it is possible to deal with such issues with moral sense and being organized. Yet, there is quite a big vacuum that we just cannot fill without an external. This is where technology comes into play. Undoubtedly, it has embraced us with the means to deal with a lot of our daily life issues and crisis including a lot of these with its great innovations. Today we can, for example, put our phones to vibration mode which allows us to keep being aware of any incoming calls without catching the attention of our surrounding people. We can foresee and identify the visitors at our doors and gates to facilitate security in our homes and offices. However, the very problem of work interruption from occasional knocks and entries by visitors at office rooms and other working spots has not been quite touched yet. Hence, this project has been conducted to develop a system that can solve this crisis to a moderate extent.

SES, which stands for smart entry system, as the name suggests, would target the area of such kinds of hampering of corporate work in school, college and university buildings particularly at office rooms. Our system is, basically, composed of three platforms: a tablet/raspberry pie installed at the door (the front end), a personal computer or mobile phone or any smart device for that matter under the control of the office staff (the back end), and a central server (the medium) to hold all the data. The entire system is based on an online app standing on the central sever accessible to both the visitor on the front end and the user themselves at the back end.

The below shows the basic set up of the SES proposed by this thesis:

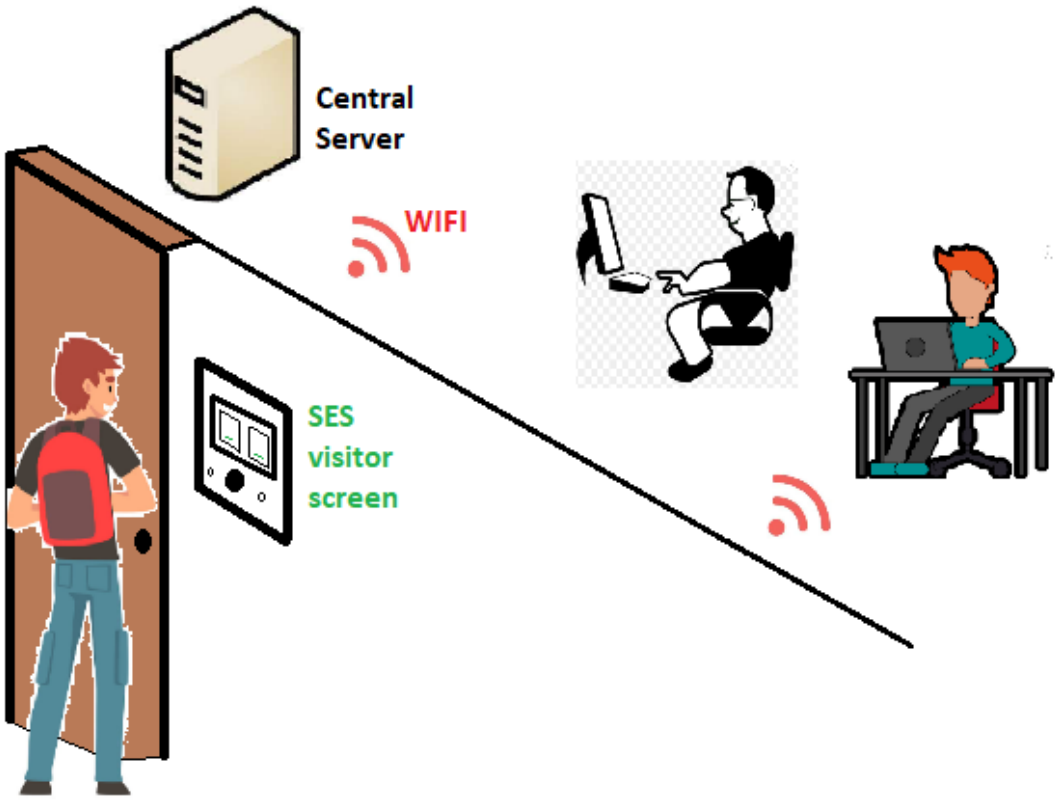


Figure 1 Visitor checking staff's status through SES screen before making entrance

As shown in the figure, the goal of this project will be met by SES to successfully perform in the following way:

1. A guest/visitor sees one or more user windows on the screen (one for each user) displaying their schedule, status and has a means to communicate.

2. If no on-going event for their chosen user, they might want to communicate with them asking for their current and hence, permission to enter the office.

3. The user gets notified on their PC and responds to the visitor according to their situation.

This system has quite a wide range of applications such as in commercial buildings, banks, government facilities, hotels, hospitals, domestic rooms of freelanced IT job holders, all of which require some sort of privacy control from time to time. The system is very affordable and easy-to-install and is thus, quite suitable in almost all kinds of premised for any anti-privacy-invasion purposes not only academic offices.

However, the SES is not without its challenges. ---Imperfections of the system---Here, the main challenge would be the choice of code. developing the app which plays the biggest role in the system. A lot of coding knowledge would be required for this. Also, a feasibility tests would be required at every stage of the app development as there a number of different functions of the app, thus, to avoid mismanagement and disagreements between different functions.

1.1 Motivation to build a smart system

A smart system is one which basically makes use of one or more hardware devices and at least one software application to implement a certain task or tasks that could otherwise be done by one or more humans. It makes our daily life simple by eliminating the needs for employing human resources for regular jobs around us and at reasonable costs.

This area of study basically expands the knowledge of various possible ways of implementing the latest technology in efficient ways and making it our everyday friend in one way or another. The real value of science and technology comes out only when it can actually make a difference to the lives of a lot of people. Such requirements in the daily lives of a lot of people around me are the main motivation behind the interest to conduct a study in this field.

Based on my educational background of Electronic Engineering and my coding knowledge, I find myself well-suited in this field; which drove me towards making the attempt to build a smart system.

1.2 Applications of the system

1.2.1 Resolve work interruptions at office rooms on University/school campus, hospitals, hotels, other organizations

An office-seated staff needs a stable environment to stay focused on work. There needs to be a way to handle possible disturbance sources around one of which could be unexpected appearance of guests making an entry into the room and starting a conversation during important work every now and then. Staff would prefer to have some

control over such incidences to avoid making errors due to distraction. This is where our SES could come well into play.

1.2.2 Conference/meeting rooms

The system can bring an end to the unacceptable interruptions of important ongoing meetings from sudden openings of door. Thus, if somebody in charge of the meeting room would put up the event to be displayed on screen outside it could stop people to mistakenly open the meeting room door (which happens sometimes) in search of the room availability.

1.2.3 Residential private room of freelance worker

The IT sector is expanding like never before and it is constantly broadening the horizons of freelance IT workers working from home. Thus, this system could come handy for such an employee to deal with regular visits from fellow family members or even outsiders.

1.3 Problem/research Statement

Environment has a great impact on every occurrence taking place within it. No matter how effective a process is, it always gets influenced by its surrounding events at least to some extent. Corporate environments are meant to be places where work individuals can concentrate and focus on their job with the necessary privacy and peace. Thus, work can be in progress. Whether it a school, university, hospital any kind of institution for that matter, a professional environment needs to be maintained.

In a workplace, a large proportion of distraction comes from visitors because a lot of the times they come up with a topic not related to the ongoing work. They may show up with

questions or requests regarding any matter. This can ruin the focus on work and once focus is ruined it can be hard to re-establish it especially if the individual is working on something complicated.

It is necessary to have a bit of organisation in this aspect. A doorkeeper would not be practically a solution in the case of an office, as there are usually numerous office rooms in institutions like a university or college. Thus, in this thesis the goal is to come up with a smart door-keeping system that could actually bring a solution to this daily mess. A system that enables the office owner to voluntarily accept or reject visitation requests both on spot and remotely with an understanding settled between the visitors and them. This is done simply by conveying messages which can be pre-set or live. In this way, disturbance can be kept at a minimum level without completely ignoring and missing the message from the visitors as tentative meetings with them is settled via the system.

With smart entrance systems like this one organisation can have its employees maintain an updated communication with co-workers and work without irritations.

Many useful smart devices and systems are being employed around us today. However, there is a limited amount of attention given to this particular area of managing appointments to meet with visitors. Thus, there is a need for a proposal of something effective here to cause a spark in this important area and hence, this thesis aims to reach the goal in the following steps:

- Investigate the state of art in this field and gather as much information as possible about the latest technical methods being employed or proposed in the field
- Brainstorm and come up with a few completely new and fresh ideas.
- Accumulate them with the selected existing techniques acknowledged to build an effective, affordable, easy to install and user-friendly set of hardware and software kind of smart system.

1.4 Approach followed in this thesis

The following steps have been taken to accomplish the goal of this thesis:

- Form a complete user story after a through discussion with supervisor.
- Based on the user story, form several sets of use cases and the possible paths of realizations needed for them.
- Select the most appropriate set of use cases and the corresponding path of realization in terms of feasibility and time available for completion.
- Decide on the coding language to conduct
- Divide the system into its functional parts: back end, database and front end.
- Start the work on designing the web application
- After a considerable and stable progress of the web application start implementing with one browser (as the back end) to access the app being put on a central server to serve as the staff and another browser to serve as the visitor
- Make the required changes to the application and continue testing till all the use cases are satisfied and the system is error-free

2 State-of-the-Art on smart entrance

This chapter represents the research work conducted on the current position of technology in the field of IOT particularly in entrance area. There are quite many systems of hardware devices and apps filling this sector of automation technology based on different techniques and serving in various ways.

Some of the moderately relevant systems in this area being proposed or employed around currently are described below in terms of their architecture, functionality and relevance to our system.

2.1 A smart meeting room scheduling and management system with utilization control and ad-hoc support based on real-time occupancy detection

From [1] we can find a system proposed by Linh Duc Tran of RMIT University, Alex Stojcevski, Thanh Pham and Tony de Souza-Daw that allows users to acknowledge the real-time availability of a meeting room and then book the Room accordingly.

This system can detect occupancy status of a meeting room at the current time and then integrate this information into the scheduling application to support ad-hoc meetings. This way it is possible to increase room utilization. The system makes use of PIR sensors and Ethernet connectivity. Occupancy data is then sent to a central application server where there is a web application developed. This web application allows people to book rooms for meetings, and also to check the utilization of the room based on some predefined policies. The figure below depicts the basic architecture of the system:

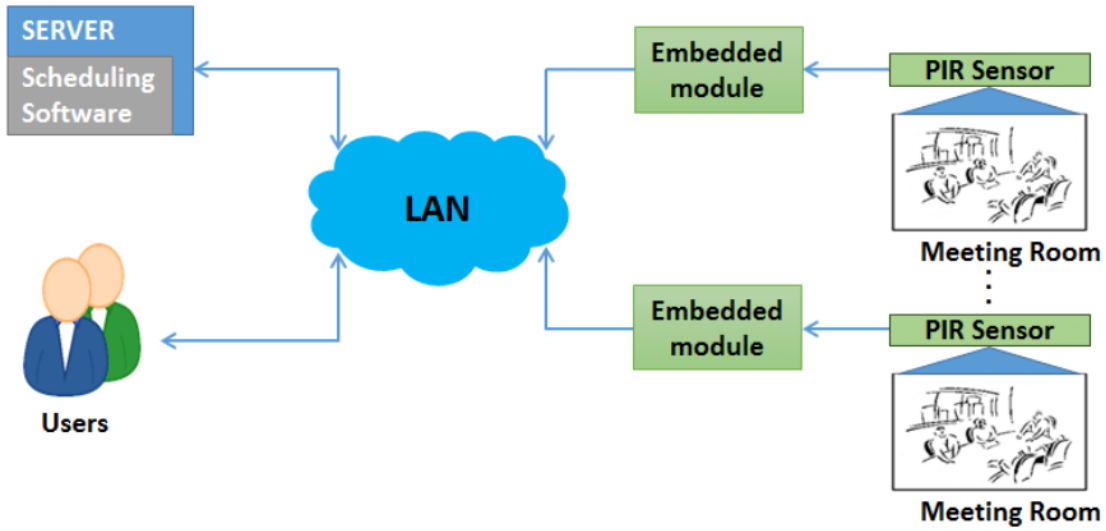


Figure 2 Proposed system overview

Relevance and differences:

- The system is a continuously updated with real-time room availability information and our SES also keeps up with updated data (but not real-time) of a person's schedule.
- It makes use of a central server where data is sent to a web application standing on that server like our system.
- It performs integration of occupancy information into the scheduling application as ours does for events in Outlook calendar.
- It automatically collects data by the use of motion sensors as ours automatically collects the user's manual input of data into Outlook calendar.

- The aim of this system is maximum utilization of meeting room while ours is to facilitate smooth and acceptable entry.

2.2 Security and Usability Improvement on a Digital Door Lock System based on Internet of Things

From [2] we can find a system proposed by Ilkyu Ha, Kyungil University, Gyeongsan, South Korea. It is a digital door lock system that serves in an IoT environment. The system targets enhancement of security and convenience of in a room. The system is capable of sending recorded images to the owner of the room whenever there is an unauthorized visitor making an attempt to perform an illegal operation. Moreover, it is capable of sending out an alarm to the owner's mobile device when the door lock is physically damaged. It gives the user/owner the flexibility to monitor and control their door lock remotely.

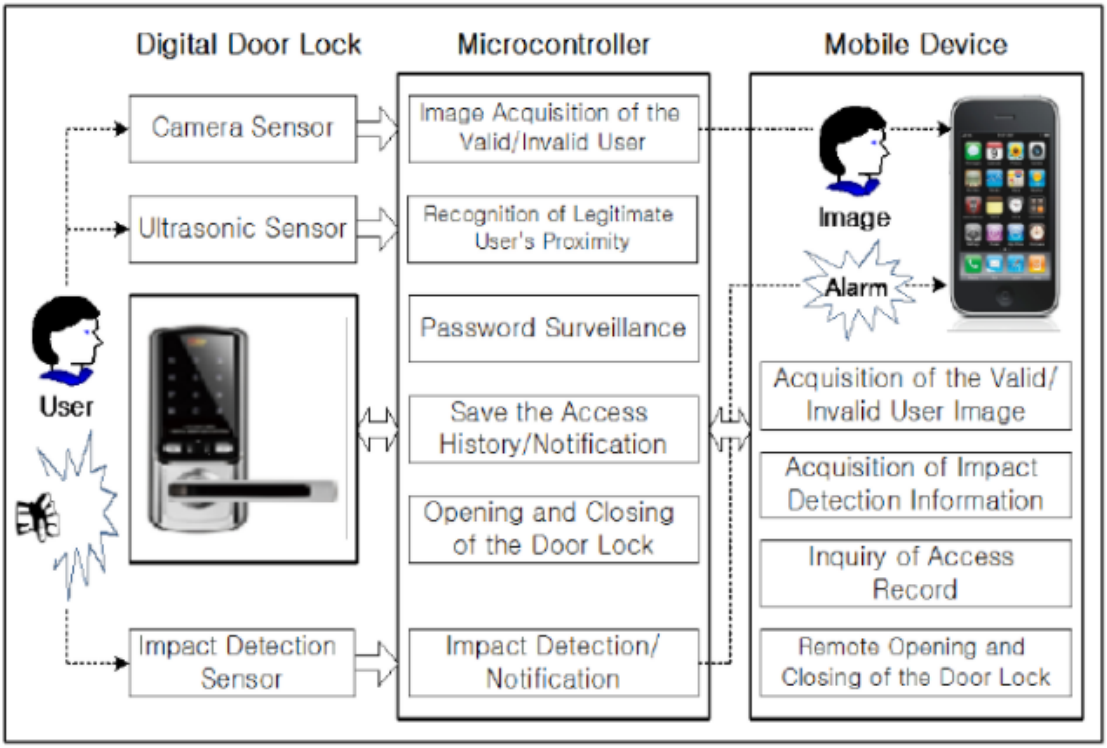


Figure 3 Structure of the Proposed Door Lock System

Relevance and differences:

- The system is a real-time door lock status detector as the SES displays the current status of the individual inside the office.
- It performs acquisition of the data from its sensors to result in outputs for the user but our system integrates Outlook data into its own event/schedule window.
- It automatically collects data by the use of motion sensors as ours automatically collects the user's manual input of data into Outlook calendar.
- The aim of this system is for maximum security of a room while ours is to avoid interruptions from unexpected entries of guests into office.

2.3 Smart Entrance System Using Computer Vision at Corporate Environment

Another great system we can has been proposed by the authors S.M Amir Khan; Md. Nazmul Islam Shuzan; Moajjem Hossain Chowdhury; M. Maksud Alam in [3]. It is a smart entrance system which uses computer vision. This system basically makes use of Jones Algorithm for face detection. In the case of a person facing down or up or elsewhere, i.e., not straight the system corrects the alignment by applying Kazemi algorithm. It also uses filtering process in order to account for occasional changes in appearance of male employees as they might have mustache and beard on their face on different days. Thus, through filtering it can get the image of male individuals to detect how they may look if they have mustache or beard. Consequently there is reduced risk of a faulty detection regardless of different facial appearance. These pictures are then added

to database. The system then performs face recognition by the use of Eignefaces which is easier and quicker in computing an employee under smart entrance system.

Relevance and differences:

There is not much relevance between this system and ours other than the ultimate goal which is, to facilitate a smart entrance through a door. The system completely focuses on security of corporate premises while SES is meant to provide an organized environment in office rooms.

2.4 Envoy and Openpath access control

A great system has been developed by OpenPath in [4]. Here, Envoy users can have OpenPath to perform integration with their Envoy which will allow them to automatically assign Openpath credentials to their visitors that are set up in Envoy so that they can keep record of entry activity of users in the Openpath activity log. Moreover, it even allows them to verify the health conditions of their visitors and also to enforce symptom waivers at check-in via the Envoy Protect platform and Openpath readers.

Relevance and differences:

- The system performs integration of data available in one platform (Envoy) to develop into another set of data (visitors with assigned Openpath credentials which somewhat resembles our integration method of Outlook data into a second set data on the visitor screen.
- The goal of the system is to provide secured entrance of individuals unlike ours.

2.5 Smart office access for your employees

In [5]we find a system called Nello one which comes with an existing door buzzer. It can be connected to an intercom via WiFi. It is beneficial in the way that a person can access their office remotely using the nello app. In addition, here we have digital keys that can be shared among employees. The nello one is easy to install (DIY-installation is used) and is affordable which makes it very useful for small companies. After installation, the administrator can add additional users to the app and assign them with individual user rights.

With nello one it is possible enable a smart office access. Every employee will come to the office with their smart phone. The admin determines at which times the door buzzer can be activated via the app. The activity feed is very handy. It tracks who opened the door at what time.

Relevance and differences:

Here, the use of a central app to control the entrance of individuals into the office is what seems very relevant to our system although the app here is used to directly control the entrance of people while in our system the central app is used to create entry permissions or rejections to individuals and hence, to control entry passively.

2.6 Smart Entrance Monitoring System Using Clothing Color Analysis in Multi-Camera Environment

An interesting system can be acknowledged in [6]. A proposal of a smart entrance system that looks out for intruders using clothing color analysis in multi-camera environment by the authors Yoo-Joo Choi, Seoul Media Institute of Technology, D.-C. Kang, W.-D. Cho, K.-J. Kim. It has the ability to detect intruders based on the colours of their clothes. It

basically serves as a support to an already existing visitor authentication system. Whenever there is a person going through authentication at the entrance, by the use of perhaps identity card or finger print, it extracts their clothing features from CCTV camera images in a multi-camera environment which are then added to a database. Hence, when a random person appears at the entrance and is authenticated by the authentication system, their clothing feature vector is automatically extracted by this system and then compared with the reference feature vectors in the database. If a matching set of clothing feature vectors is not found in the database, then the individual would be identified as an intruder.

Relevance and differences:

This system takes into account the clothing colours of visitors which could be, in future, put into our system as a feature for eliminating unwelcomed visitors at office doors. These unwelcomed visitors could be non-students or non-employees in any kind of school or institution where specific outfit is employed.

2.7 Haltian - Thingsee PRESENCE

An amazing device can be found in [7]. It is a wireless sensor for measuring people's presence through our facility. We can use the sensor for facility management and occupancy detection in various places such as offices, hospitals or restaurants. It can also monitor usage levels of meeting rooms and office desks. Occupancy data can be further used to optimise space usage as well as energy consumption.

Relevance and differences:

It has very little to do with the goal of our system. However, it does have the ability to monitor the levels of occupancy in meeting/office rooms through its sensor and thus, we

might actually consider this feature for our system at some point if not at this time around. If this could be somehow projected on the doorbell screen then our visitors would not have to aimlessly tap the ‘Call’ button on the screen to get staff’s response in the case they are not in office at that time.

2.8 Extron Room Scheduling

In [8] we acknowledge an effective room scheduling system being employed around us today called Extron room scheduling. It can actually simplify the experience of booking a meeting room very easy. It consists of very interactive panels called TouchLink Room Scheduling Panels. We can connect these panels to many popular calendaring devices like Microsoft® Exchange, Office 365™, Google Calendar™, 25Live®, EMS, NFS, and Ad Astra. These panels are configured with a software called Extron Room Agent which allows them to be connected to our PC so we can set up a meeting in a certain room on our own computer. They also provide the information that can be used to analyse room usage, activity patterns, and occupancy trends of meeting rooms.

These panels can even be equipped with a digital input that will enable them to work with Extron OCS 100 series occupancy sensors to monitor room occupancy. These sensors use ultrasonic and passive infrared detection technologies and are highly sensitive.

The system also has an interface called the TLSI 201 Interactive Wayfinding Interface which directly connects with its panels to provide real-time meeting space availability, status, and location information on a centralized display and users can locate and book meeting rooms on this central screen and that too on two possible platforms as the options; an interactive list or map.

Hence, the system is extremely flexible giving its users the options to make reservations for meeting with just a couple of taps on the room scheduling panel hanging outside

office/meeting room, or centrally from the Wayfinding Interface, or from their own mobile phones or tablet or personal computer.

Relevance and differences:

- The system, first of all, employs the room availability information in a calendar form on a front-end screen just like our system.
- As one its features, the system allows its users to put in information from their computer or mobile device which would show up on the front-end screens (on corridor screen and central screen) through a software. This is basically the way our system performs.
- However, this system focuses on successfully making room reservations for meeting which is not the concern of our system.

Summary of the state of the art

Summing up all the ideas and knowledge gathered from the various methods, techniques and outcomes proposed or utilized for an automated entrance till date, it can be said that the specific area or window of an MS-Outlook-Calendar based doorbell for entrance is still not put hands on yet. There is still room for making a new and fresh proposal of such

a system. This conclusion on the state of the art is, of course, limited only to a certain level of certainty since the research conducted could not be said to be done at the fullest.

Hence, based on the acquired information the state-of-the-art systems can be classified into two main groups in terms of their ultimate purposes:

- Visitor management for room booking
- Safety and security

Our work here is to provide an insight of a certain individual's availability to their guests for a consult and also a means to communicate with them from outside the door which is fairly a new concept.

3 Specification of system

The entire system has been partitioned into three main use-case specifications:

1. Calendar use case which describes the path of steps and events to view user's calendar from the corridor in its entirety.

2. The 'Call' button use case which demonstrates the steps through which the staff and visitor can communicate with each other.

3. User platform use case describes the steps for the user to handle their SES account.

3.1 Calendar use case

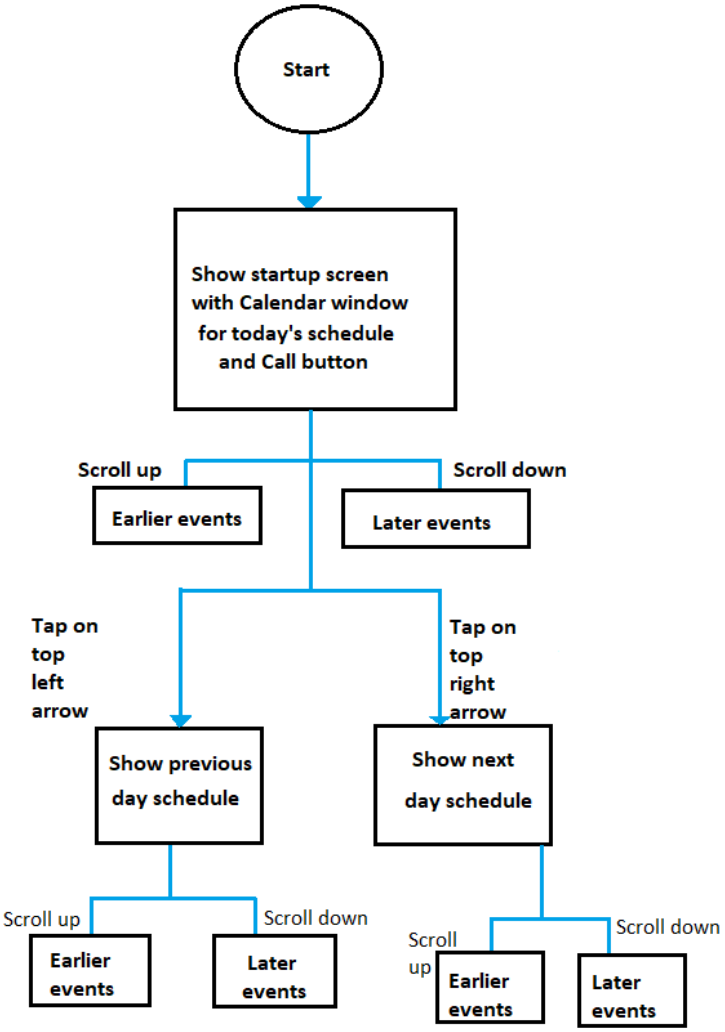


Figure 4 Calendar use case

Pre-condition

- Tablet must be switched on
- Must be connected to WIFI
- Server link page on the browser must be kept open all the time

Basic path

- This use case starts as the person in the corridor has started reading the calendar
- They scroll up or down
- Earlier or later events show up respectively
- They tap left or right button on the top right of calendar
- Events of previous or next day appears respectively

3.2 Call button use case

3.2.1 Possibility 1 (Basic response)

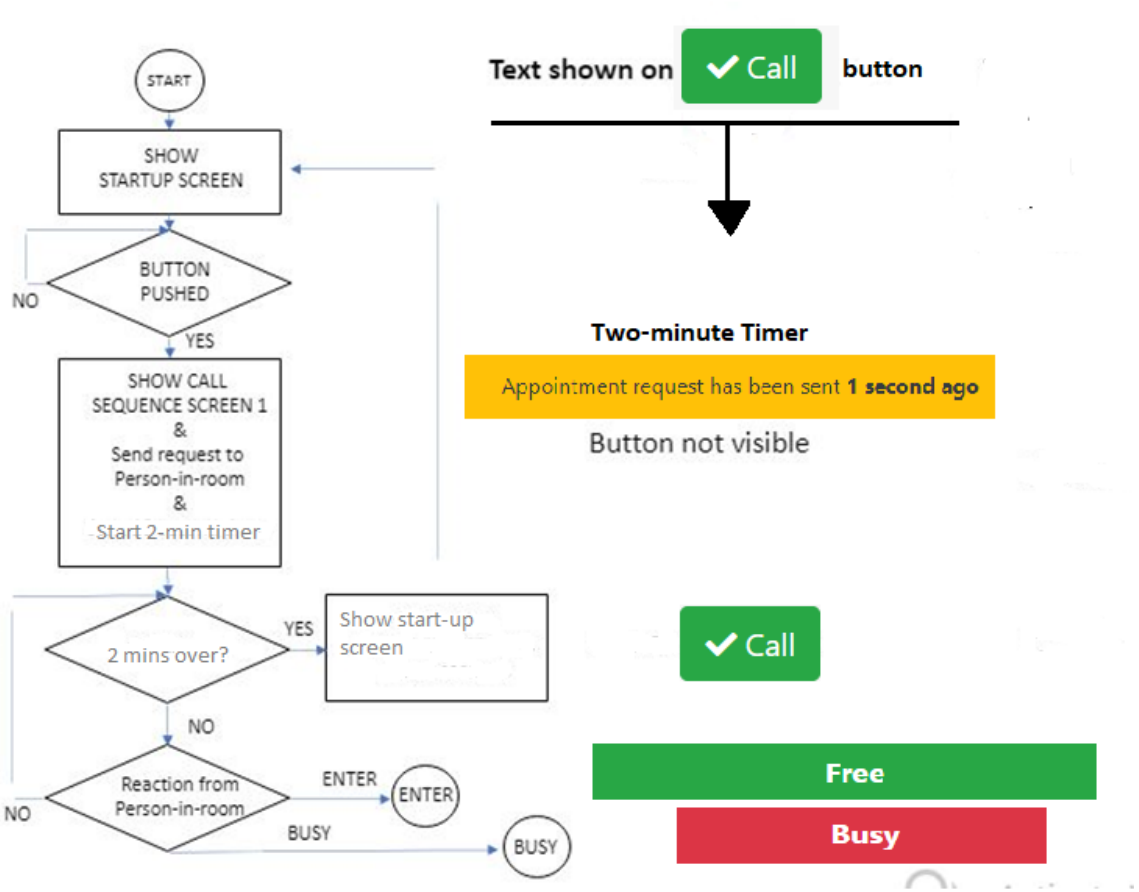


Figure 5 Call button possibility 1 use case

3.2.2 Possibility 2 (Response with text message)

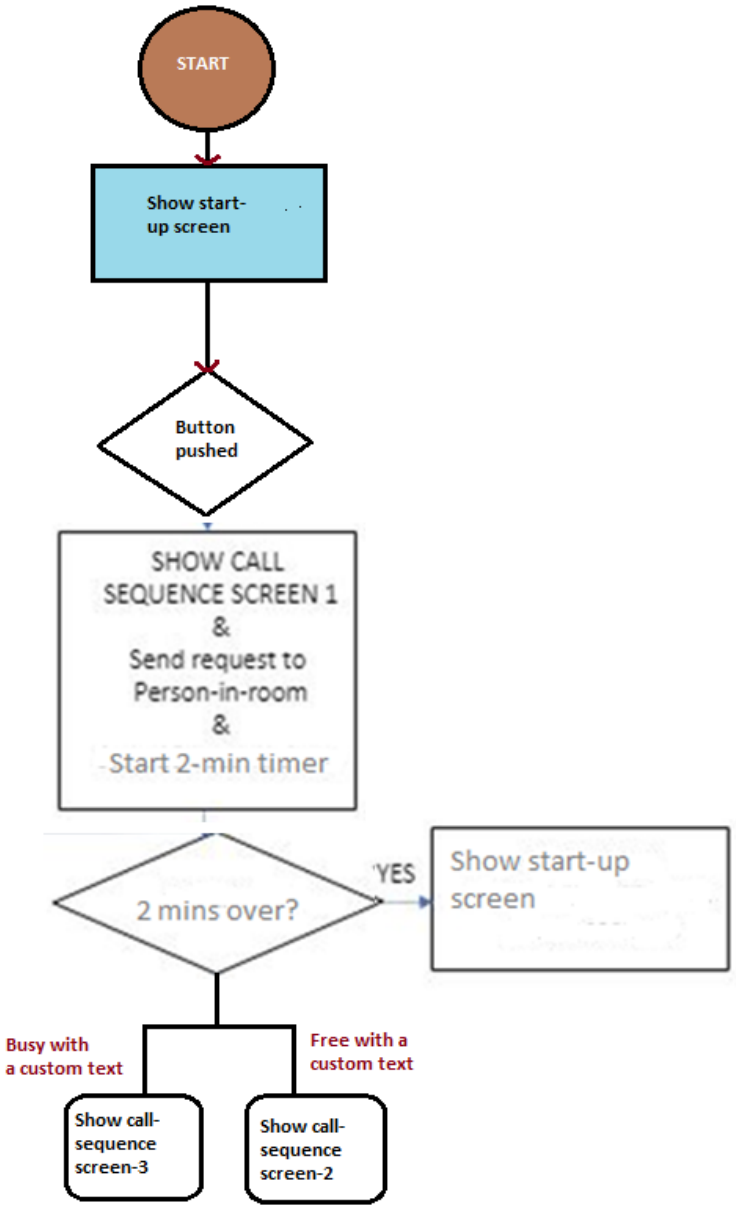


Figure 6 Call button possibility 2 use case

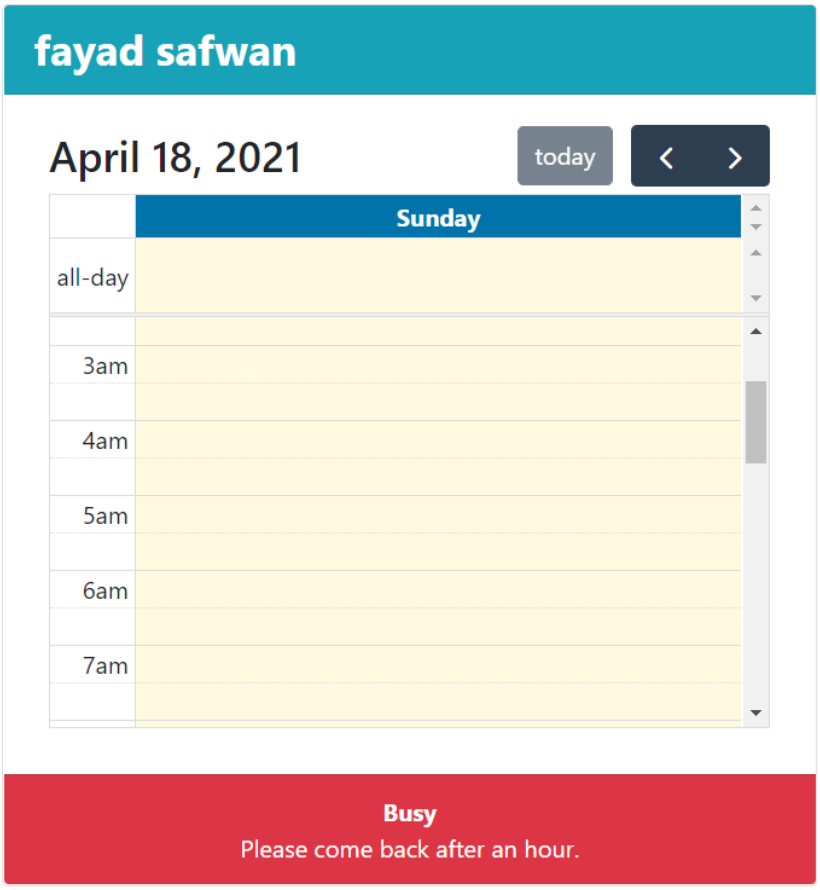


Figure 7 Call sequence screen-3

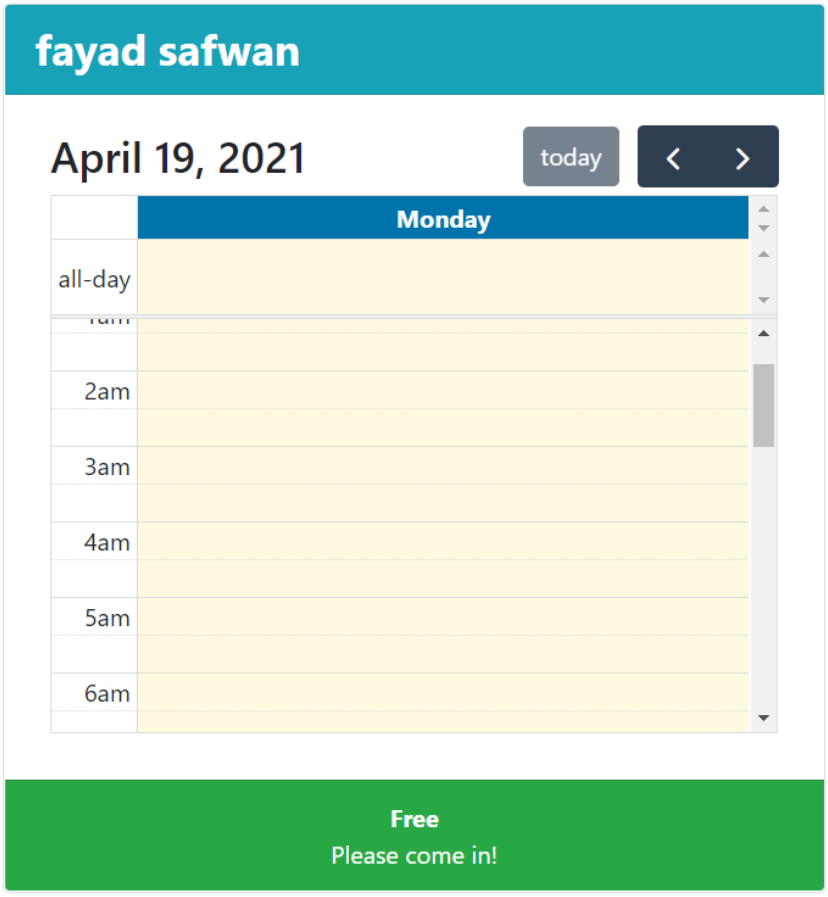


Figure 8 Call sequence screen-2

Pre-condition

- Tablet must be switched on
- Must be connected to WIFI
- Server link page on the browser must be kept open all the time

Basic path

- This use case starts while the person in the corridor has tapped the ‘Call’ button
- Appointment request sent to the user
- Two-minute timer begins
- User responds within the two-minute time period with Enter or Busy options on their end
- Response from user shows on the corridor screen accordingly

Alternative path

- At step three of the basic path user does not respond within two-time period
- Hence, at four of the basic path start-up screen shows on the tablet

3.3 User platform use case

3.3.1 Sign up/in use case

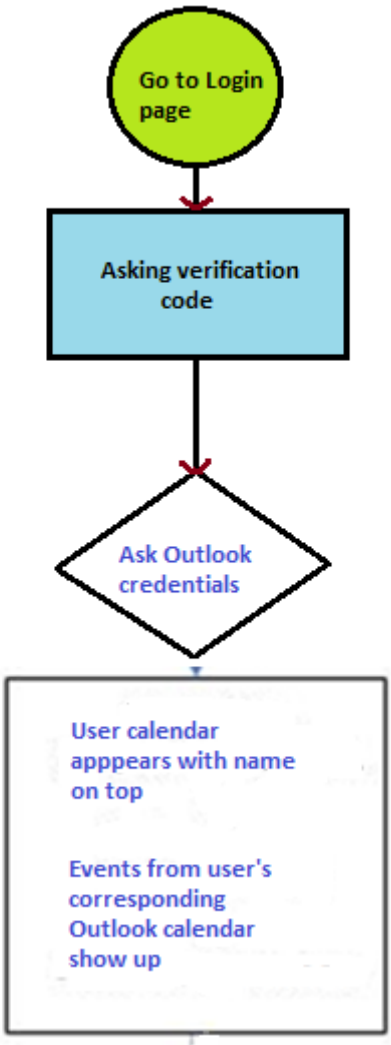


Figure 9 User registration use case

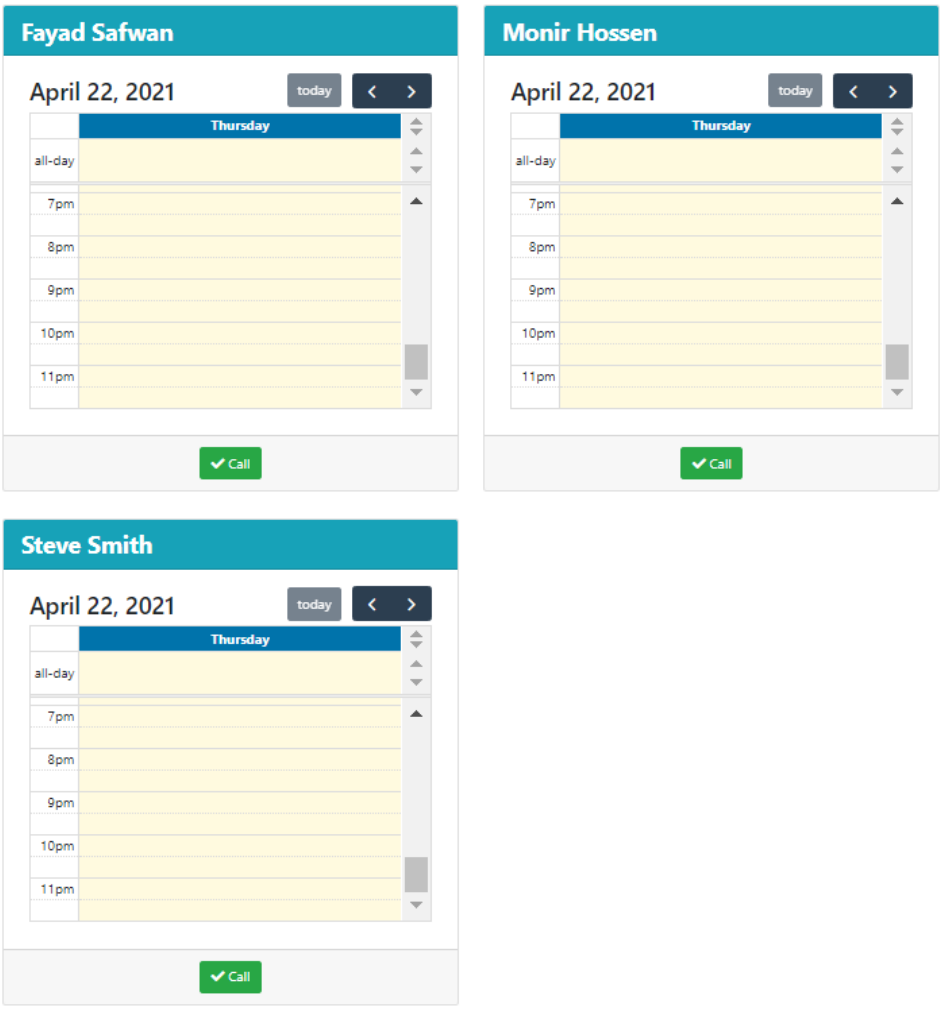


Figure 10 Multiple user windows appear row-wise

3.3.2 Add event use case

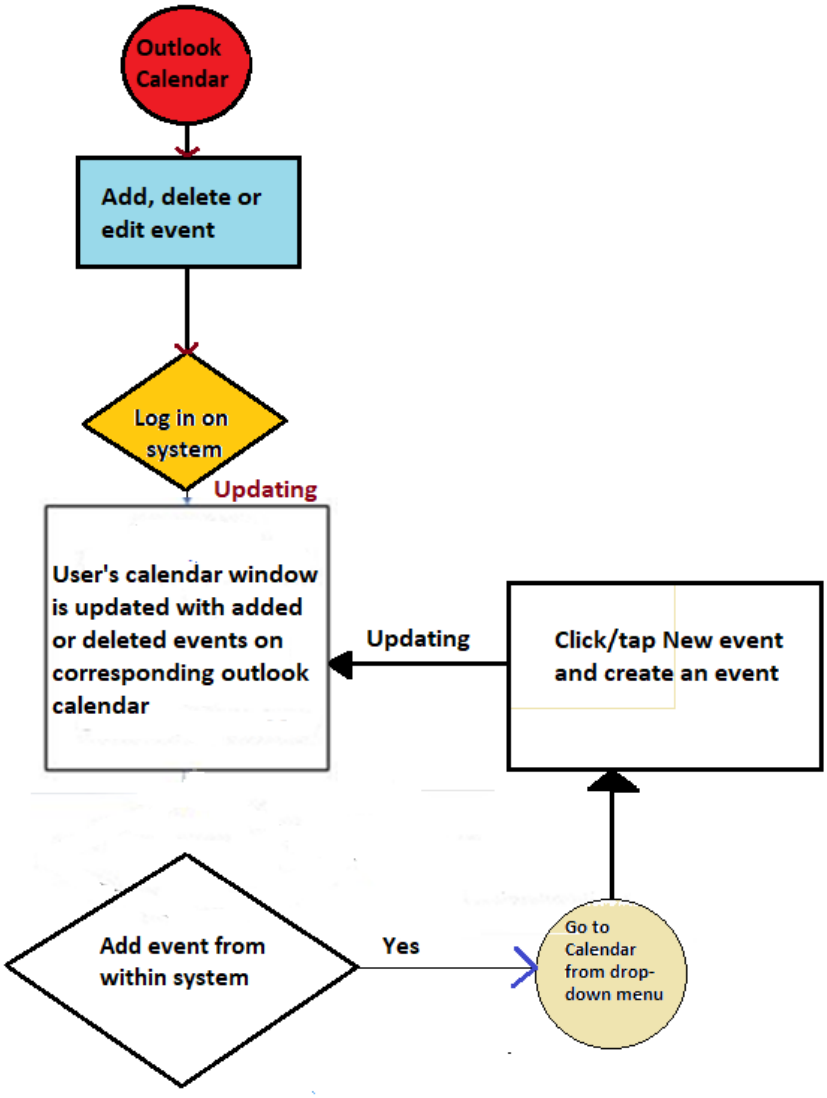


Figure 11 Add/delete event use case

3.3.3 Response to Call use case

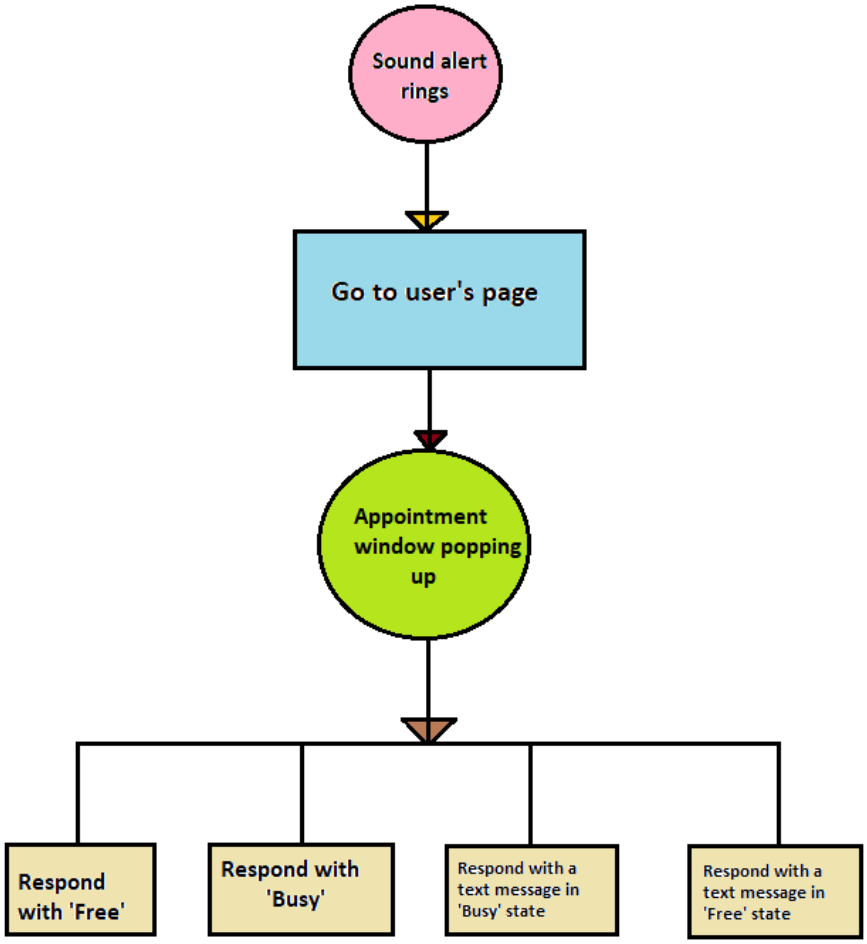


Figure 12 Response to Call use case

Restrictions of data flow

Data is only drawn from the MS Outlook Calendar, not from any other source. If any event is missing in the particular Outlook account of the user associated with our system, information will be missing on the system's calendar unless the user manually puts it in the New Event platform within the system.

The flow of data from Microsoft Outlook Calendar into the system takes place via the free Outlook API available on the internet. The code for establishing API connection, database connection and server connection is all given below in the Appendix section (see Appendix-2).

4 System design

4.1 Introduction, challenges of developing the system, functional blocks, possible architectures, choice of architecture

4.1.1 Introduction

In this system, a visitor at the door initially sees a user schedule list, a “Call” button with the person’s name on top, in each user section on the doorbell screen.

They scroll up/down or move to next or previous page or even make a call to the user according to their needs and get the information. On the other side, the user can also acknowledge their appearance and deal with accordingly without being disturbed. The system is quite simple and is basically composed of three units: the doorbell, the software application and the back-end computer for the user. This system could be applied to any situations for anti-privacy-invasion purposes not only to staff office room entrances.

4.1.2 Challenges of developing the system

As for the development part, certainly it will not come without challenges. Here, the main challenge would be developing the application which plays the central role in the system. Quite a lot of coding knowledge would be required for this. Also, feasibility tests would be required at every stage of the app development as there a number of different functions of the app, thus, to avoid mismanagement and disagreements between different functions and to ensure the progress in the right direction.

4.1.3 Functional blocks

The main functional blocks of the system are:

1. User's Outlook Calendar
2. User's account on the system or BE
3. Database
4. Calendar on the front screen or FE
5. Central server

4.1.4 Possible architectures

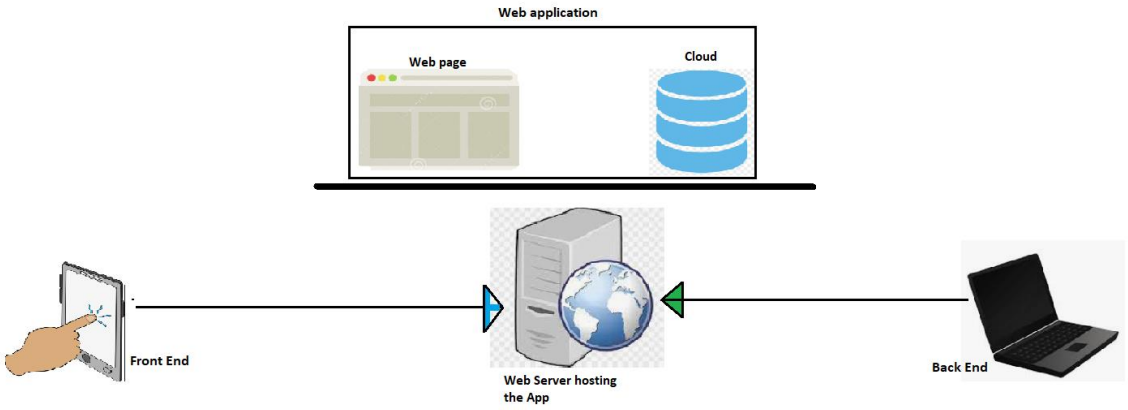


Figure 13 Server-based architecture

In the above diagram, it is visible that the software application of the SES is a Web application placed on a central server which the hardware components of the system; the tablet or Raspberry pi in the corridor and the laptop of the user access via the Internet.

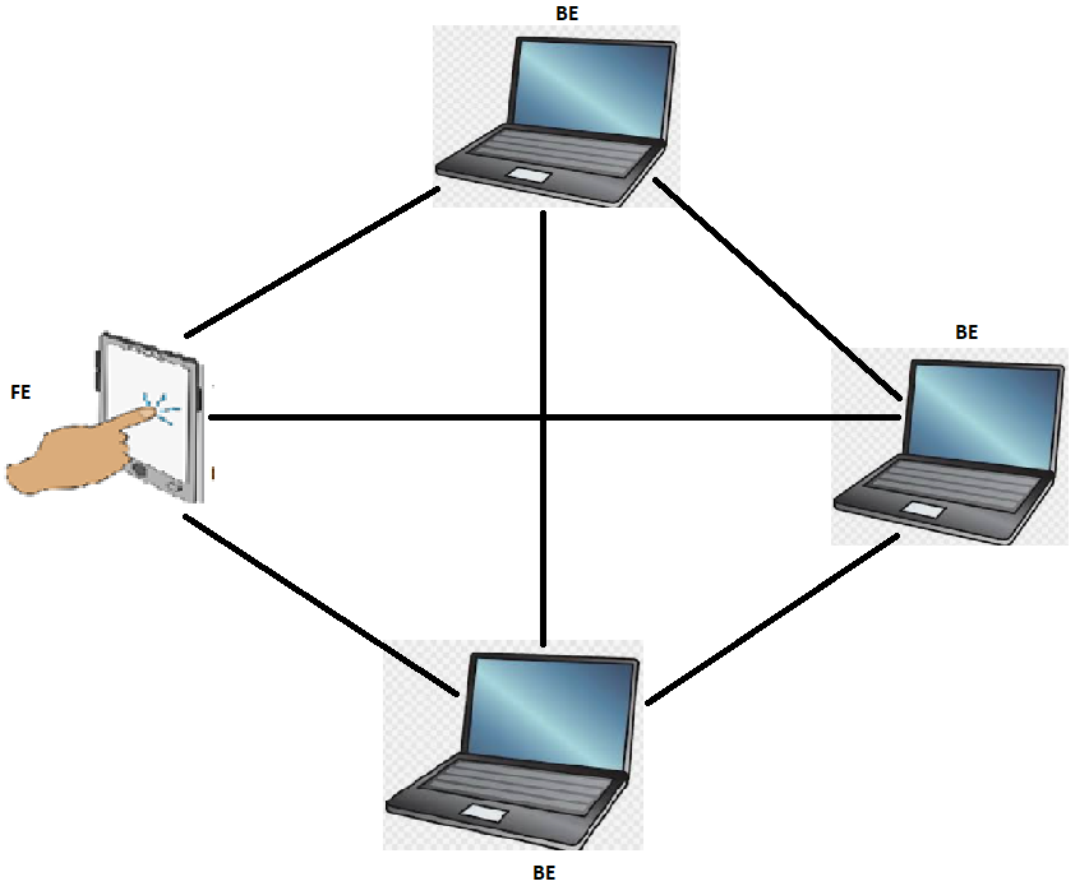


Figure 14 Peer-to-peer architecture

Here, each of the individual devices would have the system application running locally.

4.2 HW needed for functional blocks

4.2.1 HW choice options for user interface in the corridor

Below are the preferred choice options for HW for the doorbell taking into account the availability, affordability, the required configuration and features, and installation cost:

I. Fire tablet 7
II. Raspberry PI Zero Wireless (RASPBERRY-PI-Z-W
III. IUniker 2.8" touch screen
IV. Memory card
V. USB power supply
VI. Suitable housing
VII. Laptop/desktop personal computer or smart phone

4.2.2 HW chosen for system implementations

When choosing the hardware, the decision is made here mainly in terms of cost and ease of installation and the required configuration. The following HW items were chosen as a start:

- Nokia 2.4

Figure 15 Nokia 2.4

- USB power supply
- Mount
- Laptop computer on staff end

4.3 SW choices for functional blocks, design of the application

4.3.1 Options for software platforms

Here, we have the flexibility to employ any of the following software platforms for the operation since the SES is a Web-application based system which mainly requires a browser and an operating system. The operating system could be any of the following:

- Android.
- CentOS.
- iOS.
- Linux.
- Mac OS.
- MS Windows.
- Ubuntu.
- Unix

As for the web browser we have a wide range of options also, such as, Microsoft Internet Explorer, Mozilla Firefox, Google Chrome, Apple's Safari, Opera, etc.

4.3.2 Design of the visitor page or FE

Some of the possible designs for the visitor page that were held as options for our system are given below:

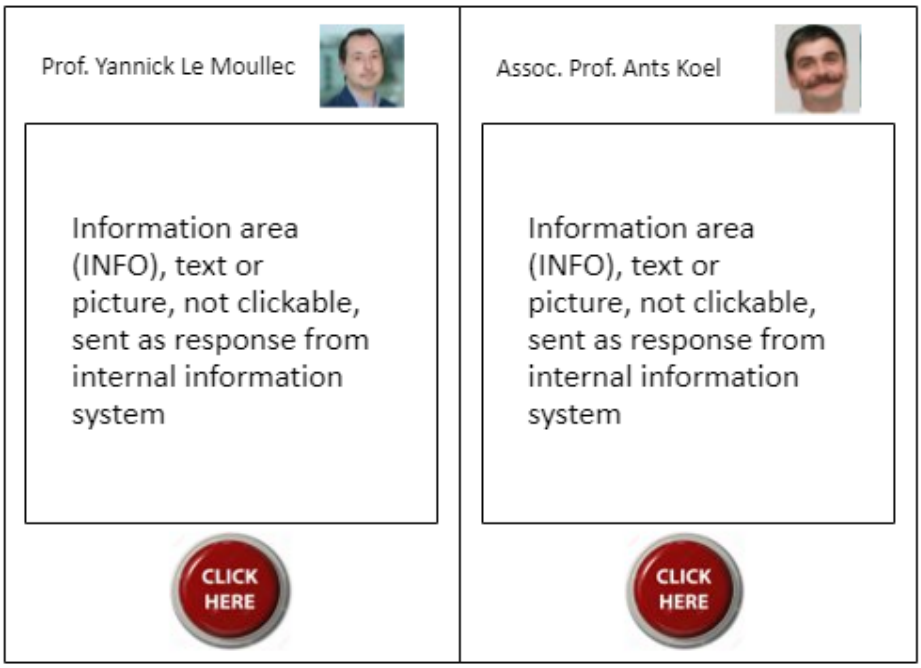


Figure 16 Column-wise-user-slots view

INFO = Information area, text or picture, not clickable, sent as response from internal information system

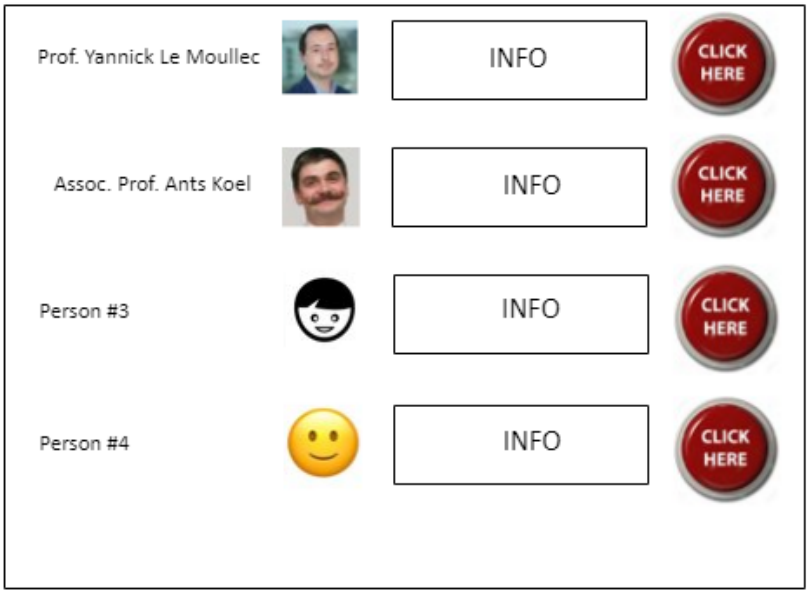


Figure 17 Row-wise-user-slots view

Figure 18 Tab-wise-user-slots view

After a decent time of analysis the first option, i.e., the column-wise-user-slot view for the FE was considered in terms of feasibility and the time left in hand.

It basically consists of mini calendar-like windows that are designed for a registered user, i.e., a person who has at least once signed in on our system using their Outlook credentials, the Call button and responses from the user and the system itself.

A reference to the code used for the design of the user calendar:

```
<script src="https://code.jquery.com/jquery-3.4.1.slim.min.js"
      integrity="sha384-
J6qa4849blE2+poT4WnyKhv5vZF5SrPo0iEjwBvKU7imGFAV0wwj1yYfoRSJoZ+n"
      crossorigin="anonymous"></script>
<link href='{{url('/')}}/fullcalendar/lib/main.css' rel='stylesheet'
/>
<script src='{{url('/')}}/fullcalendar/lib/main.js'></script>
```

Figure 19 Code used for the design of the user calendar

```
<h6><small>Appointment request has been sent <strong>{{$thisRequest-
>created_at->diffForHumans()}}</strong></small></h6>
```

Figure 20 Code for the appointment sending screen

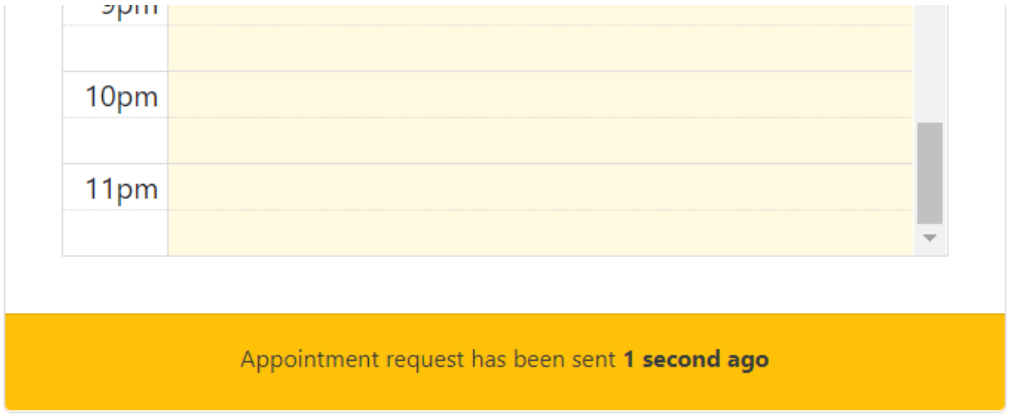


Figure 21 Appointment for meeting/visit

```
     <label style="cursor: pointer;"><input type="radio"
name="is_free" value="1" checked style="cursor:
pointer;"> <strong>Free</strong></label>
            
          <label style="cursor: pointer;"><input type="radio"
name="is_free" value="0" style="cursor:
pointer;"> <strong>Busy</strong></label>
```

Figure 22 Code for handling and showing the Busy/Free response

Figure 23 User status

4.3.3 Design of the user page or BE

The user page is roughly the same as the visitor page but there are some additional features in its layout. There is a header with the name of the system, a user symbol, the user's name and a dropdown menu. To design this portion of the page the following code was used:

```
 <div class="collapse navbar-collapse" id="navbarCollapse">
          <ul class="navbar-nav mr-auto">

          </ul>
          <ul class="navbar-nav justify-content-end">
              <li class="nav-item dropdown">
                <a class="nav-link dropdown-toggle" data-
toggle="dropdown" href="#" role="button"
                  aria-haspopup="true" aria-expanded="false">
                  @if(isset($user_avatar))
                    <img src="{{ $user_avatar }}" class="rounded-
circle align-self-center" style="width: 32px;">
                  @else
                    <i class="far fa-user-circle fa-lg rounded-circle
align-self-center" style="width: 32px;"></i>
                  @endif
                  {{ $userName }}
                </a>
                <div class="dropdown-menu dropdown-menu-right">
                  <h5 class="dropdown-item-text mb-0">{{ $userName
}}</h5>
                  <p class="dropdown-item-text text-muted mb-0">{{
$userEmail }}</p>
                  <div class="dropdown-divider"></div>
                  <a href="{{ url('/') }}/calendar" class="dropdown-
item">My Calendar</a>
                  <div class="dropdown-divider"></div>
                  <a href="{{ url('/') }}/signout" class="dropdown-
item">Sign Out</a>
                </div>
              </li>

          </ul>
```

```
        </div>
```

Figure 24 Code for header of the user page

The app/system name displaying on top of the page was basically dragged from what was put in the main file/ .env file used for database connection:

```
APP_NAME=Development-Of-Smart-Entry-Solution
```

Figure 25 Code for setting App name

Figure 26 User web page header

4.3.4 Design of the user platform

The ‘New event’ creation window was constructed with the use of the following code:

```
class Event extends Model
{
    use HasFactory;

    protected $fillable = [
        'event',
        'email',
        'subject',
        'organizers',
        'attendees',
        'start',
        'end',
        'body',
    ];
```

Figure 27 Code for New event window

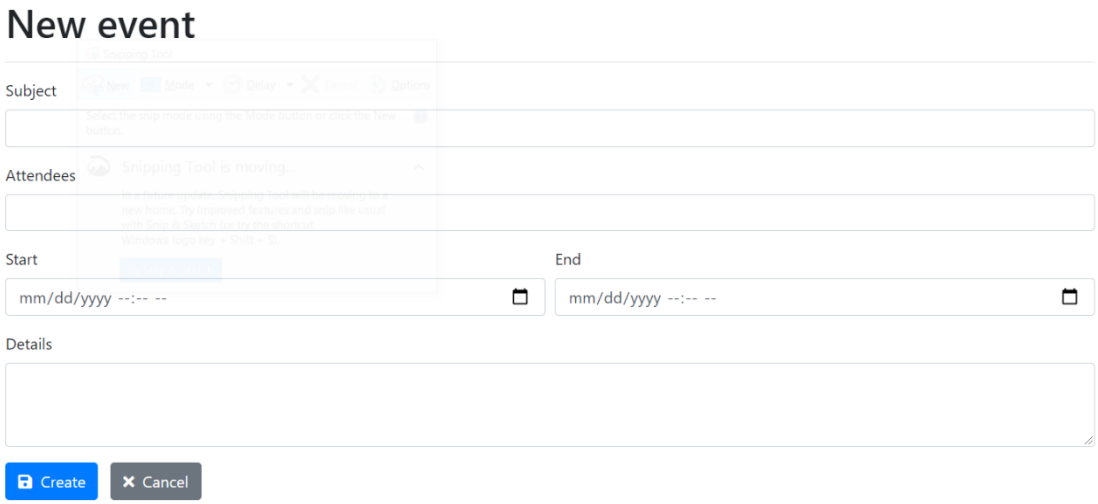


Figure 28 Creating a new event in My Calendar

A reference to the code used to design the user login page is given below:

```
<h1>Verify Yourself</h1>
<hr>
<form method="POST" action="{{ url('login') }}">
  @csrf
  <div class="form-group">
    <label>Security Code</label>
    <input type="text" class="form-control" name="code" />
  </div>
  <button type="submit" class="btn btn-primary mr-2"><i class="fa fa-
check"></i>  Proceed</button>
  <a class="btn btn-secondary" href={{ url('/') }}><i class="fa fa-
times "></i>  Cancel</a>
</form>
```

Figure 29 Code for the construction of User login page

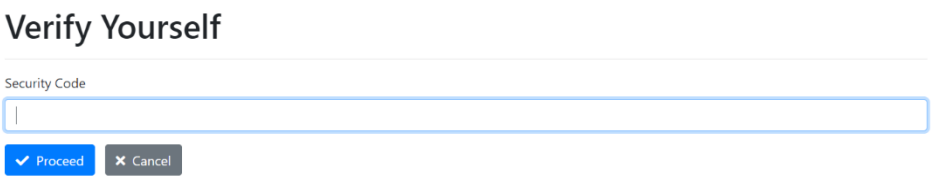


Figure 30 System asking for verification code

5 System implementation

5.1 Integration of HW/SW components

At the user end, referred to as the BE, a personal computer or a smart phone/tablet is used. The whole system basically stands on a central server and hence the user is required to employ a web browser (any web browser) to go the server link and access the system.

They are required to first type a security code number and then their MS Outlook Calendar credentials which then allows them to sign in on the system. The code number is required only at the very first time for a particular device. It is shown figure 4.1 in the previous chapter.

Next comes the usual Outlook sign in window and afterwards the user's page shows up as shown below:

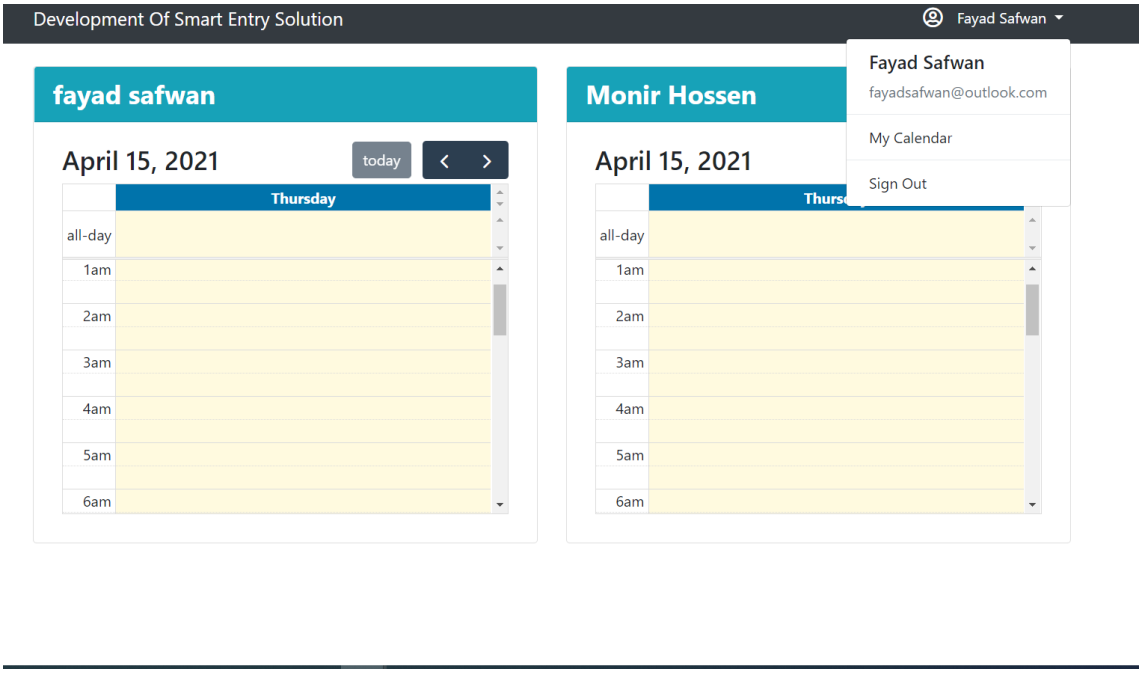
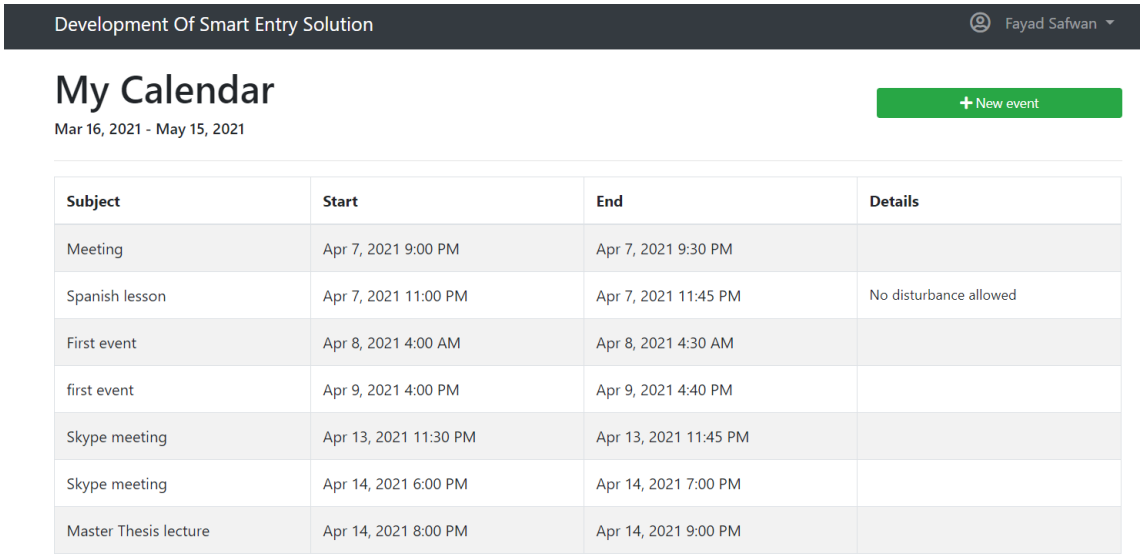


Figure 31 User account platform

From the above we can see that a user can see their time schedule as well as those of other users. However, they are only able to add or remove events to their own calendar from within the system. The dropdown arrow on the top right corner provides the means to go their calendar. The figure shows the 'My Calendar' section in the system which shows the series of events between two months' time frame. The 'New Event' button enables us to add events as shown below:



Subject	Start	End	Details
Meeting	Apr 7, 2021 9:00 PM	Apr 7, 2021 9:30 PM	
Spanish lesson	Apr 7, 2021 11:00 PM	Apr 7, 2021 11:45 PM	No disturbance allowed
First event	Apr 8, 2021 4:00 AM	Apr 8, 2021 4:30 AM	
first event	Apr 9, 2021 4:00 PM	Apr 9, 2021 4:40 PM	
Skype meeting	Apr 13, 2021 11:30 PM	Apr 13, 2021 11:45 PM	
Skype meeting	Apr 14, 2021 6:00 PM	Apr 14, 2021 7:00 PM	
Master Thesis lecture	Apr 14, 2021 8:00 PM	Apr 14, 2021 9:00 PM	

Figure 32 My Calendar

It is possible to add a new event with a few parameters added to it but it is not possible to edit the event once it is created nor is it possible to remove an event from the event list from within the system. For that the user would have to go to their account in Outlook Calendar. Figure 4.2 in chapter-3 shows the new event creating window.

Generally, a staff would sign in on their account in Outlook Calendar and add or remove or edit events. Following that the updates from Outlook calendar would go into the system. However, the user would need to stay signed in or sign in after every update on the system for their schedule within the system to get updated.

A visitor in the corridor walks to the office door and looks at the tablet screen and they see the following initial page:

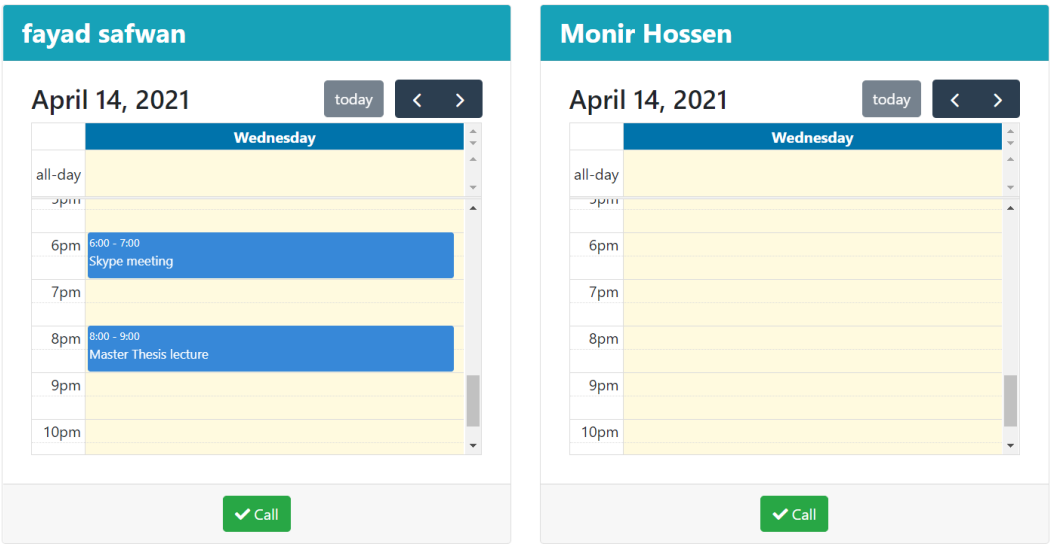


Figure 33 Default screen on the FE

From the above figure we can see that for each staff at the office there is a separate section with their name mentioned at the top. The visitor is able to see the current schedule of each staff, i.e., around the current time they are standing at the door. Also, there is an option to scroll up and down to other hours of the day as well as to the next and previous day schedules.

Now, as the visitor taps on the “Call” button at the bottom on the tab, a notification is sent to the user’s account on the system. The staff gets an alert tone in their device on which they are logged in on the system. The following window appears on the user page:

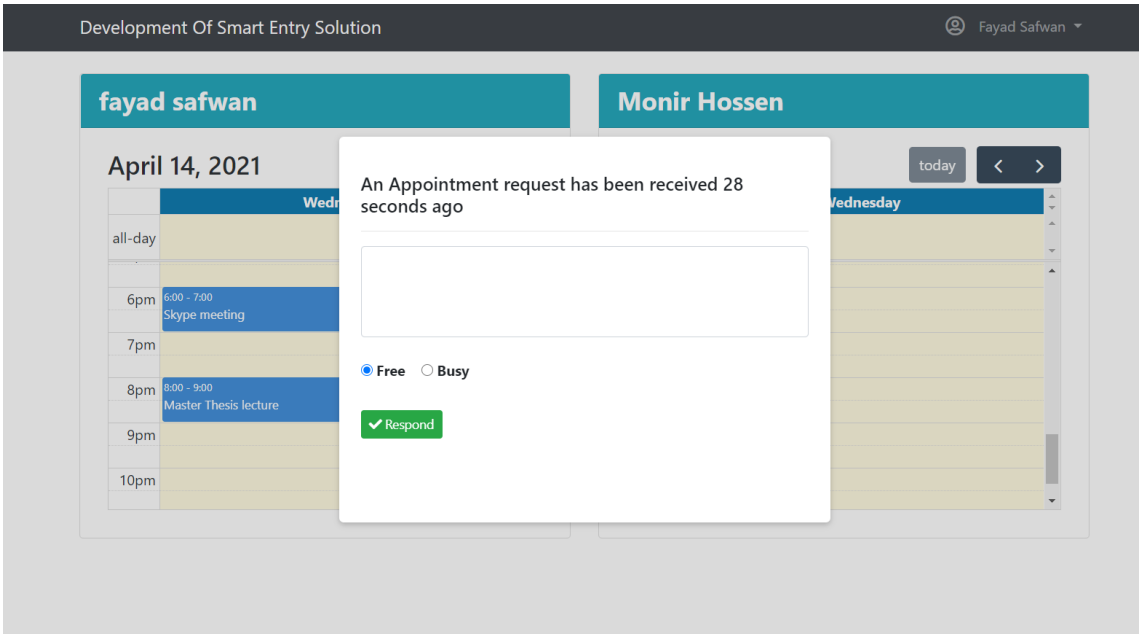


Figure 34 Entry request notification window

From the diagram we can see a window pops up with two buttons ‘Busy’ and ‘Free’ and an empty area to write any personal message to the guest. This window not only pops on the user page of the system but rather on corner of the computer screen of the user as long as the user stays logged in on the system.

Also, a sound alert has been created to ring as the notification window pops up. This is done to grab the user’s attention. This alert tone rings after every 10 seconds for a total period of two minutes. If there is no response from the staff till then the request fades away.

From here on, the user then does one of the following:

I. They click/tap on Busy
II. They click/tap on Free
III. They type a personal text for the visitor

A positive response appears in a green field while a negative response appears in a red field as shown in the following figures:

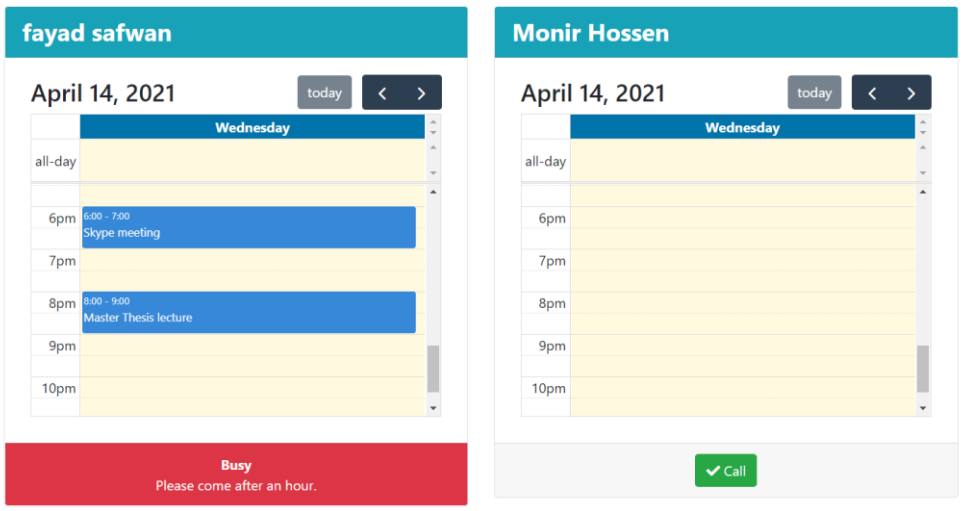


Figure 35 Staff responding negatively along with a personal text

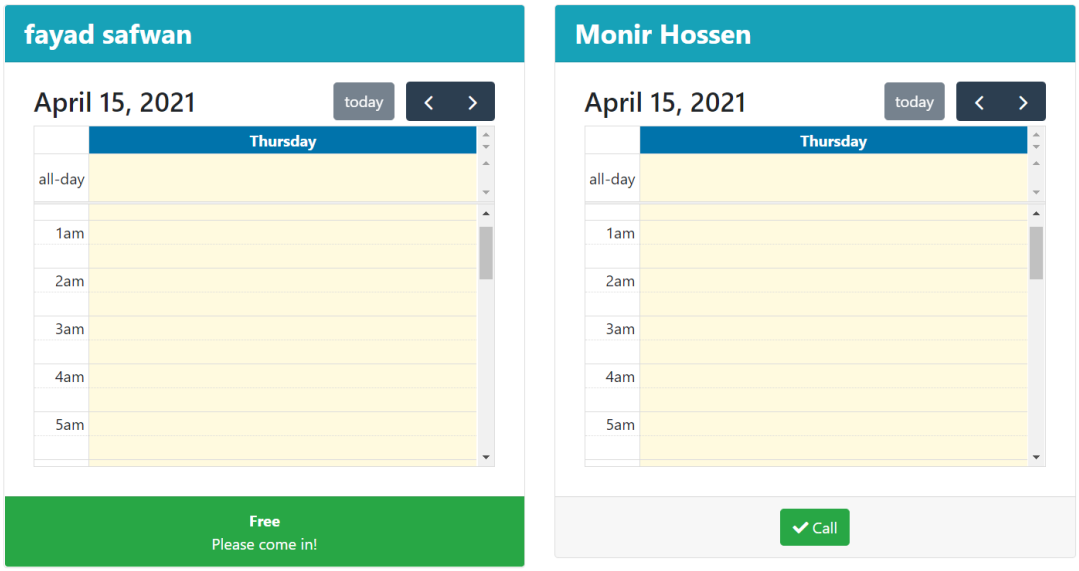


Figure 36 Staff responding positively along with a personal text

Here, the user has to select the free or busy button every time regardless of the presence of a personal text or else there might be inconsistency between the text and the status.

As an entry request is made to the staff the visitor sees a timer on the screen for a maximum period of two minutes as shown below:

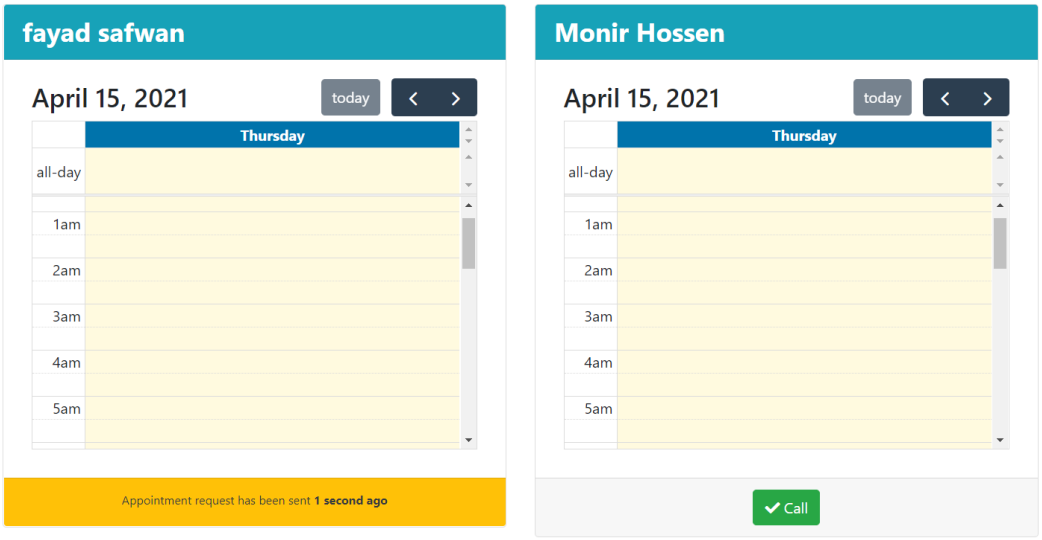


Figure 37 Timer begins as the Call button is tapped

If the user fails to respond within that time the visitor would see a message saying ‘No response’.

5.2 System verification

Our verification process started off with two users for the system. Each user with an account on MS Outlook Calendar. The users signed in on their Outlook account and updated their schedule and then logged in on our system. Following that, they went to the login page of the system in the server, first time from their devices (Personal computers). The front-end screen page was kept open in another computer.

A person was made to observe the series of changes taking place in the front-end screen as the two users operated at their ends.

The following software platforms were employed for the system verification process:

1. Windows 10
2. Windows 7
3. Google Chrome
4. Android 9
5. MS Outlook Calendar

The following hardware devices were used during the system verification process:

1. Acer Laptop with Core I3 processor
2. Asus laptop with Core I3 processor
3. Samsung J7 mobile device
4. Mercusis WIFI router

Observations made on the FE and BE:

In the beginning, the front end is completely empty as no users have registered or signed in on the system yet. As the first user signed in using the security code and their Outlook credentials, their window appeared in the middle of the front-end page as shown in the figure below:

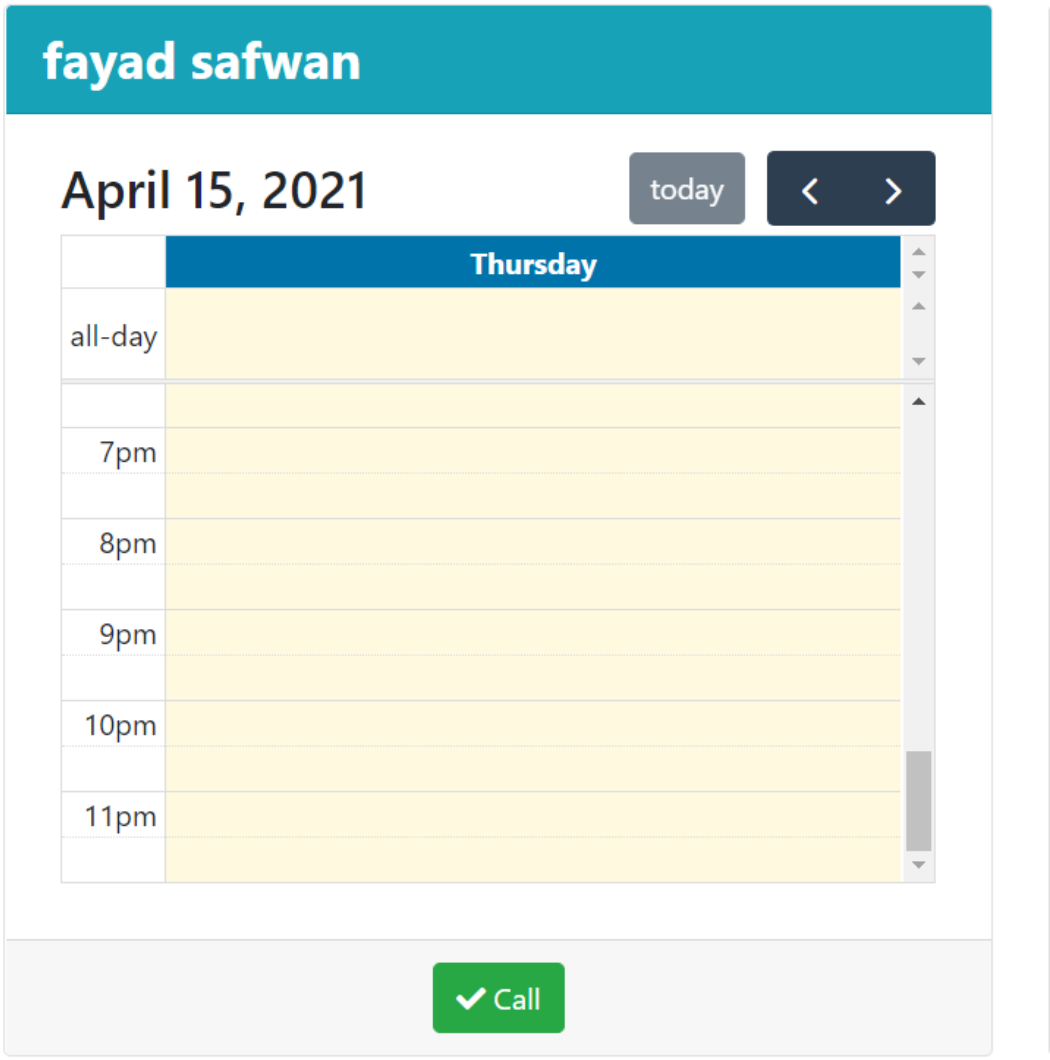


Figure 38 Front end screen with only one registered user

After that, as the second user signs in following the same procedure their window appears next to the first user's as shown below:

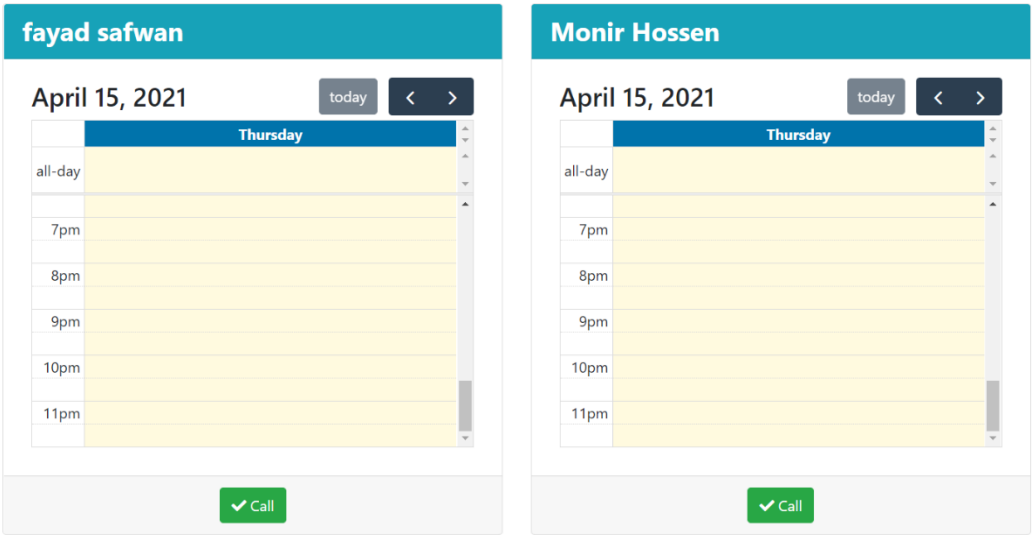


Figure 39 FE screen with two registered users

Now, events are added by the two users from their MS Outlook accounts both on the app and on MS Outlook on Google Chrome and also, from within their accounts on our system. Following that, some events were edited and deleted from MS Outlook accounts. All the updates got into the user's window in the system. Next, the 'Call' button is employed to check for different responses from the users and also what happens in the case of no response from the users.

In the whole process, a set of observations were made as seen from both the user's view and the visitor's view, noted down and classified into two categories: expected and unexpected observations.

Positive or expected observations:

1) After signing in on the system using a certain Outlook account for the first time the new user window appears on the front screen next to the existing ones (if any).
2) As a user adds a new event within the system platform the user Calendar window is immediately updated with it.
3) The pressing of Call button causes an appointment request window to appear on the user's page.
4) A sound is also heart on user's computer as the notification arrives.
5) User gets all the options mentioned in use cases for making response.
6) Busy/free responses appear on visitor screen in reasonable time.
7) Responding with a text causes the text to be shown on screen within a descent time.
8) The response ultimately goes away from the screen and we can see the Call button again.
9) Creating new event on Outlook account causes the event to appear on the system window immediately when already logged in on the smart entry system.
10) The user's calendar window view always lies over the current time events area on the list.

Negative or unexpected observations:

1) After the update of Outlook calendar, the updates do not show on the front end until and unless the user logs in on the system at least once (after each update).
2) After logging in the updates are taken in but the new window for this new user does not immediately show on the front-end screen. For that, the user has to sign in on the system at least once.
3) It takes a while for the appointment window to appear following the tapping of Call.

4) Page refreshed after one minute and thus anything the user might be typing during the refresh period would be cleared up.
5) The ‘No response’ message on the screen normally does not usually appear. The screen only goes back to the initial/start-up state.
6) On the BE, the web page often gets broken rather than getting refreshed automatically if the user is on other pages for a long time. However, as soon as the user opens that tab it comes back reloading. This takes one to two seconds. Seldom, the user might have to manually refresh the page.
7) Our users on the BE can see the titles of the events of the co-users rather than just ‘Busy’ like we see in Outlook. This is a bit of privacy violation.
8) The same privacy issue is on FE. The outsiders can also see the event details which might not be acceptable to the users sometimes.
9) When re-logging in on the system without restarting the device, it does not ask for password which could be risky in some cases.

6 Conclusion and future work

6.1 Summary

Smart entry solutions possess the capability to create new possibilities to reduced disturbance, to enhance safety and security. They can help increase work efficiency in corporate environments by offering scopes to keep things organized to a great extent. A lot of such systems or devices have been already released in the market, some are categorized as visitor management, some are smart entrance while some are being called room booking. They serve a wide variety of purposes for people's daily life but they do have some common features and employ some common methods as well.

A critical issue in developing such a system is to decide on the user cases, to decide which path to take, to arrange the order of the different functions required to design a system under a certain user story. Then comes the question of coding language selection. Different languages well-serve different aspects of the system. Of course, there is a big limitation of the width of options here for a single person as one does not most usually possess the skills of many different coding languages. Initially, a number of different possible user cases were created and two possible languages, namely, Python and PhP. Later, considering the system demands/requirements from the chosen user case and the time in hand for completion of the project it was decide that the software would be developed as a web application using PhP, HTML, Javascript and so on.

Based on the results obtained from the final implementations of the system, it can be said that we have covered almost all the requirements from the user cases except a very few, such as, keeping the event details of different users hidden from one another (like we see on MS Outlook calendar) and the visitors and also getting the automatic 'No response' message on the screen in the case of user not responding within the allocated time. Of course, there are honestly some unavoidable imperfections like lagging issues whose absence would have otherwise made the system seem much more professional.

Nevertheless, on the basis of the overall performance, it can be concluded that the initial purpose of the thesis has been accomplished. The repeatedly conducted practical realizations have been analysed in terms of goal reached or goal not reached and time factors. The results are encouraging and pave the way for further development of such a smart entrance system in the ways briefly discussed in what follows.

6.2 Future scope

In the context of this thesis, the future work can be divided into two categories:

1. Shorter-term perspective
2. Longer-term perspectives

- Shorter-term perspectives

Considering the placement of the whole on a server and the use of it through a browser acceptable or somewhat standard it is possible to solve the small issues or imperfections that we experience in the smart entry system as we use it through making changes in the PHP code at various points. Let’s look at some of the minor improvements that can be attained through the following techniques:

1. The replacement of event's title/details by simply the term 'Busy' on both ends of the system could preserve privacy of users. For that, we will need to replace the value of title key in \App\Http\Controllers\HomeController.php (Line 107).

2. The sound alert might not often be useful if the user is not using loudspeakers or headphones. In that case, we might need a separate notification window to pop up perhaps in one corner of the computer screen. For that, we will need to use some Jquery Library like Sweet Alert or Wnoty etc.

3. Automatic update of new events from MS Outlook Calendar on the system front page without having to stay logged in on the system constantly could make things much easier for the user. But it's not possible to achieve that from our arrangement. Developer will have to take a different course like the use of Python or Java app builder which will result in a different, perhaps more effective app.

4. The user end could be designed with a completely different appearance from that of the front end to make our system look more professional. This will require an additional set of coding for the back end.

5. The web application could be enclosed within an ethernet or Local Area Network to avoid access to unwanted users, i.e., users outside the organization.

- Longer term perspectives

The entire system could be developed as a desktop-based app based on the software that would not require the use of a server. The use of server unfortunately gives access to the system to basically anybody with the link to the server link. Plus, there is a hassle of the page occasionally not loading. Often times the server could be down.

In future, it could be possible to achieve that through the following-

1. We need to build an API.
2. We need a local PHP server installed on the desktop.
3. Then we can locally browse the software through Browser.
4. If we want to build an APP we need to use C# or Java App builder.

References

[1] Linh Duc Tran, Alex Stojcevski, Thanh Pham, Tony de Souza-Daw, "A smart meeting room scheduling and management system with utilization control and ad-hoc support based on real-time occupancy detection,"IEEE Sixth International Conference on Communications and Electronics (ICCE), 2016.

[2] Ilkyu Ha, "Security and Usability Improvement on a Digital Door Lock System based on Internet of Things," International Journal of Security and its Applications, Vol.9, No.8 (2015), pp.45-54, 2015.

[3] S.M Amir Khan, Md. Nazmul Islam Shuzan, Moajjem Hossain Chowdhury, M. Maksud Alam, "Smart Entrance System Using Computer Vision at Corporate Environment", IEEE ICECCT, pp.1-5, 2019.

[4] "Envoy and Open path access control." openpath, openpath, 2021, Web. 5 May 2021.

[5] "Smart office access for your employees." nello, LOCUMI LABS GmbH, 2021, Web 5 May 2021.

[6] Yoo-Joo Choi, D.-C. Kang, W.-D. Cho, K.-J. Kim, "Smart Entrance Monitoring System Using Clothing Color Analysis in Multi-Camera Environment," International journal on information, Vol.14, 2011.

[7] "Haltian - Thingsee PRESENCE." Bot-Hive, Haltian, 2020, Web. 6 May 2021.

[8] "Extron Room Scheduling." Extron, Extron.

Appendix 1- Non-exclusive licence for reproduction and publication of a graduation thesis

I Safwan Talukder Fayad

1. Grant Tallinn University of Technology free licence (non-exclusive licence) for my thesis Development of Smart Entry/access Solution for Academic Office, supervised by Ants Koel.

1.1. to be reproduced for the purposes of preservation and electronic publication of the graduation thesis, incl. to be entered in the digital collection of the library of Tallinn University of Technology until expiry of the term of copyright;

1.2. to be published via the web of Tallinn University of Technology, incl. to be entered in the digital collection of the library of Tallinn University of Technology until expiry of the term of copyright.

2. I am aware that the author also retains the rights specified in clause 1 of the non-exclusive licence.

3. I confirm that granting the non-exclusive licence does not infringe other persons' intellectual property rights, the rights arising from the Personal Data Protection Act or rights arising from other legislation.

10.05.2021

Appendix 2- Code for database connection

```
APP_NAME=Development-Of-Smart-Entry-Solution

APP_ENV=local

APP_KEY=base64:ywYYlrEYUAepgP/qhEygmuq7wpzXwjWBxoXrc+34oXc=

APP_DEBUG=true

APP_URL=http://localhost:8000


LOG_CHANNEL=stack

LOG_LEVEL=debug


DB_CONNECTION=mysql

DB_HOST=127.0.0.1

DB_PORT=3306

DB_DATABASE=dses

DB_USERNAME=root

DB_PASSWORD=


BROADCAST_DRIVER=log
```

```
CACHE_DRIVER=file

QUEUE_CONNECTION=sync

SESSION_DRIVER=file

SESSION_LIFETIME=120


MEMCACHED_HOST=127.0.0.1


REDIS_HOST=127.0.0.1

REDIS_PASSWORD=null

REDIS_PORT=6379


MAIL_MAILER=smtp

MAIL_HOST=mailhog

MAIL_PORT=1025

MAIL_USERNAME=null

MAIL_PASSWORD=null

MAIL_ENCRYPTION=null

MAIL_FROM_ADDRESS=null

MAIL_FROM_NAME="${APP_NAME}"


AWS_ACCESS_KEY_ID=

AWS_SECRET_ACCESS_KEY=
```

AWS_DEFAULT_REGION=us-east-1

AWS_BUCKET=

PUSHER_APP_ID=

PUSHER_APP_KEY=

PUSHER_APP_SECRET=

PUSHER_APP_CLUSTER=mt1

MIX_PUSHER_APP_KEY="${PUSHER_APP_KEY}"

MIX_PUSHER_APP_CLUSTER="${PUSHER_APP_CLUSTER}"

//This section is for Microsoft outlook registration

OAUTH_APP_ID=39a3c938-beb3-4eb2-a0fb-fd7e2b84de85

OAUTH_APP_SECRET=sthnZNH092_oegXXRG16^|)

OAUTH_REDIRECT_URI=http://localhost:8000/callback

OAUTH_SCOPES='openid profile offline_access user.read mailboxsettings.read calendars.readwrite'

OAUTH_AUTHORITY=https://login.microsoftonline.com/common

OAUTH_AUTHORIZE_ENDPOINT=/oauth2/v2.0/authorize

OAUTH_TOKEN_ENDPOINT=/oauth2/v2.0/token