

Project Biography – Calendar Assistant
Coding the Law Fall 2023
Adrianne Downey

Link to Google Colab:

https://colab.research.google.com/drive/1CDj3RkdW2FruFK1Jf3_cyvXIIIGFUJGh#scrollTo=iDkKgxBwibYv

The OpenAI key listed has since disabled. Please contact me if another is needed for grading purposes.

Link to my “syllabus” folder:

https://drive.google.com/drive/folders/1zgYpVD-goXfl6yDH8P1lpcv3kkvZzH7v?usp=drive_link

Link to my GitHub repository for this code:

https://github.com/adriannedowney/ctl/blob/main/ad_calendar_assistant.py

Please see the end of this document for my partner letter.

Framing

Following the rise of swapping from paper to technology, students require an effective way to keep their tasks on their radar to complete them in a timely manner. Therefore, I need to design a product that parses due dates and assignments from different syllabi to help Suffolk Law students monitor their assignments through their Google Calendars with the assistance of Artificial Intelligence.

While all are welcome to use this product, the relevant stakeholders are Suffolk students who have struggled to remain consistent with traditional assignment-tracking methods. These users often find that they start strongly with their chosen method, such as choosing an appealing academic planner, but then it becomes a waste of money because they forget to update it throughout the year. Suffolk students are specific stakeholders here because I am my own partner and am currently in the process of completing my third year at Suffolk Law.

Adrianne is a third-year law student who has struggled to commit to an assignment-tracking method since her first year of her undergraduate degree. She has attempted using sticky notes with specific tasks, a monthly calendar to keep her big-picture due dates in mind, a white board that she updated at the end of each class for every single class she was in during a given semester, and a planner that met her desired needs, which included larger weekly-based blocks and smaller monthly-based calendars. She had the most success with her white board, but that set up was only conducive to her living environment. That method proved to not be feasible upon moving home due to COVID-19. She hopes to use her Google Calendar to remind her of her assignments, especially because she can combine that calendar with her calendar for Moot Court Honor Board, which details all of the cite-checking and auditing rounds that she supervises.

Research

Currently, the market for electronically generated syllabus assistants is limited. At the moment, Syllabuddy is the only available option. However, Syllabuddy was created to focus on Apple Calendar and Google Calendar. In addition, it is not clear whether Syllabuddy includes the

use of Artificial Intelligence. As a result, users are limited to the traditional methods of assignment-tracking if they are not willing to spend the time to enter each of their assignments as Outlook events on their own. These methods include using a classic planner, writing to-do lists on sticky notes or other individual pieces of paper, using a white board and desk calendar as mentioned above, reopening the syllabus before each class meeting, simply choosing to wing it based on what the professor said at the end of the last session, and choosing to not do anything at all.

After discussing this project idea with my peers, we reached the conclusion that this is a product students would benefit immensely from. Many of these discussions were had in the Moot Court Honor Board suite, while students were updating their to-do lists and checking their class syllabi immediately before going into class for the evening. They discussed how they wished they had a more feasible solution to take the burden of planners off their minds. It is much easier to have a software sort a syllabus into reminders than handwrite individual assignments on a daily basis. They also talked about they would like it to connect with their personal emails. I have also discussed this topic at length with Caroline McCarty and Jessica Vandervort, as they have also gone through similar struggles when it came to finding the right method to track their assignments.

Ideation & Prototyping

There are several different approaches I could take to resolve this issue. First, I could rely heavily on Regular Expressions to refine my data inputs and decrease the demand put on the OpenAPI platform. As part of this method, I would pull out the date, time, and assignments for each class meeting. However, this might not be the most effective route, due to the power of artificial intelligence. This would possibly create more exact responses to the necessary sections of the calendar invite, but it may take extensive coding that may miss certain ideas along the way.

Second, I could try to even the balance on the Regular Expressions and OpenAI by having the date and time be extracted by the Regular Expressions and rely on the artificial intelligence to extract the assignments and the class name. This might be more effective, as classes do not have the same styles of assignments and it would be complicated to pull out course names by Regular Expressions due to the diversity in course titles. Regarding course assignments, some classes assign page numbers, while others just do page names or statutes. It would be too timely to seek a Regular Expression to extract this data when artificial intelligence could do it at a quicker frequency.

Third, I could have my code produce a CSV file and then create code that would convert it into an ICS file. This would be useful to provide users with an excel sheet along with an ICS file of the due dates. This would be great for someone who wishes to have the excel sheet along with the ICS files to track their assignments. However, this might be better served by jumping straight to an ICS file, as the excel file may likely be the equivalent of a physical planner or to-do list.

Fourth, I can produce an ICS file directly from the information parsed by OpenAI. Once I have the necessary pieces of information (the class name, their due dates, and the assignments), I can create a prompt that tells the software to put the information into a specific format, including a date start, date end, and a summary. Upon completion of the ICS file, I can then deposit into Google Calendar. This differs from the three methods discussed above because in those

solutions, I would produce a CSV file and then convert it to an ICS file, instead of skipping straight to the final product.

User Testing

First, I ran this test with a few different syllabi, as my own user. I found that the product worked well when it came to the different formats, even if the dates were not structured in the same manner. For example, in one of the tests I ran, I used a syllabus from an Environmental Law class. I was not sure how it would go, as the dates were entirely numerical and the assignments did not follow a boxed format, unlike the syllabus with which I prepared the code. However, going through the entire process after creating the code made it go seamlessly. Everything went so quickly and was accurate. I was very happy with my final outcome, but would love if I would be able to implement the necessary line breaks to also implement the same calendar invites in Outlook. Please see the attached emails below regarding my recorded feedback along the way.

In addition, I also had Jessica Vandervort and Caroline McCarty test this product out. It was a great way for me to refine my instructions, as I learned what did and did not make sense as they were working through running my code. In the end, we had a very successful output – they were both able to produce Calendar invites that met their exact needs. I have consent to share any emails pertaining to their tests. Jessica noted that she was really excited for this product and that she intends to make use of it in the Spring semester. This is great because although I am my own user, she still fits within the relevant stakeholders. She uses Google Calendar for her personal commitments as well, so this was a great way for her to incorporate both her school assignments with the rest of her activities and commitments. Caroline noted the broad use this tool can have as I continue to expand its reach. In addition, she complemented the instructions I gave as she worked through the code to confirm that she was following the steps as required. Please see their feedback emails attached following our meeting at the end of this document before the partner letter.

Refinement

While designing this solution, I considered multiple different styles and configurations to achieve the most productive outcome. First, I considered using Regular Expressions to extract the dates, times, and assignments, which would then be refined by OpenAI. Upon further research, that did not seem to be the most effective use of the code. While Regular Expressions would have worked for the assignments, there was more that could have been done with OpenAI. Therefore, in my second rendition, I used Regular Expressions for just the date and time, while I allowed OpenAI to extract the assignments and class names, and solidify the dates and times. This seemed to lead to a little more success, but it was still not perfect. First, it pushed the limits of the OpenAI packages that were available for requests per minute because the way my code was structured, it required four responses per entry instead of three. Second, the code was clunky when considering separate entries for each column. In my next version, I condensed the date and time OpenAI calls into one call to meet the request requirements. This was successful. In addition, in this version, I had the output of those OpenAI calls form a CSV file. Although I was not able to separate the content, I was able to import ics and the code pulled out the necessary material from the CSV file to form the information required for the ICS file. At that point, I mainly needed to determine how to produce a downloadable file to allow my user to download the file and add it to their Outlook calendar. In my next version, I considered condensing my

OpenAI calls even further to include the assignment with the date and Regular Expressions to pull the time. However, this was not fully effective, as the Regular Expressions did not make it into the final code. In addition, this did not properly parse the dates.

As part of my final version, I used Regular Expressions to complete the proper formation of my dates produced by the LLM prompts. In addition, instead of producing a CSV file and taking the extra time and OpenAI credits to convert the CSV file to an ICS file, I prompted the code to only produce an ICS file. Additionally, I made the decision to swap from Outlook to Google Calendar, as Google Calendar is still useful to my user (myself), as I use it for other school and home responsibilities. Google Calendar is more user-friendly and does not have a character limit per line.

Intro Pitch

I am really passionate about finding a solution for this! I am definitely a menace when it comes to maintaining any sort of physical planner, so I am excited to see what I can do as my own partner to keep myself more organized and have my assignments at a wider glance. This will be a better solution than checking my syllabi before each class as well, as I can integrate it with my other responsibilities that are available to review in the palm of my hand. Please see the slides from my presentation at the end of this biography.

Complexity/R robustness

To complete this product, I used a combination of Regular Expressions and data extraction through a language learning model on Google Colab. The coding within Google Colab was in Python. OpenAI extracted the course name, the due date of each assignment, and the actual content assigned for each class meeting. I used Regular Expressions to remove the dashes initially present in the API call so they meet the required format for calendar invites. I also used Regular Expressions to extract any times present in each syllabus, but that ultimately was not used to produce the final product.

Impact & Efficiencies

This process significantly speeds up the process of logging assignments, both in terms of physical and electronic recording, because the user does not need to take the time to manually enter each assignment into a planner or its own calendar invite. In addition, the user will be able to keep their assignments on their radar more effectively, because students frequently check their emails and get calendar reminders for events. This saves the student from having to revisit their planner or even the class's syllabus to determine what they need to prepare for each class meeting.

Fit/Completeness

Although this project is limited by the available OpenAI package based on my current usage, it is still able to complete the task at hand. It is able to parse different styles of syllabi that fit within the size requirements for OpenAI. In addition, I was able to have it produce a functional event in Google Calendar. Although this was not my initial goal, it still meets my purpose, as I am able to integrate my assignments with my other non-academic commitments. Each event includes the homework assignment due on a given day and the event is set to last the duration of the day.

Documentation

When using this version of the Calendar Assistant, users must:

1. Enter the syllabus into their favorite OCR platform to receive a “.txt” file. A useful version is: <https://pdfcandy.com/pdf-ocr.html>. On this website, please make sure you are using the PDF OCR feature.
2. Take a brief glance over the syllabus before inserting it into the designated Google Drive folder. For now, users must remove everything **but** the class name and the actual schedule of assignments listed. As we expand, this will no longer be an issue.
3. Users must mount their Google drives to the code by running the cell starting with [100]. Please be sure to have a file named “syllabus” inside your Google Drive so you may properly access the files you have prepared.
4. Users must then give a pathway to a Google folder that will contain the syllabus to be parsed. Users would get this information by listing the name of the folder where “syllabus” is in the file path in the cell starting with [105]. Please run the cell that starts with [105].
5. Users must then run the cell starting with “[106] sample = 1.” This cell actually parses through the text document.
 - a. If you are looking to rerun this cell, be sure to complete step 4 again as well to produce a fresh output.
6. Users must then run the cell “[107] temp_calendar” to populate the variable.
7. Then, users will run the cell labelled “[108].” This cell defines functions that form the information in the actual calendar invite.
8. Next, users will run the cell beginning with “ics_file = “““BEGIN:VCALENDAR.” This cell will form the text required for the calendar invite and download the produced file to the user’s computer, rather than a virtual drive.
9. Upon saving this ICS file, users will go into their Google Calendar and click on the gear.
10. Then, users will select “Settings.” Once there, users will select “Import & Export” from the lefthand column.
11. Finally, users will select the file they wish to upload and click “Import.”
12. Then users will be able to see the events on their Google Calendars for their corresponding times.

Real World Viability

This platform is ready to go as is. Users will be able to use this platform as long as they have access to a Google account that allows them access to a drive, which is free. Google is a very common workspace platform, so this should not be an issue for users. If they have yet to connect an email to make an account, it is very simple to do so. The other required steps are very simple when followed as instructed in the Documentation section. The required steps are necessary simply to make this a personal interaction. If users wished to parse the file already provided, they may do so as well.

Currently, the use of this platform is limited to Google Calendar, as there are stricter content-length requirements for different subject lines for Outlook calendar events. To maximize the utility of this product, it is best used on Google Calendar.

My testers have also agreed that this is a significantly better approach to managing their growing workloads than by using a paper planner. As we all have work duties outside of school, we cannot keep our paper planners on us all the time and do not want to clutter our computer

desktops with sticky notes. Consequently, this is the next best thing we could use to achieve our goals of tracking our assignments virtually.

Sustainability

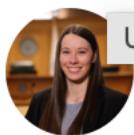
Following the release of this platform, I plan to maintain its connection to my API key so users may continue to take advantage of OpenAI in this process. I also plan to keep this page bookmarked to utilize it in the future along with sharing it with others who might be interested in using the platform.

I also hope to make updates to improve this product. First, I would like to create a mechanism that would allow users to OCR the documents in Google Colab instead of needing to pursue outside assistance for that process. In addition, I would also like to see if I can allow invites to be created for classes that only assign material by weeks, such as reading a list of cases by Week 9 of the course. This would require users to include the start date of the course and noting on which days of the week the class meets. Then, the material could be produced on both meeting days, although there was not a date specifically provided on the syllabus. I would also like to allow it to parse more than one document at a time. Also, I would like to work on the line splitting requirements to eventually move this over to outlook as well.

As my own partner, I am ready to implement this user interface as-is, especially in consideration of my plans of my consideration for expansion. This is something that will take away the need to remember to enter each of my assignments manually, whether that be in a planner or on a virtual interface. It is also more effective to help me keep all of my priorities in mind, as I have my Google Calendar for my responsibilities as the Executive Editor of the Moot Court Honor Board on my phone as well. That way, I can balance my responsibilities between journal and school while working off of my phone. Doing this whole process from top to bottom took all of three minutes, so I am excited for this improvement with my daily tasks.

Feedback from User Testing

Emails and other recorded feedback from discussions with Adrienne have been provided on the following pages, along with her partner letter.



User Feedback

Adrienne Marie Downey <a...

Yesterday at 9:54 PM

To: Adrienne Marie Downey

Hi,

Is there a way we can streamline the output from the code straight into an ICS file, rather than having to jump from the CSV file to the ICS file? It might reduce the stress on the processing system to get there.

It seems like we are getting closer – you have the correct information pulled, but we need to make sure it is getting sorted into its own event by date. It would be great if we could have the times assignments might be due if applicable, but I think we still achieve our purpose if we have an all-day event. The student will know when that class's meeting time is, so if the assignment says it is due before class, the student will know what that means.

Best,
Adrienne

User Feedback



🕒 Adrianne Marie Downey <adow...

Today at 1:01PM

To: 🕒 Adrianne Marie Downey

Hi there,

Thank you for sending your output along! It seems like we are getting to where we need to be.

Is there a way to get this material into an actual calendar event? I think that's our final step, and then we should be on our way!

Thanks for taking this on!

Best,
Adrianne

User Feedback



✓ **Adrianne Marie Downey <ado...**

Today at 11:59 PM

To: ✓ **Adrianne Marie Downey**

Hi,

This is great!

However, it looks like I am having some trouble with getting these files into Outlook. I also use Google Calendar for a lot of my other responsibilities, so I think the purpose will be met even if we switch platforms.

The fewer the cells, the better!

Thank you,
Adrianne

Final User Feedback



✓ Adrianne Marie Downey <ado...

Today at 12:15 AM

To: ✓ Adrianne Marie Downey

Hi Adrianne,

Great job on this final product! I know this wasn't the switch we had in mind, but I still think this completes what I needed it to do.

I am really excited to implement this – I think this will really capture what we need it to! I also can't wait to see how this improves in the future.

Thanks for all the hard work!

Best,
Adrianne

Additional Notes from Adrianne:

- She originally thought Regular Expressions for primarily extracting the date and time each assignment is due would be effective. However, she was afraid that by using Regular Expressions to pull out the date and time, we would end up extracting other numbers that fit the format of the date and time but are instead statutory codes or fractions. This was seen with the UCC codes, as having Regular Expressions pull dates that contain dashes flagged every entry of the UCC.
- When forming the ICS invites, she found that each data entry needed to have its own UID. Although we got one to randomly generate for each call, each different event also needs their own UID for every event that is listed.
- She expressed that we may be wasting resources by combining our parsed data into a CSV file and then converting it into an ICS file, so she would like to deviate from our initial plan and skip a step.

Collab Project

😊 ← ↲ ↳



⌚ Jessica Vandervort <jvandervo...

Today at 11:51PM

To: ✓ Adrienne Marie Downey

Hi Adrienne!

Great job on the calendar app project! It's so helpful and I'll definitely make use of it in the spring semester!

Thank you for sharing it with us!

Best,
Jessica

User Feedback for AD

😊 ← ↲ →



✓ Caroline McCarty <cmccarty3...

Today at 12:54 AM

To: ✓ Adrienne Marie Downey

Hi Adrienne!

First, awesome project idea that will be so useful not just for law students, but anyone that uses syllabi.

Your step-by-step instruction from creating a "syllabus" folder to OCR-ing and removing "fluff" at the beginning of the text file were all really helpful.

Although I encountered an error, you were quick to know that the text file for the syllabus I used jumbled the month and day, to no fault of your own. Once that was corrected, it was smooth sailing from there.

Your project has the ability to help so many people, especially those that need to spend less time on inputting due dates on their calendar (and more time on studying!). Even though I do not wish to enroll in school again after graduation, I would use your project if I had to do it all over again!

Thank you for letting me be a test user. Great job!

Best,

Caroline McCarty

Suffolk University Law School J.D. Candidate, 2024

Adrianne Downey
120 Tremont St
Boston, MA 02108

December 15, 2023

Dear Adrianne,

Thank you so much for the hard work you put into the Calendar Assistant! It took a lot of trials and tribulations along the way, but I am glad that we figured out a solution that will meet my needs. I am excited to not only implement this beginning version of the code, but to also be involved in future versions of this code.

In the future, I would love to see if this code could produce an ICS file that would be read by Outlook. Although I manage a lot of my personal life through my Google Calendar, I would love to be able to combine my assignments on my Outlook Calendar, which also contains my work meetings and work-related commitments. I know this is more complex, so that is why I recommended we switch the focus of our efforts so we could hit the ground running. Because Google is such a popular platform, I have no doubt that others who wish to use this resource will have the necessary materials to.

I am looking forward to using this platform in the Spring to parse through my course requirements. I am very excited to hear that you plan to keep this code active and improve upon it during the months to come. I am also grateful that you are continuing to allow access to OpenAI through this code. I will be in an extra class next semester compared to the last three semesters, so I will need all the resources I can to keep myself on track. This is especially so, as my heavy course load days are on Mondays, and then I will be right back to school bright and early on Tuesdays. In addition, this will be useful for when I am commuting in, as it is a lot easier to review which assignments are due on a given day on the train from common platform, instead of lugging around a paper planner with the rest of my notebooks. Sticky notes never really worked well for me either, as I would manage to lose them before I could check everything off my list.

I wish you well as you complete your final exams for the Fall semester! Please do not hesitate to reach out if you would like more feedback on future versions of this platform. I would love to continue to contribute to the development of this program.

Thank you,
Adrianne Downey

Calendar Assistant

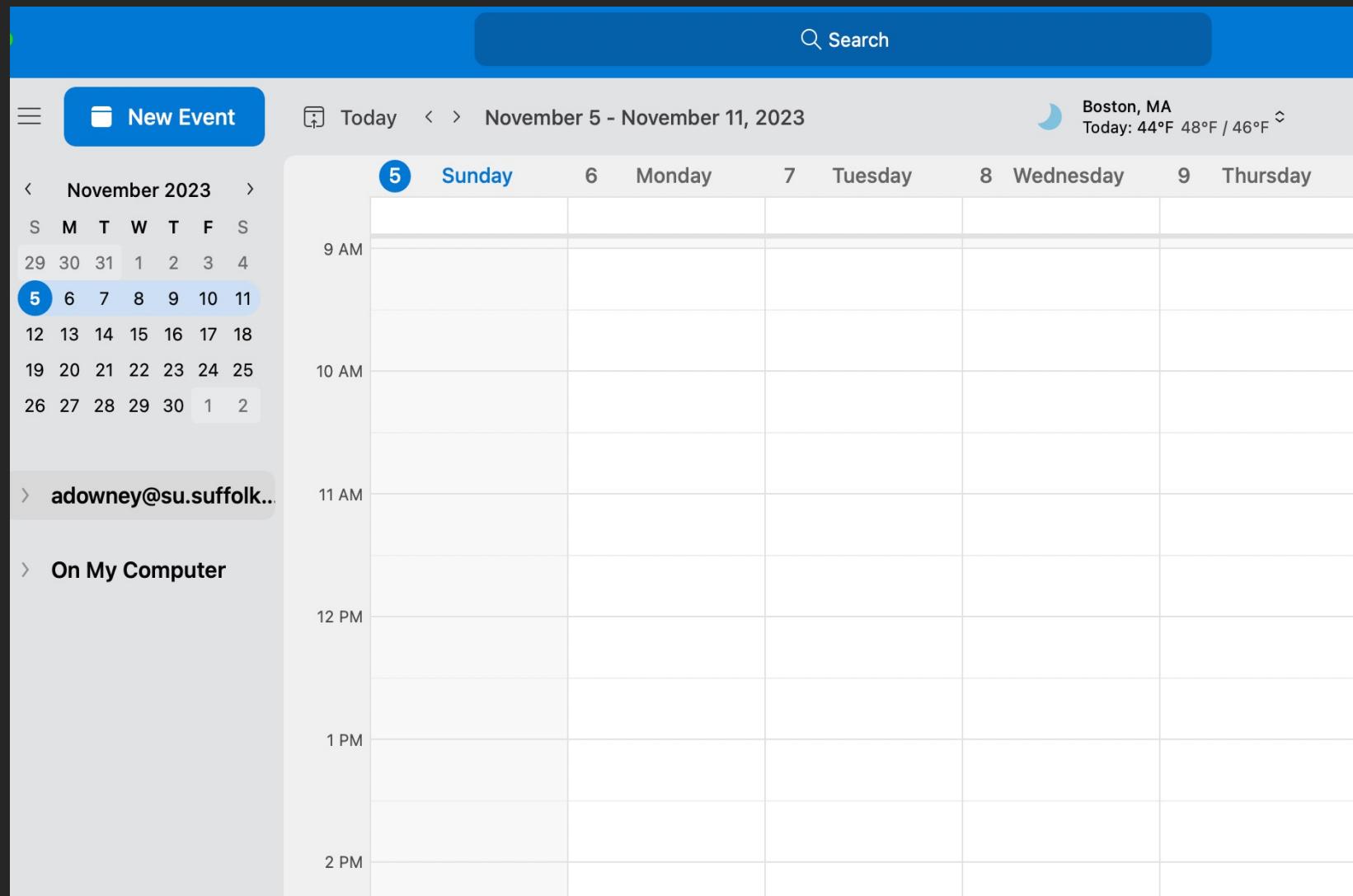
ADRIANNE DOWNEY

Agenda

1. Problem Statement
2. Options Currently Available
3. Relevant Stakeholders/Users
4. Partner Introduction
5. Sketch of a Solution

Problem Statement

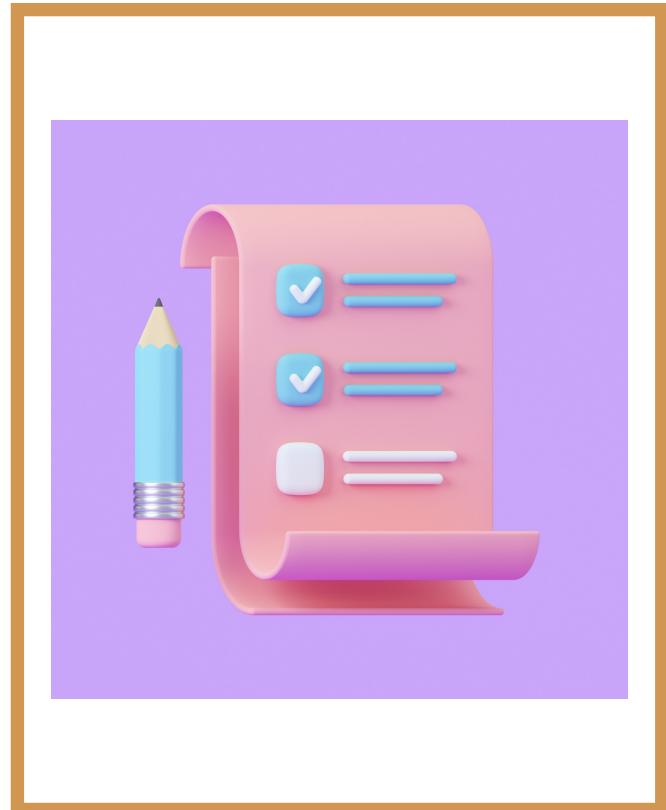
I need to design a product that parses due dates and assignments from different syllabi to help Suffolk Law students monitor their assignments through their Outlook calendars.



Options Currently Available

A screenshot of a calendar application interface. At the top, it shows a red circular icon with a white dot and the text "Calendar (adowney@su.suffolk.edu) ▾". Below this is a search bar with the placeholder "New Event". Underneath the search bar are several input fields and dropdown menus:

- "Add required people" with an "Optional" link.
- Date and time fields: "11/ 5/2023" (with a calendar icon), "from 4:30 PM", "to 5:00 PM", and a "All day" toggle switch.
- Recurrence settings: "Does not repeat" with a dropdown arrow.
- "Add a location" field with a "Teams meeting" toggle switch.
- Reminder settings: "15 minutes before" with a dropdown arrow.



Relevant Stakeholders/Users





Partner Introduction

Sketch of a Solution

1. OCR different styles of syllabi.
2. Upload the OCRed syllabi into a folder on Google Drive.
3. Write regexes to pull out the corresponding due dates and assignments.
4. Write LLM prompts to create ICS files to turn the data into Outlook invites.

Date	Topic	Homework
October 10 (Tuesday)	Criminal Law & Procedure (Part I): MBE Skill Focus: Rule Mastery Criminal Law Initial Learning Questions Part 1	Criminal Law Property Offenses (Est. Completion 40 min) Criminal Law Retention Building Quiz Criminal Law Essay #1 (Practice not submitted for grading) Criminal Law Essay #1 Model Answer Criminal Law Essay #2 (Submit for Grading Before Class on October 16)
October 16	Criminal Law & Procedure (Part II): Criminal Law Initial Learning Questions Part 2 Review of select Retention Building questions	Practice Midterm Part 1: Multiple Choice Part 2: Essay Submit for Grading by October 22 at 11:59 pm Review Practice Midterm Explanatory and Model Answers

Date
Topic
Homework
August 28 Course
(Optional – Read Civil Procedure outline)
Overview/Introduction to
Systematic Problem Solving Jurisdiction and Venue – Personal Jurisdiction (Est Completion 37 min)
Jurisdiction and Venue – Subject Matter Jurisdiction (Est Completion 55 min)
Jurisdiction and Venue – Joinder and Venue (Est Completion 33 min)