## horizontal line



Health Screening Solutions

12.12.2019

**─**

The Googlizers

Alli Fox, Ben Zell, Emmett Simmons,

Zawaad Shah, Aaron Nguyen

Augustana University

Sioux Falls, SD 57197

# Preface

*Written by: Alli Fox \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_*

*Reviewed by: Benjamin Zell \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_*

*Aaron Nguyen\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_*

*Zawaad Shah \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_*

*Emmett Simmons\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_*

This document is expected to be read by the health administrator/product owner as well as the individual tasked with the long-term maintenance of the product.

**V1.0** - This is the initial prototype as a minimum viable product. This version was presented to the product owner as an example of what the product can do.

# 

# Introduction

*Written by: Alli Fox \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_*

*Reviewed by: Benjamin Zell \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_*

*Aaron Nguyen\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_*

*Zawaad Shah \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_*

*Emmett Simmons\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_*

The current health screening administrator has been manually booking the appointment screenings starting by sending a mass email to the Augustana University employees and coordinating a proper time for each employee individually. This is a time consuming and inefficient process that requires the admin to set aside multiple days to ensure all screenings are booked. With this system in place, the process of booking screening events for Augustana employees will become easier, quicker, and more efficient.

Our system works primarily through Google to set up appointment slots for the screening, and then notify all employees of the screening that was set up. If the employee cannot book the appointment themselves or choose not to, they will have the ability to contact the admin and the admin will be able to manually book an appointment for them or place them on a waitlist. This system will also notify the employee of their booked slot and notify them of their appointment when they wish to be notified. The admin can also export the calendar to a spreadsheet for various storage capabilities and functions.

The system set in place by the health screening administrator, as stated above, is in need of total replacement. The system designed will replace the admins current process for a more streamlined experience and will work well with the Augustana University email processes already in place.

Some of the primary objectives of hosting health screenings is to promote a healthy community on campus, as well as to help improve the bottom line of the financial statements of the university with lowered health insurance costs. This system will make health screenings a less arduous process as well as be a financially responsible decision. By using student/employee labor to design and construct the system as well as basing the system on the google appointment slots, which is something Augustana was paying to receive previously, this system will not incur any additional costs that wouldn’t be paid originally while decreasing the workload and wasted time of the administrator and the employees being screened.

# 

# Glossary

*Written by: Aaron Nguyen \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_*

*Reviewed by: Alli Fox \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_*

*Zawaad Shah \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_*

*Emmett Simmons\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_*

*Benjamin Zell\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_*

**.xls/.xlsx** - file formats for Microsoft Excel spreadsheets. The .xls format is default in Microsoft (MS) Excel version 2003 or before, while the .xlsx format is the current default format and was implemented since MS Excel version 2007 and after.

**Admin** - an abbreviation for “Administrator.”

**Administrator** - a person responsible for organizing and operating a business. He or she has a higher authority in a software system.

**Attachments** - additional documentations or files that are included with an email or a message for the recipients’ information.

**Browser** - a program with a graphical user interface for displaying HTML files, used to navigate the World Wide Web. E.g. Google Chrome, Microsoft Edge, etc.

**Client** - a person or organization using the services of a lawyer or other professional person or company.

**Database** - a structured set of data held in a computer, especially one that is accessible in various ways.

**Employee** - a person employed for wages or salary, especially at nonexecutive level.

**Export** - to take data from one program and open in another, usually changing the file format in the process.

**Notification** - a brief message to attract the user’s attention or to remind the user of a current or upcoming event.

**Patient** - a person receiving or registered to receive medical treatment.

**Report** - a detailed documentation produced after a screening to record various information about the event for future reference.

**Screening** - a medical procedure designed to specifically diagnose a common health issue.

**Slot** - a unit of time during which a screening will be carried out. A patient would register a single slot for each type of screening.

**Summary** - a brief statement or account of the main points of a screening.

**System** - a set of integrated devices that input, output, process, and store data and information.

# 

# User Requirements

*Written by: Alli Fox \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_*

*Benjamin Zell\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_*

*Aaron Nguyen\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_*

*Zawaad Shah \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_*

*Emmett Simmons\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_*

## **User system**

* 1. **Booking**
     1. The user shall be able to view time slot availability
     2. The user shall store booking information to the database
     3. The user shall allow a user to access/add/update information after confirming the user authenticated profile.
     4. The user shall be able to generate user notes and send them to the admin along with the appointment
     5. The user shall be able to view all available types of screenings
     6. The user shall be able to view all available time slots for a screening
     7. The user shall be able to book time slot for screening
     8. The user shall be able to add personal notes to the appointment
  2. **Appointment Updates**
     1. The user shall be able to change or edit appointment times
  3. **Notification**
     1. The user shall receive a confirmation email after booking a time slot, and send it to the user, which includes screening time and instructions.
     2. The user shall see a confirmation on screen of the appointment they signed up for
     3. The user shall get an email for new screenings
     4. The user shall get a reminder notification
     5. The user shall be able to receive specific instructions for each screening
     6. Attachments shall be sent in confirmation email
  4. **Authentication**
     1. The user shall be required to sign in to their Google accounts in order to book
     2. The user shall have the option to create, update, store and manage patient info

## 

## **Admin System**

* 1. **Schedule**
     1. The system shall allow admin to add time slot for a certain kind of screening
     2. The administrator shall be able to setup new screenings
     3. The administrator shall be able to delete screenings
     4. The administrator shall be able to update screenings
     5. The administrator shall allow screening to occur multiple days, at various intervals for various users at a time
  2. **Booking**
     1. The system shall allow admin to book/delete a screening for a user
     2. The system shall allow admin to access all user notes
  3. **Notifications**
     1. The administrator shall be notified when a time slot is occupied
  4. **Attachments**
     1. The administrator shall be able to upload attachments to emails for employees
  5. **Authentication**
     1. The administrator shall sign in with Google prior to having access to client records
  6. **Reporting**
     1. The administrator shall have the ability to print out the screening schedule
  7. **Database**
     1. The administrator shall have the ability to record attendance for screenings in excel
     2. The administrator shall have the ability to store all patient records
     3. The administrator shall be able to search for all patient records in excel
     4. The administrator shall be able to export screening schedule to a spreadsheet
     5. The administrator shall be able to view scheduled appointments in the designated google calendar

# System Architecture

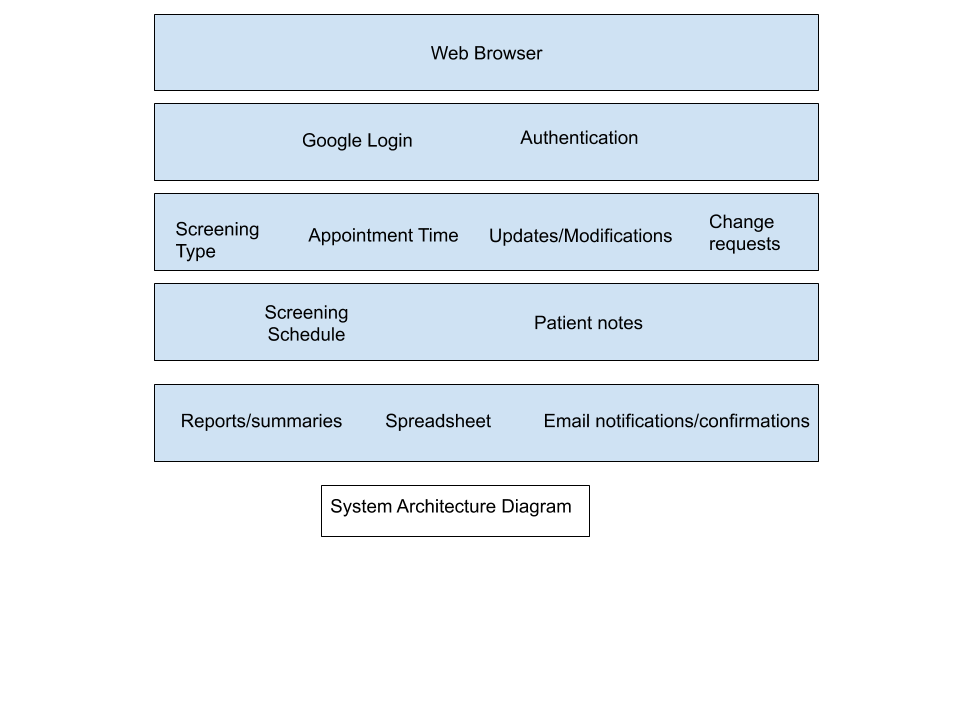
*Written by: Zawaad Shah \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_*

*Reviewed by: Benjamin Zell\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_*

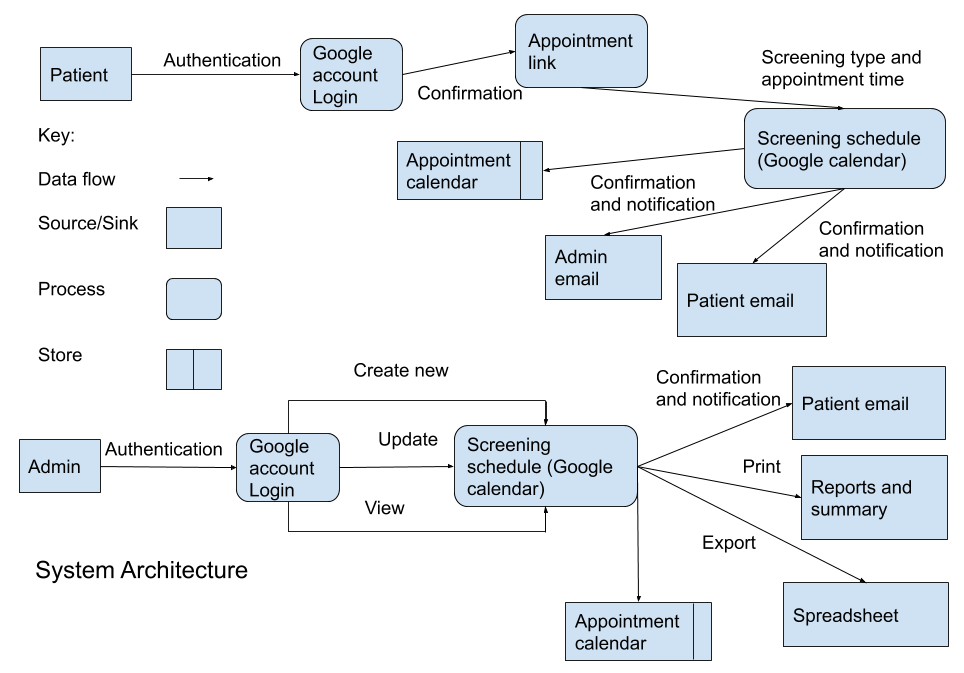
*Aaron Nguyen\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_*

*Alli Fox \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_*

*Emmett Simmons\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_*



The system architecture consists of a web browser which is required to access the Google Calendar web application. Authentication is carried out via Google accounts. The calendar allows creation of a new screening schedule, appointment time and updates and modifications to the current schedule. The appointment schedule is viewed via Google Calendar. Reports can be printed and/or exported to a spreadsheet. Notifications and confirmations and sent via Gmail.



The system is dependent almost entirely on the architecture of Google Calendar. The user will be required to login through their Google account in order to access the calendar and appointment schedules. Google takes care of authenticating user information. A new appointment schedule is created through the calendar and both confirmation and notifications are sent via email to the users. The user can access the appointment schedule to register for a time using the appointment link which is emailed. Schedule information is stored in the calendar page and can be exported into a spreadsheet and/or printed.

# System Requirements

*Written by: Alli Fox \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_*

*Aaron Nguyen \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_*

*Zawaad Shah \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_*

*Benjamin Zell \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_*

*Emmett Simmons \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_*

1. **Functional requirements**
   1. The system shall be implemented as a Google calendar
   2. Admin access to the system shall be provided through a web-browser and Google
   3. Access to the appointment page shall require Google authentication
   4. The system shall send appointment updates via email
   5. The system shall send external information(attachments) to the patient via email
   6. The system shall have the ability to export a summary of appointment times/patient information into a .xls/.xlsx file
   7. The system shall update the screening schedule immediately with each submission/modification from the patient
   8. The system shall generate notification emails for the user, from user specification
   9. The system should be mobile friendly
   10. When the admin creates a screening, an email shall automatically be sent to all users the admin has on file notifying them of the upcoming screening and providing a link to the sign up form
2. **Non-functional requirements**
   1. Patient information shall only be visible to the patient and system administrator
   2. The login authentication shall not be arduous
   3. The system shall not take longer than 5 seconds to switch to the next web page
   4. The system shall be continuously available at all hours
   5. The user shall be able to sign up for multiple screenings simultaneously
   6. Spouses of users shall be able to book a screening
   7. Users should be asked to leave additional contact information (Ex. Phone number) in the comment box
   8. The excel sheet shall have separate tables including the employees’ information, attendance from previous screenings, waitlists from previous years, and the upcoming screening schedules

# 

# System Models

*Written by: Benjamin Zell \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_*

*Reviewed by: Alli Fox \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_*

*Zawaad Shah \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_*

*Aaron Nguyen\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_*

*Emmett Simmons\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_*







# 

# System Evolution

*Written by: Alli Fox \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_*

*Reviewed by: Benjamin Zell \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_*

*Zawaad Shah \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_*

*Aaron Nguyen\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_*

*Emmett Simmons \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_*

Since the system is largely browser based, there is little to worry about with hardware evolution. However, the system is heavily reliant upon processes supplied by both Google and Microsoft. If either of these companies were to drastically change or stop supporting these processes, the system would need to be changed dramatically if not entirely replaced. Fortunately, the odds of this happening are very slim. There is a higher chance of scope change or the user requirements changing in the future.

The system is based upon Google appointment slots. For the most part, when Google updates this process of their calendar process, this system will still function as required. If/when dramatic changes occur, they will be handled accordingly. If this process were to lose support or be removed, our system would cease to exist or function. This has a low possibility of happening.

The calendar will be able to be exported to an Excel sheet when the administrator wishes to. On the slim chances Excel is no longer available to be used, Google sheets can also be used with limited loss of functionality. The odds of Excel not being available for use in the future is also very unlikely to happen.

In the future, the look and feel of the application should be enhanced for a better user experience.

# Appendices

*Written by: Emmett Simmons \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_*

*Reviewed by: Benjamin Zell \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_*

*Zawaad Shah \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_*

*Aaron Nguyen\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_*

*Alli Fox \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_*

Script for parsing calendar files

//This script sits in a folder with a ical file that has been saved as a txt and creates a new txt that can be opened and read easily in Excel

import java.util.\*;

import java.io.\*;

public class ParseIcal

{

public static void main(String[] args) throws FileNotFoundException

{

Scanner consoleInput = new Scanner(System.in);

System.out.print("Please write the name of file to be parsed:");

String parsingFileName=consoleInput.next();

Scanner in = new Scanner(new File(parsingFileName+".txt"));

PrintWriter output = new PrintWriter(parsingFileName+"EXCEL.txt");

output.printf("Screening\tUser\tStart\tEnd\n");

while(in.hasNextLine())

{

String nextLine = in.nextLine();

String currentEvent = "";

if(nextLine.startsWith("BEGIN:VEVENT"))

{

currentEvent = currentEvent+in.nextLine()+"\n";

while(!in.nextLine().startsWith("END:"))

currentEvent = currentEvent+in.nextLine()+"\n";

}

String[] event = parseEvent(currentEvent);

output.printf("%s\t%s\t%s\t%s\n",event[0],event[1],event[2],event[3]);

}

output.close();

in.close();

}

//Pre: A String beginning with "BEGIN:VEVENT" and ending with "END:VEVENT"

//Return: An array of Strings containing [Event name, User Name, Event Start, and Event End]

public static String[] parseEvent(String event)

{

Scanner f = new Scanner(event);

String startDate="";

String endDate="";

String userName="";

String eventName="";

while(f.hasNextLine())

{

String pass = f.nextLine();

if(pass.startsWith("DTSTART:")) startDate=parseDate(pass.substring(8,23));

if(pass.startsWith("DTEND:")) endDate=parseDate(pass.substring(6,21));

if(pass.startsWith("SUMMARY:"))

{

int start = pass.indexOf("(");

int end = pass.indexOf(")");

userName=pass.substring(start+1,end-1);

}

if(pass.startsWith("DESCRIPTION:")) eventName=pass.substring(12);

}

String[] out=new String[4];

out[0]=eventName;

out[1]=userName;

out[2]=startDate;

out[3]=endDate;

return out;

}

//Pre: A String Containing 8 numbers representing the date and 8 characters representing the local time

//Return: A String of the form "Month Day, Year Time:Of:Day"

public static String parseDate(String dateCode)

{

char[] date = dateCode.toCharArray();

String year = ""+date[0]+date[1]+date[2]+date[3];

String monthCode = ""+date[4]+date[5];

String month="";

switch(monthCode)

{

case "01": month="January"; break;

case "02": month="February"; break;

case "03": month="March"; break;

case "04": month="April"; break;

case "05": month="May"; break;

case "06": month="June"; break;

case "07": month="July"; break;

case "08": month="August"; break;

case "09": month="September"; break;

case "10": month="October"; break;

case "11": month="November"; break;

case "12": month="December"; break;

}

String day=""+date[6]+date[7];

String hour=""+date[9]+date[10];

String minute=""+date[11]+date[12];

String second=""+date[13]+date[14];

return month+" "+day+", "+year+" "+hour+":"+minute+":"+second;

}

}