

12/06/2023



Fall 2023 - ACS Software Engineering

Group1 – Rakesh Kumar Raushan, Omer Yurdabakan,

Renee Suresh Manukonda, Sravya Sri Jagu

Client – Sweetwater Music Center

Project – Room and Gear Booking System 3.0

Room and Gear Booking System 3.0

Final Documentation

Table of Contents:

Part 1 – Introduction

Purpose

Goals and Objective

Constrains

Part 2 – Project Glossary

Part 3 – Use case & Class Models

Part 4 – Design & Implementation Modules

Part 5 – Demo Screens

Part 6 – Testing Reports

Part 7 – Development Tools and Read Me Document

Part 8 – Code Design and Standards

Part 9 – Member Contributions

Part 10 – References

Part 1

Introduction

- Purdue Fort Wayne Sweetwater Music Center provides a tool called Room and Gear Booking system.
- User (Student/Teacher) can create/update/cancel an event such as room and/or gear booking.
- Airtable database is used to keep track of user, gear, and event data, on top of Next.js app.
- The Airtable is database is getting expensive due to increasing subscription prices.
- Clients request the discovery of alternative tools.
- Baserow is suitable for a replacement: Open-source and free or Sass version available.

Purpose

- Main Goal: Create an API for backend calls to the NextJS platform securely using authentication.
- Enhance the Next.js app with secure sign in, increase the protection of user data and database backend.
- Move Airtable API calls from frontend to the backend, so that Airtable API keys are not exposed.

Goals and Objectives

Our primary objective for this project is to

- Enhance the Room and Gear Booking System application with secure sign in using Microsoft Azure Active Directory.
- Notify the Admin via email when a user without having the required permission tries to book an event.
- Hide the API call from front end to not expose user and system data also enhance room and gear booking logic.
- Research on the Replacement of Airtable database with Baserow database to keep facilitating the event scheduling and room booking tool.

We are exploring a secure authentication mechanism using Next-Auth, also started exploring Baserow database as alternative, and gathering user requirements such as booking gear in advance with availability validation logic, an intuitive interface for booking gear and rooms, a system capable of handling high volumes of requests and providing quick response times, and a secure login portal to protect sensitive user data. We started building a prototype for the improved booking system that included a user-friendly interface for booking gear and rooms, a system for validating gear availability and preventing double bookings, a secure login portal for user authentication, and a scalable system that could adapt to the growing demands of the user base.

On top of that, we are committed to enhancing the app's overall usability and making sure it can manage a high volume of requests from multiple users while maintaining liability and security.

Constraints

Below are the constraints we are anticipating

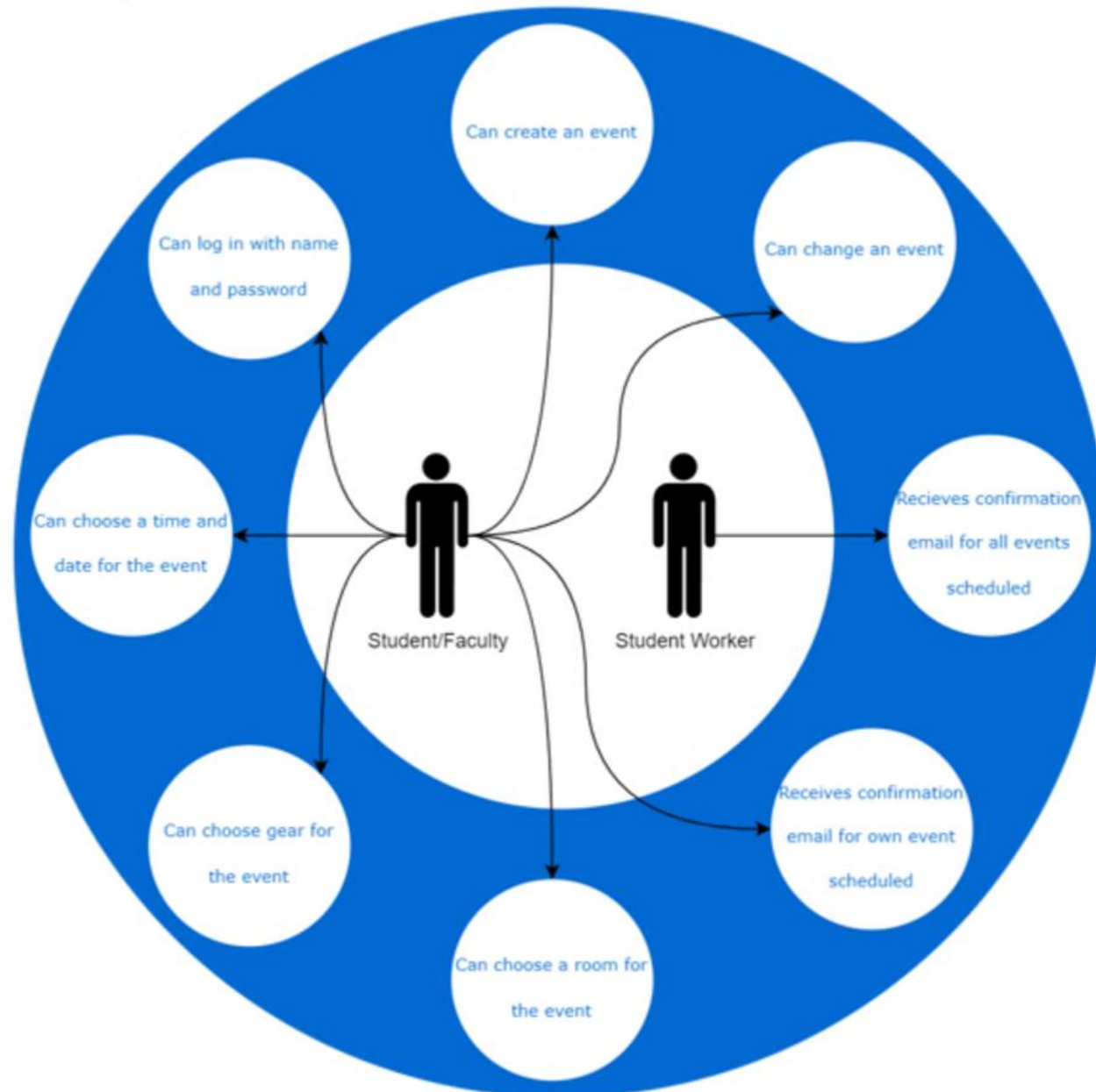
- Time: We have a strict deadline, which pushes us to discuss with the client, prioritize the work, and finalize the final requirement.
- Budget: Our budget was limited, so we need to produce cost-effective solutions and prioritize features based on their importance and impact on the user experience.
- Technical Limitations: The existing technology stack, including Airtable/Baserow and React.js and Next.js, imposed certain limitations on the implementation of new features, so we need to find compatible solutions that worked well with the current setup.

Part 2 - Project Glossary

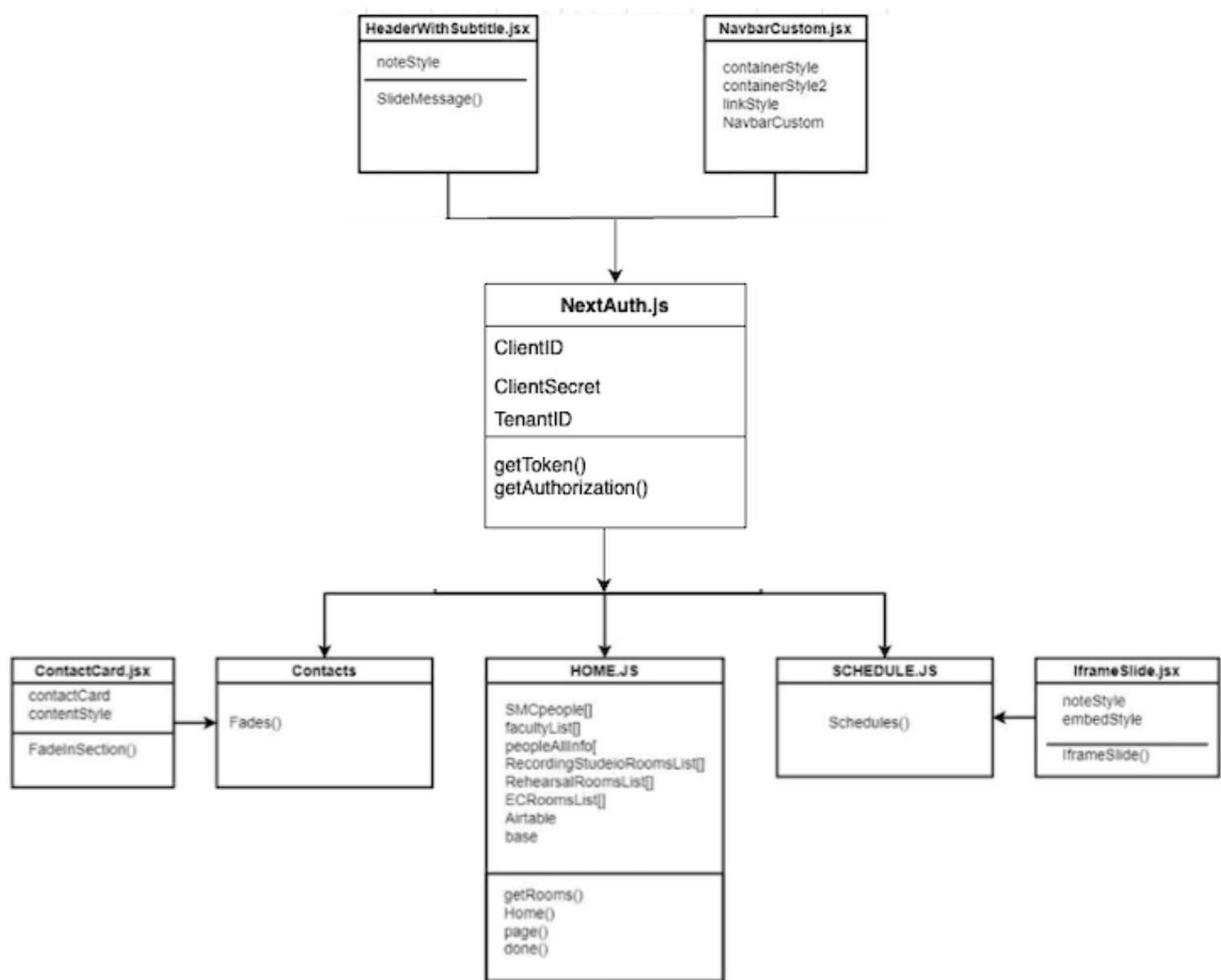
- SMC: Sweetwater Music Center.
- User Interface: The parts of program that users can interact with, including buttons, menus, and forms.
- User Experience: The overall experience that user has while interacting with program, including ease of use and satisfaction.
- Security: To provide a secure sign in using Microsoft Azure Active Directory login.
- Prototype: A working model of a program that is issued to test the user interface and user experience before the final product is released.
- Accessibility: The design of a program so that it is usable by people with disabilities, such as hearing impairments.
- Usability Testing: The process of testing a program with real users to evaluate its ease of use, efficiency, and effectiveness.
- Responsive Design: The design of a program to adapt to different screen sizes and devices.
- Component: In React, components are independent and reusable bits of code. You can think of them as classes.
- Airtable: Airtable is used to create databases that hold data for users to access, store, and organize.
- Front-end: The front-end of a program or project is the part that the user can see and includes the code necessary for how the webpage looks and how users can access it.
- Back-end: The back end of a program is how the program works and functions and includes things that are not accessible to the user, such as the system, logic, and data.
- JavaScript: JavaScript is a front-end programming language.
- React.js: React is a front-end JavaScript framework.
- GitHub: GitHub is a platform that hosts code and programs. It is accessible from anywhere and allows multiple people to work on a project.
- Vercel: Vercel is a platform for developers that provides the tools, workflows, and infrastructure you need to build and deploy your web apps faster, without the need for additional configuration
- Gear: As used in this document, refers to audio and video equipment available for students and faculty at the Sweetwater Music Center.

Part 3 – Use case & Class Models

Use Case Model

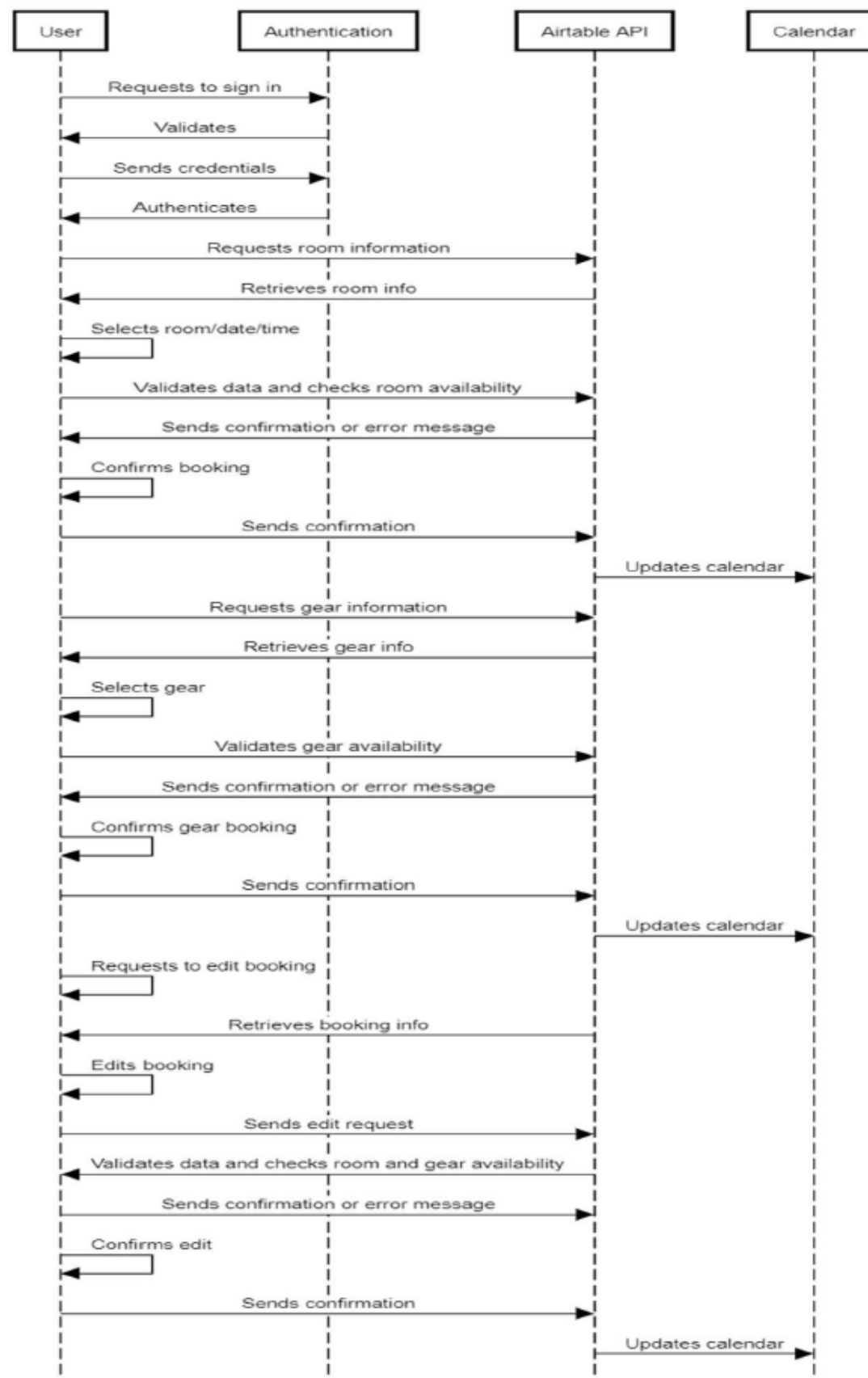


Class Module



Sequence Model

Room and Equipment Booking System



Part 4 – Design & Implementation Modules

Use Case Templates

Name	Create an Event (Authenticate User)
ID	SD1
Description	Provide a Secured provide a secure sign in using Microsoft Azure Active Directory login.
Organizational Benefits	It provides data security
Assumptions	The user data base Airtable exist
Trigger	When user login and open the page to book an event
Pre-condition	The existing user login page should be working to replace the authentication mechanism
Post-condition	User should be authenticated successfully, and book and event and API key should not get exposed.

Name	Create an Event (Notify Admin)
ID	SD2
Description	Provide a mechanism which will send a notification email when user is not already exist in system to book an event.
Organizational Benefits	It provides more security to the system, so administration has more visibilities.
Assumptions	Users should be able to login successfully.
Trigger	When user try to book an event
Pre-condition	Users should be able to create event successfully
Post-condition	Users should be able to see the error message.

Authentication component Detailed Description:

Authentication flow has been implemented using NextAuth and Microsoft Azure Active Directory (Azure AD)

- NextAuth is a popular authentication library in the JavaScript ecosystem, often used with frameworks like Next.js.
- Microsoft Azure Active Directory (Azure AD) is a cloud-based identity and access management service provided by Microsoft. Integrating NextAuth with Microsoft Azure AD allows to enable authentication in the application using Azure AD as the identity provider.
 - clientID:
 - The Client ID, also known as the "Application ID" or "App ID," is a unique identifier assigned to your application when you register it with the identity provider (such as Azure AD in this case).
 - clientSecret:
 - The Client Secret is a confidential piece of information that acts as a password for your application. It adds an additional layer of security to the authentication process.
 - tenantID:
 - The Tenant ID, also referred to as the "Directory ID," represents the Azure Active Directory instance to which your application belongs.
 - getToken():
 - Obtaining an access token, for the user to the authorization server's login page
 - getAuthorization():
 - Authorize user by the server when passing a valid token.

Notification component Detailed Description:

- Airtable itself does not have built-in functionality for sending email notifications directly. However, it can set up email notifications in combination with other services or tools that integrate with Airtable.

- When the condition matches in the database table it sends the email to the configured automation when the automation is on, and automation provides a configuration of the email recipients and email body.

Part 5 – Demo Screen

Home Page:

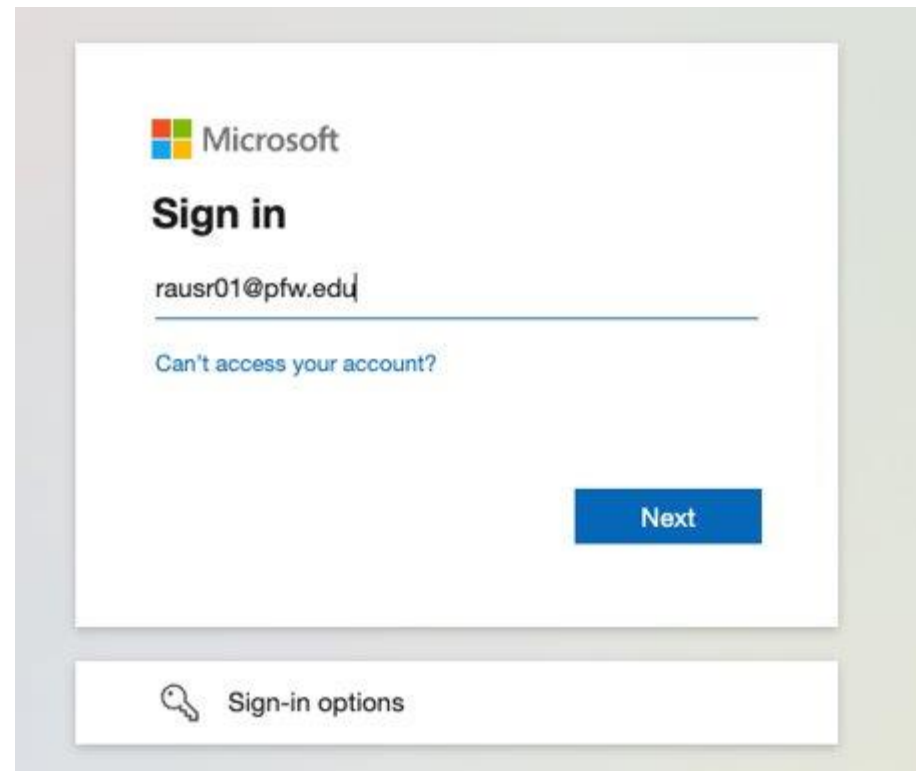
Sign In:



Popup screen to Sign in using Microsoft Azure AD



Enter Microsoft Email Address



Enter Microsoft AD credentials:



Microsoft page after successful authentication:



Login successful

PURDUE UNIVERSITY.
FORT WAYNE

[Home](#) [Book an Event](#) [Schedules](#) [Gear](#) [Contact Us](#) [Sign Out \(Rakesh Kumar Raushan\)](#)

Schedule SMC Events

Everyone can take advantage of scheduling time in the edit & collaboration spaces in the SMC building. Approved students registered for certain classes have privileges to schedule time in the recording studio, rehearsal room and control room.

[Click here to Book an Event](#)

Create Event:

Event Details

Session Title
MusicPFW

Event Type
Summer Booking 🌈

Intended Use

Room Selection

⚠️ If the Edit & Collaboration Spaces is selected, option to add gear(s) to your booking will be available at the end of the form.

Room Type
Recording Studio 📺

Select studio room(s)
SMC (Rehearsal Room) 📺

Session Time

⚠️ Based on your chosen Session Time, we will notify you with the availability of the room(s) selected above.

Select Start Date
November 20, 2023 10:

Select End Date
November 21, 2023 11:

check availability

Courses

Successfully created Event:

Gear Checkout

Need to checkout gear for the event?

☒ Gear check-out

Filter Gear List

Presonus: Monitor Station v2 (2 of 2)

Focusrite: Scarlett 2i2

JBL: EON610 (2 of 2)

Boss: FS-5L Foot Switch

Shure: SE 215 (4 of 4)

Shure: SE 215 (1 of 2)

Submission Successful!
Please check your inbox for booking confirmation.

Chosen Gear:

JBL: EON610 (1 of 2)

SUBMIT

Event Created in Airtable Database:

ACS Fall 2023 SMC Building (Development)

Views

All Events

4 hidden fields

Filter

Group

Sort

Color

Share and sync

Find a view

Approved Time

Notion/Booking Embeds

SMC Booking Form

Rehearsal Space Booking Form

Recording Studio Booking Fo...

SMC Upcoming Events

All Event

SMC Studio

All Edit Suites

SMC Rehearsal Room

More collaborative views

All Events

Create...

Grid

Form

Calendar

Gallery

Kanban

Timeline

Event	Last Modified By	Event Name	Status	Event Type	Students	Student Email
D'Wan Ellington: 09/19/2023 9:00 PM		D'Wan Ellington	Booked	Student Project		
N&C Rehearsal: 09/19/2023 8:31 PM		N&C Rehearsal	Booked	Rehearsal		
Elju and co.: 09/19/2023 8:00 PM		Elju and co.	Booked	Rehearsal		
Applied Guitar: 09/19/2023 6:00 PM		Applied Guitar	Booked	Class		
Practice: 09/19/2023 6:00 PM		Practice	Booked	Rehearsal		
Music Products Merchandising: 09/19/2023 5:00 PM		Music Products Merchand...	Booked	Class		
Popular Music Ensemble 1: 09/19/2023 5:00 PM		Popular Music Ensemble 1	Booked	Class		
Mixing : 09/19/2023 5:00 PM		Mixing	Booked	Student Project		
Practice (again): Canceled		Practice (again)	Canceled	Rehearsal		
Applied Guitar: 09/19/2023 4:00 PM		Applied Guitar	Booked	Class		
Recording 3 project: 09/19/2023 4:00 PM (Richard Chang)		Recording 3 project	Booked	Recording Session	Richard Chang	richardchang@email.com
practice drums: 09/19/2023 4:00 PM		practice drums	Booked	Rehearsal		
Performance Class: 09/19/2023 3:00 PM (nathan young)		Performance Class	Booked	Class		
drummin: 09/19/2023 3:00 PM		drummin	Booked	Rehearsal		
Music Publishing: 09/19/2023 1:30 PM		Music Publishing	Booked	Class		
MUSC 18241 Intro to Performance Techniques: 09/19/2023 12:30 PM		MUSC 18241 Intro to Perfo...	Booked	Class		
Applied Piano: 09/19/2023 11:00 AM (Sophia Lee)		Applied Piano	Booked	Class		
MUSC 18241 Intro to Performance Techniques: 09/19/2023 10:00 AM	Rakesh Raushan	MUSC 18241 Intro to Perfo...	Booked	Class	Rakesh Raushan	raus01@pfw.com
Event Created						

User not having permission to book an event:

PURDUE UNIVERSITY
FORT WAYNE

Home

Book an Event

Schedules

Gear

Contact Us

Sign Out (Rakesh Kumar Raushan)

Error

Looks like user is not registered in the system. — An Email is sent to the Administrator to get this fixed!

Make entry into Airtable Database for user does not have valid permission to book an event:

Manufacturer Classes Requested Users +			
Add fields Filter Group Sort Color Share and sync			
<input type="checkbox"/>	Name	Email	Status
1	Omer Yurdabakan	oyurdaba@pfw.edu	Todo
2	Rakesh Kumar Raushan	rausr01@pfw.edu	Todo
+			

Airtable automation which sends a notification to Admin for the user does not have valid permission to book an event:

Automations List

Unmanaged Event

Approved Event Confirmation

Gear Linker for Tickets

Event Canceled

Summer Bookings Weekly

Delete "Unmanaged Events"

Summer Event Approved Notice

Room Conversion (for all Calendars)

Request New User

Request New User

When a record enters a view, send an email

Configuration to send email notification when user is not having permission

Automation which send email notification to Admin

Properties

Action type: Send email

Labels: Description

Configuration: Action will run... Always

To: oyurdaba@pfw.edu

Subject: User does not exist

Message: Use markdown or HTML for rich text formatting: **bold**, *italics*, # Headings, * Bullets,
 for line breaks, and more

Email sent to admin for the user does not have valid permission to book an event:

User does not exist

AA

Airtable Automations <noreply+automations@airtableemail.com>

To: Omer Yurdabakan

Tue 11/28/2023 5:47 PM

You don't often get email from noreply+automations@airtableemail.com. [Learn why this is important](#)

CAUTION: This email originated from outside the university. DO NOT click links or open attachments unless you recognize the sender and know the content is safe.

This email was sent to notify you that below user does not exist in SMC users table.

Name

Rakesh Kumar Raushan

Email

rausr01@pfw.edu

Part 6 – Testing Reports

Test Case ID	Test Case	Pre-condition	Steps	Test Data	Expected Result	Actual Result	Status
Login_Test_01	Validate Sign In page loads	User should know the Room and Booking system URL	Access the URL of Room and Booking system on any browser	N/A	User should see a Sign In link on the home page of the application	User can see a Sign In link on the home page of the application after loading successfully	Pass
Login_Test_02	Validate Microsoft Azure AD login screen popup	Users click on sign in link	Access the URL of Room and Booking system on any browser and click on sign in link	N/A	User should see Microsoft Azure AD login screen popup	A Microsoft Azure AD login screen popped up	Pass
Login_Test_03	Test sign in functionality	Sign in page should load properly.	Give in the input details for sign up like Purdue Fort Wayne email if and password.	Purdue Fort Wayne email id (PFW email ID) and password	After giving correct email id and password, the sign in process should be successful and the user should be able to login.	User signed in successfully	Pass
Login_Test_04	Test sign-out functionality	Users should be logged in to the web-application.	Click on the sign-out button, after which the user should be logged out of the system. Verify that after signing out, the user is redirected to the login page.	NA	The user should be successfully logged out and the login session should be ended.	User logged out successfully and landed on the Sign in page	Pass
Login_Test_05	Validate failed to sign in with wrong password	Sign in page should load properly.	Give in the input details for sign up like Purdue Fort Wayne email if and wrong password.	NA	After giving correct email id and wrong password, the sign in process should be unsuccessful	User is not able to login with wrong password	Pass
Notification_Test_01	Book an Event - Test creating an Event (by user does not have valid permission)	Users should not have valid permission to create an event. Users should be logged in.	Users should follow the steps mentioned in Login steps.	Same as mentioned in login steps	Users who do not have valid permission should not have access to create an event. System should throw an error stating “Looks like user is not	User seeing an error message on UI page	Pass

					registered in the system”. System should send an email to the admin about the same.		
Notification_Test_02	Book an Event - Admin should receive an email	Users should not have valid permission to create an event. Users should be logged in.	Users should follow the steps mentioned in Login steps.	Same as mentioned in login steps	An email should be sent to admin about the user who does not have valid permission to create an event	Admin received the email about the user who does not have valid permission to create an event	Pass
Notification_Test_03	Book an Event – The email should contain the user details	Users should not have valid permission to create an event. Users should be logged in.	Users should follow the steps mentioned in Login steps.	Same as mentioned in login steps	The email should contain the user details. Name and email address	The email contains the user details name and email address	Pass
BookEvent_Test_01	Book an Event - Test creating an Event (by authorized user)	Users should have access to create an event. Users should be logged in.	Click on the book an event button and then click on create event which will take the user to user- details page. The user should fill in the details like session title, event type, intended use, room type, select studio room, select start date and end date and time. Click on check availability, if available, select gear and submit.	Event Details such as Session title, event type, intended use, room type, studio rooms selection, start-end date and time, and gear selection.	Authorized users should be able to create the event which should be successfully created and reflected in the calendar.	Successfully created the event and event data stored in the Airtable database	Pass
BookEvent_Test_02	Book an Event - Check availability	Users should give all inputs in the event details page.	After giving all the details in the event details page, click on the check availability button.	NA	The system should check the room and date-time availability and allow the user to successfully create an event.	System successfully created event and stored in the Airtable database	Pass

BookEvent_Test_03	Book an Event - Gear check-out	Users should give all inputs in the event details page.	After giving all the details in the event details page, the user should select the gear required for the session.	UI page should show all the available gear to book	Users should be able to successfully select the gear required for the session.	Users can see all the available gear on the screen to select	Pass
BookEvent_Test_04	Book an Event - Test if the available gears are displaying properly along with the pictures.	User should give all inputs in the event details page and reach the gear selection process.	After giving all the details in the event details page, the user should be able to see available gear with its images displayed.	Available gear should appear along with images of the gears.	Users should be able to successfully view the gears along with their images.	Users can be able to see the gear along with their image	Pass

Part 7 – Development Tools and Read Me Document

Development Tools:

- **Airtable:** Airtable is used to create databases that hold data for users to access, store, and organize.
- **NextAuth.js:** NextAuth is a completely secured and flexible authentication library designed to sync with any OAuth service.
- **Microsoft Authentication Azure AD:** Authenticating user against Azure Active Directory
- **JavaScript:** JavaScript is a full-stack programming language.
- **React.js and Next.js:** Popular frontend JavaScript framework.
- **Github:** Github is a platform that hosts code. It is accessible from anywhere and allows collaboration.
- **Vercel:** Vercel is a platform for developers that provides the tools, workflows, and infrastructure you need to build and deploy your web apps faster, without the need for additional configuration. One clicks Next.js app deployment.
- **Gear:** As used in this document, refers to audio and video equipment available for students and faculty at the Sweetwater Music Center.

Read Me:

Register an application with the Microsoft identity platform

- Follow the guide in the link: <https://learn.microsoft.com/en-us/entra/identity-platform/quickstart-register-app>
- Select “Accounts in this organizational directory only (Purdue University Fort Wayne only - Single tenant)” under “Supported account types”.
- Select “Web” from select a platform dropdown menu under Redirect URI section and insert “{your vercel website URL}/api/auth/callback/azure-ad” in the bar.

Supported account types

Who can use this application or access this API?

☒ Accounts in this organizational directory only (Purdue University Fort Wayne only - Single tenant)

☐ Accounts in any organizational directory (Any Microsoft Entra ID tenant - Multitenant)

☐ Accounts in any organizational directory (Any Microsoft Entra ID tenant - Multitenant) and personal Microsoft accounts (e.g. Skype, Xbox)

☐ Personal Microsoft accounts only

[Help me choose...](#)

Redirect URI (optional)

We'll return the authentication response to this URI after successfully authenticating the user. Providing this now is optional and it can be changed later, but a value is required for most authentication scenarios.

Web

e.g. https://example.com/auth

Register an app you're working on here. Integrate gallery apps and other apps from outside your organization by adding from [Enterprise](#)

By proceeding, you agree to the [Microsoft Platform Policies](#)

Register

- Create a “Client Secret” following the step included in the guide. This secret value can only be read once, make sure to copy it to a temporary location.

Providing Environment Variables in Vercel

- To run the code in local IDE create a file called “.env.local” in the root of the project directory and define all the variables below.
- To deploy the code in production below variable should be configured in Vercel

```
1 # This file contains the environment variables
2 NEXTAUTH_SECRET=
3 AZURE_AD_CLIENT_ID=
4 AZURE_AD_CLIENT_SECRET=
5 AZURE_AD_TENANT_ID=
6
7 NEXT_PUBLIC_AIRTABLE_KEY=
8 NEXT_PUBLIC_AIRTABLE_BASE_ID=
```

- In the app registration screen in Microsoft Azure platform, once you have created the app, you will be supplied with this information:

<h2>Essentials</h2>	
Display name	Client credentials
nextauth-test	0 certificate, 1 secret
Application (client) ID	Redirect URIs
963477db-d80c-4d99-a228-6b98e637ce26	1 web, 0 spa, 0 public client
Object ID	Application ID URI
c7c2f7ff-b4ef-4c95-9fe0-33c0e9a8d999	api//963477db-d80c-4d99-a228-6b98e637ce26
Directory (tenant) ID	Managed application in local directory
b7dc318e-8abb-4c84-9a6a-3ae9fff0999f	nextauth-test
Supported account types	
My organization only	

- Notice Application(client) ID and Directory (tenant) ID which corresponds to environment variables in the supplied sample env.sample file.
- Your AZURE_AD_CLIENT_SECRET is the value you created in “Client Secret” step.
- Following this guide: <https://vercel.com/docs/projects/environment-variables> create environment variables for deployment.
- *NEXT_PUBLIC_AIRTABLE_KEY*=
- *NEXT_PUBLIC_AIRTABLE_BASE_ID*=
- Variables will be obtained from Airtable application.
- Create Nextauth secret key: Run the command below in your command line. Alternatively, you can use this website to create one - <https://www.cryptool.org/en/cto/openssl> or go to this website <https://generate-secret.vercel.app/32> and copy the value.
- Copy the value generated and assign it to NEXTAUTH_SECRET value.
- `openssl rand -base64 32`
- Create NEXTAUTH_URL=https://yourwebsite.com where the URL is your website link supplied from Vercel.

Code Repository:

Source Code: The source code for the website can be found in the GitHub repository PFW-Music/smcWebsite.

This is a Next.js project bootstrapped with create-next-app. The working branch name is “ACS560Fall2023” and the GitHub link is <https://github.com/PFW-Music/smcWebsite/tree/ACS560Fall2023>

Getting Started

First, run the development server:

Type the following into the terminal:

```
npm run dev
```

Open <http://localhost:3000/> with your browser to see the result.

You can start editing the page by modifying pages/index.js. The page auto-updates as you edit the file.

API routes can be accessed on <http://localhost:3000/api/hello>. This endpoint can be edited in pages/api/hello.js.

The pages/api directory is mapped to /api/*. Files in this directory are treated as API routes instead of React pages.

This project uses next/font to automatically optimize, and load Inter, a custom Google Font.

Notes:

The ‘.env.local’ file is not on GitHub to prevent the leakage of user data. For more information on creating this file, please refer to the above section of Read me. It must be changed in the ‘.env.local’ file in the root of the project directory. If this change needs to be deployed, the environment variable on Vercel must also be changed.

Part 8 – Code Design and Standards

Documentation

- Comments - pieces of code that are longer than 5 lines and have functionality that is not self-explanatory should have a comment above explaining the purpose and logic of the code.

Programming Practices

- Naming- variables, constants, functions, classes, and components should be given descriptive names that adequately describe their purpose and aid in code readability

Design

- Code Reuse - codebase should be containerized into classes, components, and functions for the purpose of reusing code. This practice helps reduce bugs and keep the codebase adaptive to change.
- Encapsulation - wherever beneficial, code should be separated into classes and components with proper use of data hiding and modularization.

Design Patterns

- Component-Based Architecture - Codebase should be centered around components that can be used repeatedly. Breaking down User Interface components should lead to a manageable codebase.
- Render Props - Components should provide render functions to their children to allow for better code reuse.
- Context API - Data should be shared between components without needing to pass data through other component levels.

Cohesion and Coupling

- Cohesion - Components and methods should be closely related and focus on a single task or Responsibility.
- Coupling - Components and modules should be independent of one another. Code is modular and easy to update without affecting the codebase.

Part 9 – Member Contributions

Team Member	Requirement	Design Model	Documentation	Implementation	Testing	Presentation
Omer	✓	✓	✓	✓	✓	✓
Bravya	✓	✓	✓		✓	✓
Reene	✓	✓	✓		✓	✓
Rakesh	✓	✓	✓	✓	✓	✓

Part 10 – References

- <https://azure.microsoft.com/>
- <https://next-auth.js.org/>
- <https://vercel.com>
- <https://tailwindcss.com>
- <https://nextui.org>
- <https://react.dev/reference/react>
- <https://airtable.com>
- <https://github.com>