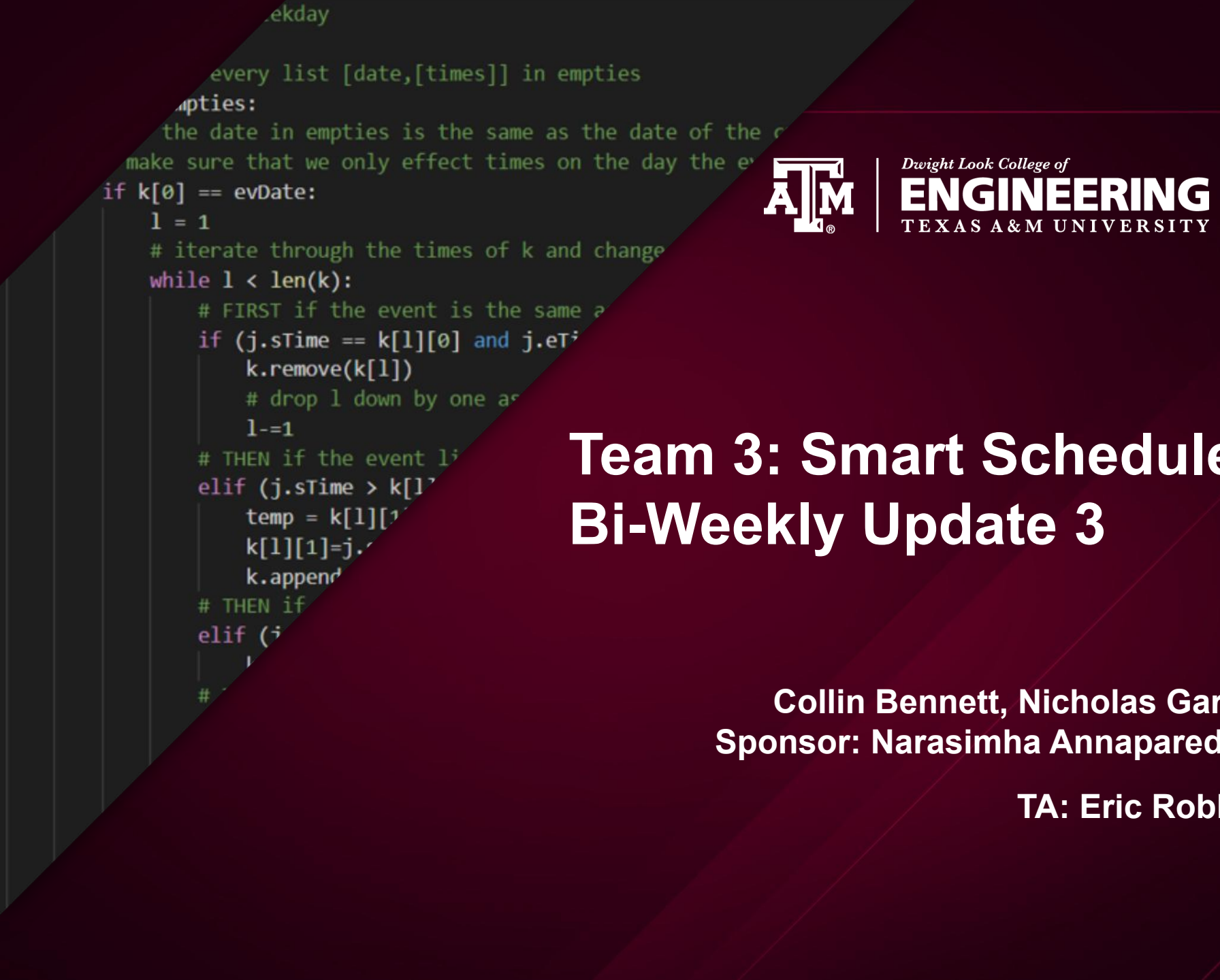


Dwight Look College of ENGINEERING
TEXAS A&M UNIVERSITY

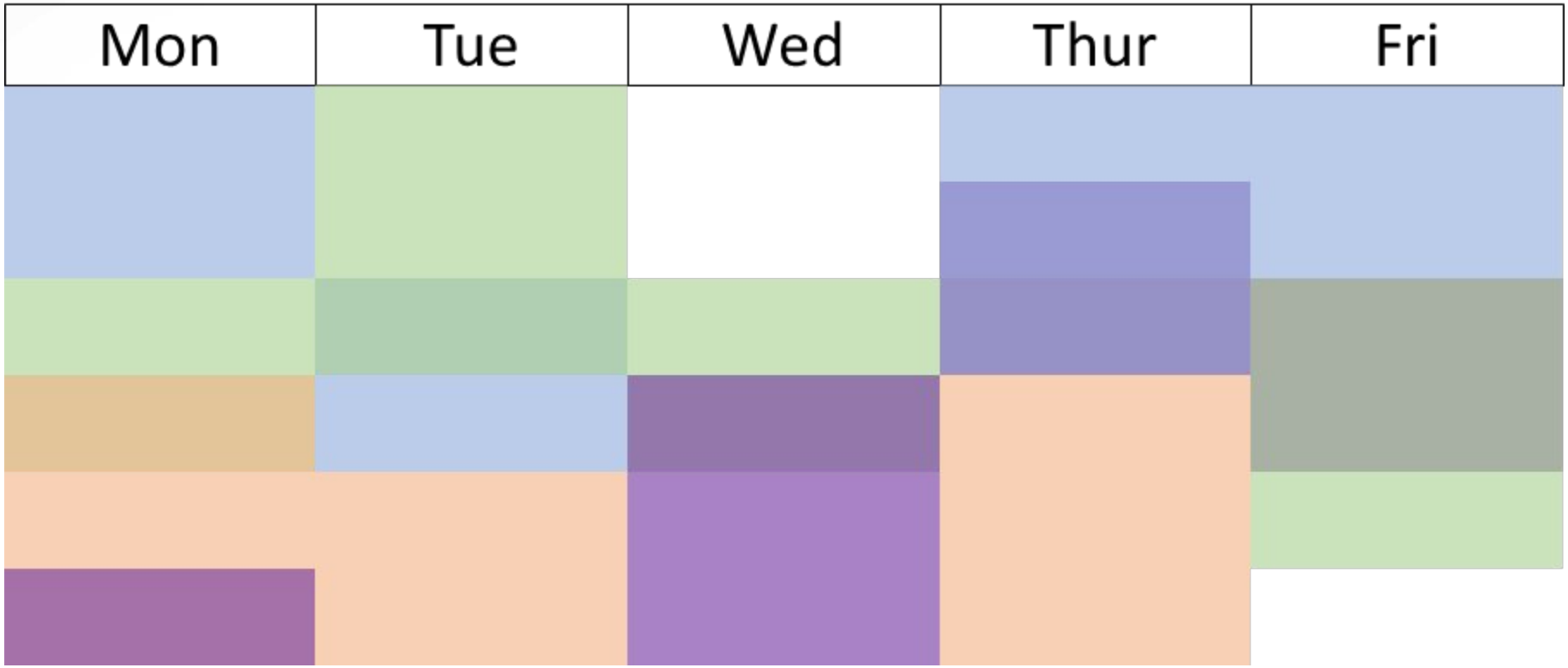
Team 3: Smart Scheduler
Bi-Weekly Update 3

Collin Bennett, Nicholas Garde
Sponsor: Narasimha Annapareddy

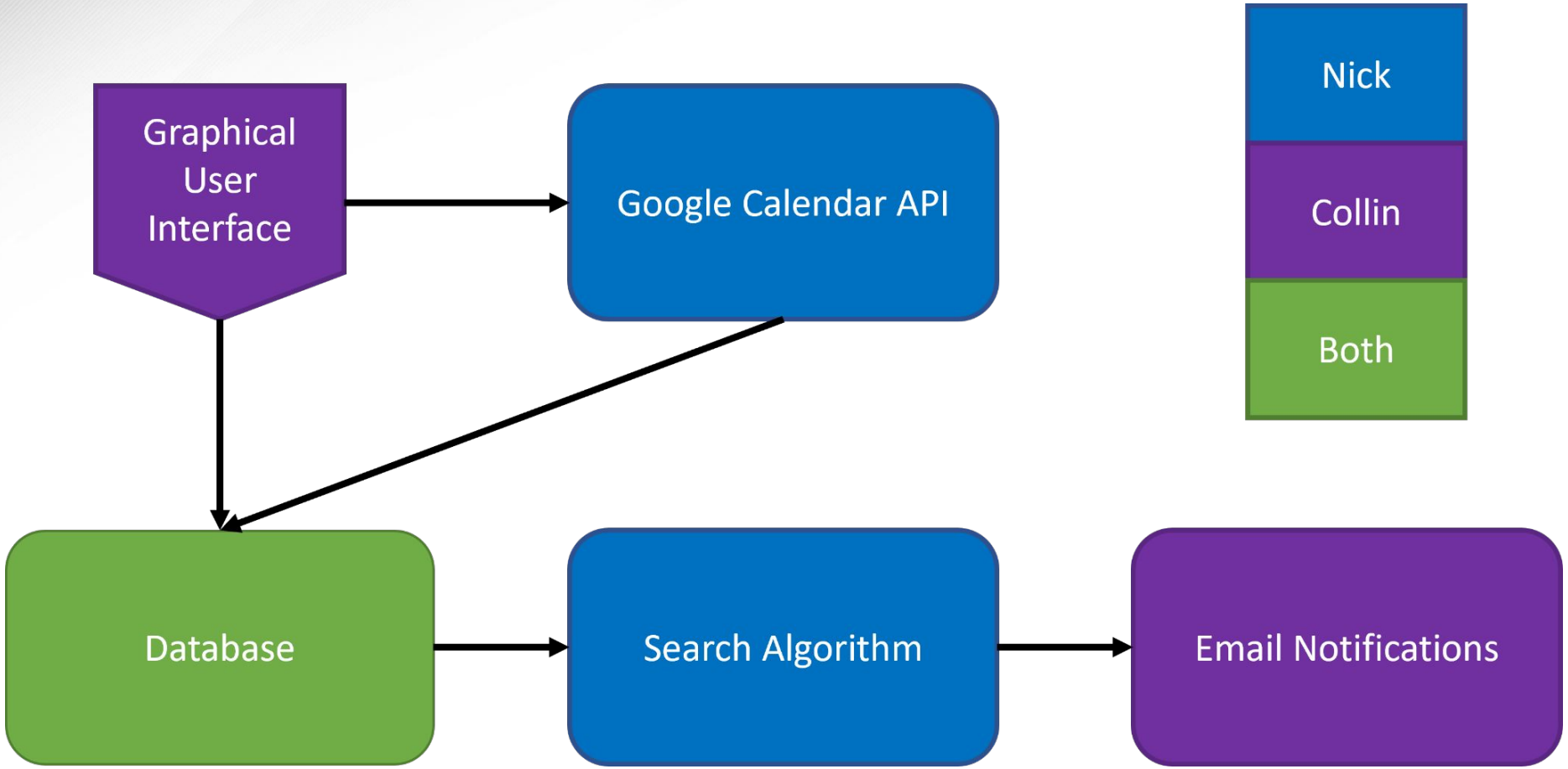
TA: Eric Robles

Introduction

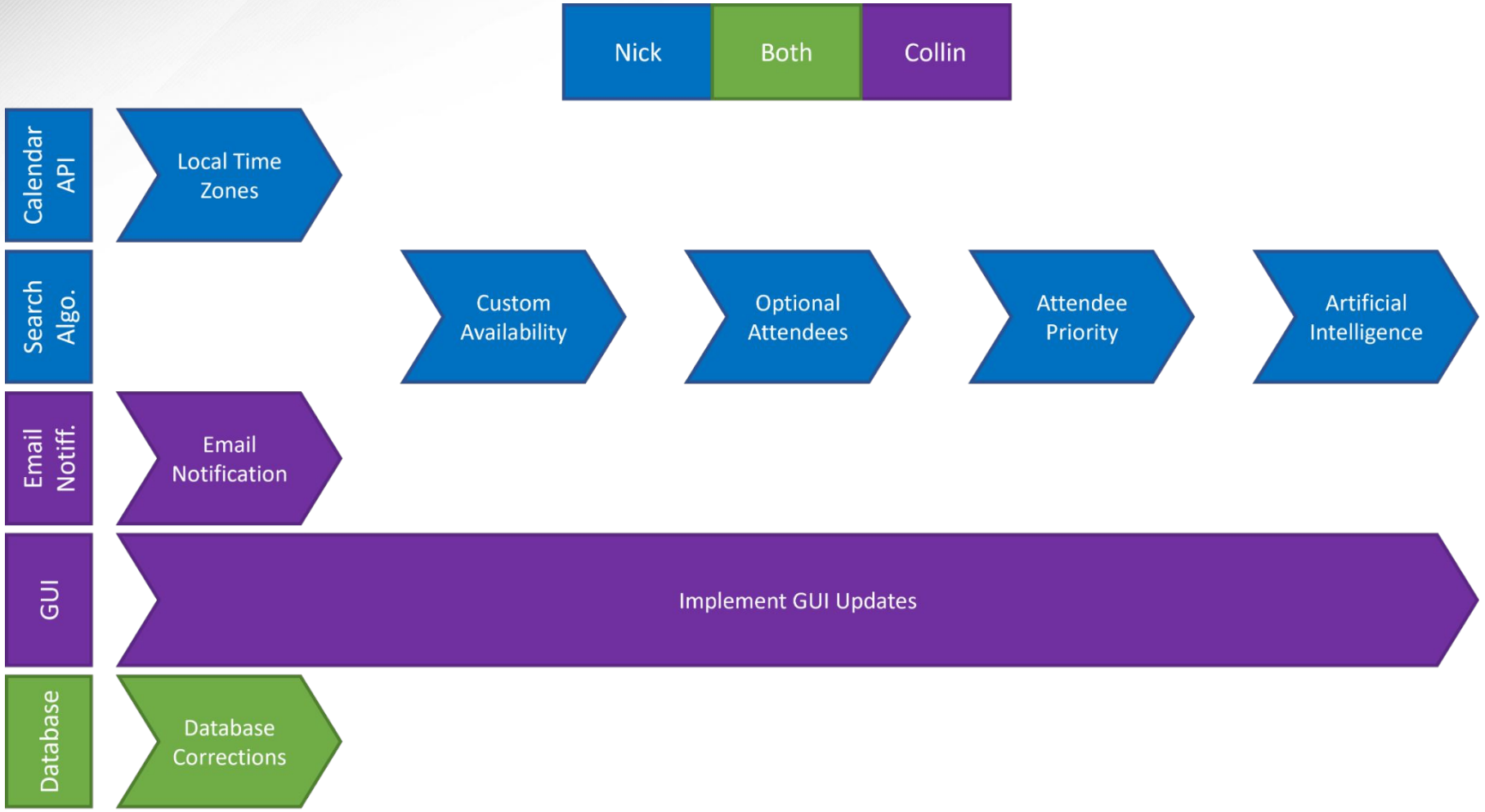
- Scheduling meetings can be very difficult, so we are simplifying the process.



Project/Subsystem Overview



Project Timeline



Graphical User Interface

Owner: Collin

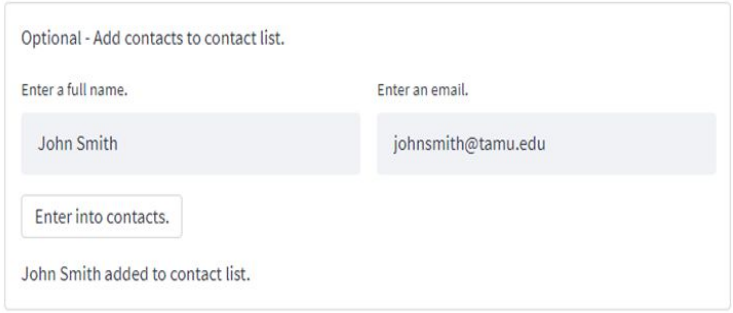
Since last presentation: 10 hours of effort	Ongoing progress/problems and plans until the next presentation
- Fixed a time display bug - Corrected page loading errors	- Implement optional/required attendees - Delete RSVP page (email)

Graphical User Interface

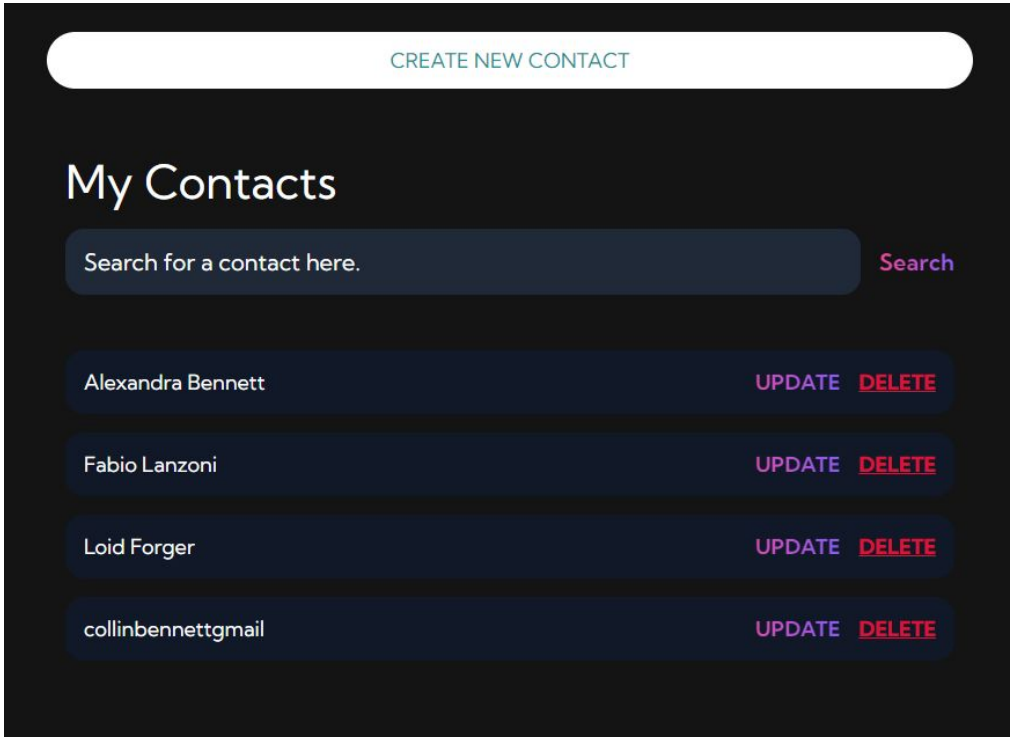
Owner: Collin

- Planning to add the ability to import contacts from Google
- Making updating and deleting contacts simpler

Before



After



Database

Owner: Both

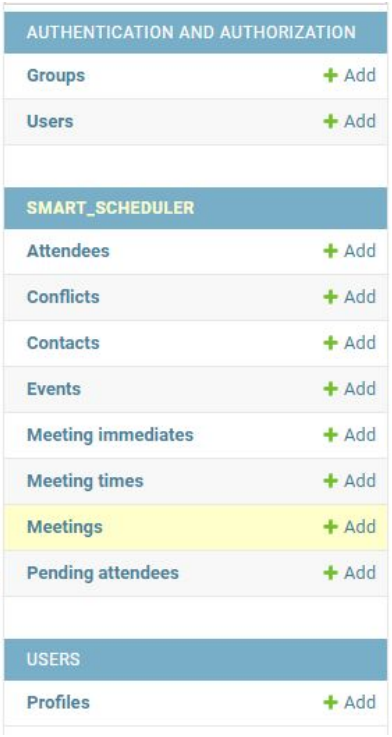
Since last presentation: 5 hrs of effort	Ongoing progress/problems and plans until the next presentation
The SQLite database has been ported to PostgreSQL and hosted on AWS Slight structure changes	Slight modifications/refinements will be made as needed

Database

Owner: Both

Before

After



Search Algorithm

Owner: Nick

Since last presentation: 10 hrs of effort	Ongoing progress/problems and plans until the next presentation
New data structure to be compatible with local time zones and user preferences	Take into account optional attendees and priority of attendees

Search Algorithm

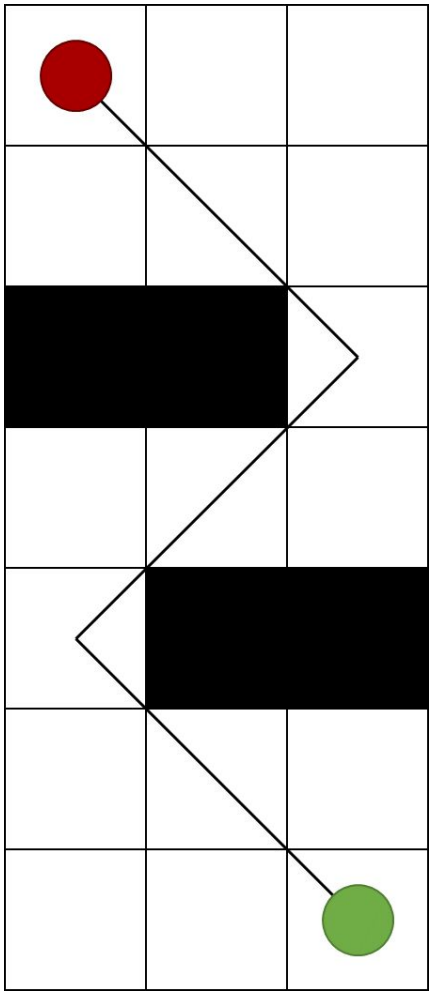
Owner: Nick

Current Design

- Places events onto calendar data structure
- Searches by # of conflicts
- Take into account number of participants in each meeting
- 24hr data structure

Planned Design

- Optional Attendees
- Take into account priority of participants



Calendar API

Owner: Nick

Since last presentation: 5 hrs of effort	Ongoing progress/problems and plans until the next presentation
Pull user time zone	Optional “Remember Me?”

Calendar API

Owner: Nick

Current Design

- Pulls user events from their calendar
- Pushes created events back onto their calendar
- Retrieve user time zones

Planned Design

- Option to skip authentication

Email Notifications

Owner: Collin

Since last presentation: 5 hours of effort	Ongoing progress/problems and plans until the next presentation
- Email notifications have been integrated into the app as a Github branch	Testing and refinement of email notifications

Execution Plan

				Completed	In Progress	Not Started
Subsystem	August 31st	September 14th	September 28th	October 12th	October 26th	November 7th
Search Algorithm (Nicholas)	Review changes to algorithm and receive sponsor feedback	Implement timezone functionality	Implement custom start and end times, valid days	Researching better machine learning and lowest-cost solutions	Iron out any potential oddities in the search algorithm	Have search algorithm completely done
Database (Both)	Move from firebase to relational database	Simplify database, getting rid of unnecessary tables	Have database ported from SQLite to PostgreSQL and hosted.	Continue work on other subsystems	Check database for correctness	Database is done, hosted and complete
GUI (Collin)	Django website bug fixes and refinements	Have some form fields optional to speed up the process	Have required and optional attendee functionality integrated	Get additional user feedback on the website style	Implement changes based on the user feedback	GUI is finalized and styled
Email Notification (Collin)	Rework email to SMTP from API functionality	Email is properly styled	Email is fully functional on a basic level	Email is not being sent to user's spam folder	Validate and test email	Email is sending without error and styled
Google APIs (Nick)	Review and research Google APIs in Django	Integrate timezone functionality with Google API	Integrate Google contacts imports	Begin testing correctness of API calls	Finalize testing of API calls	Google APIs are finalized

Validation Plan

Completed	In Progress	Not Started

Task	Specification	Method	Result	Owner
Google Calendar - Pull Contacts	Test Fetching Accuracy (No errors)	Using Google's OAuth2.0 to request contact data from the user from their email	Pending	Collin Bennett
Google Calendar - Insert Event Times	Google API is properly inserting correct event times	Ensure that calendar events can be added to specified user calendars without error. Specifically, ensuring that users in different time zones are being sent the proper information.	Pending	Nicholas Garde
Database - User Authentication	Database is storing user account details	Ensure PostgreSQL solution is working properly	Complete	Collin Bennett
Database - Event Parameters	Ensure event parameters are added properly	Add several events of varying nature to Database from the GUI, ensuring that they are properly added each time by comparing the inputted values to those stored in Database.	Complete	Collin Bennett

Validation Plan (Continued)

Task	Specification	Method	Result	Owner
Database - User Events	Ensure User events are properly added	Pushing each user's calendar to database to be stored until the data is needed for the search	Success	Collin Bennett
Search Algorithm - Time Zones	Ensure User events have proper times	Ensure that users in different timezones are correctly accounted for by the algorithm	Pending	Nicholas Garde
Search Algorithm - Additional Parameters	Valid times and weekdays	Ensure that additional parameters are correctly accounted for by the algorithm	Pending	Nicholas Garde
Email API Correctness	Testing of Email Notification Scenarios	Ensure the email link clicks make the proper writes to the database and move efficiently.	Pending	Collin Bennett
GUI - Ease of Navigation	Exploring GUI navigation ease-of-use	Ensure the GUI navigation works properly, nothing is out of place, and correct error messages are displayed.	Pending	Collin Bennett
Bug fixing	Attempting to break and Find Errors in Our Program	Ensure that the program works under all conditions by scheduling numerous programs under varying conditions and checking that they are scheduling correctly.	Pending	Collin Bennett and Nicholas Garde

Questions?