Lab 1 Schedule Puzzle Product

Kofi Amoah

Old Dominion University

CS 411, Fall 2023

Professor Thomas Kennedy

November 17, 2023

Table of Contents

List of Figures

1.   Introduction

People have a plethora of tasks to complete and sometimes scheduling, time management, and prioritizing can be a hassle. In a study conducted by Acuity Training, key takeaways were that less than 1 in 5 people had a proper time management system (Richardson, 2022). Additionally, 82% of people did not have a time management system; they just used a list or their email inbox. (Richardson, 2022).

Scheduling tasks, when done the right way can help an individual make the most out of their resources. Scheduling helps establish a consistent and predictable routine, which can reduce stress and improve efficiency (Profit.co, n.d.). This will lead to better decision-making, get rid of uncertainties, and build better time management. It will also make accomplishing tasks easier and increase productivity.

With people having a myriad of tasks and little time to complete them, it can be hard to prioritize. Someone might make a schedule, and they may later realize that that schedule does not work for them. As shown in Figure 1, is the current schedule creation process, which is all dependent on the user.

[*This space is intentionally left blank.*]
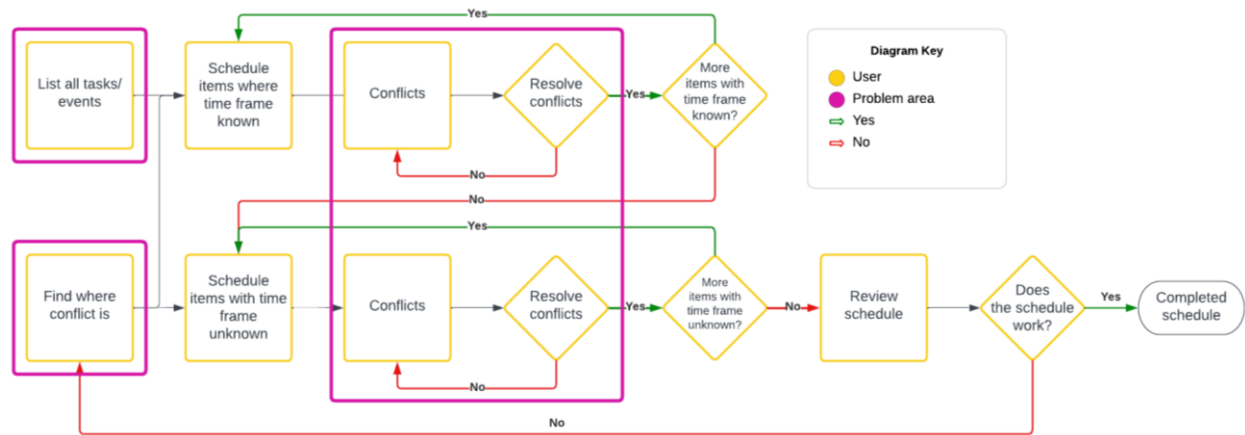
Figure 1: Current Process Flow

Scheduling tasks with a system that comes equipped with automated scheduling is an effective way to resolve this. This is where Schedule Puzzle comes in; it will alleviate strict and rigid schedule creation by allowing users to work with the system. The product will also allow the user to help with conflict resolution during the process. Once completed, the user can select from multiple schedules and refine any schedule they pick.

[This space is intentionally left blank.]

2.   Schedule Puzzle Product Description

Schedule Puzzle is a web application that will automate schedule creation. The user will input the name of the task, its duration, and the number of times to repeat it if desired. The automation feature of the app then creates a schedule based on the above criteria. Schedule Puzzle will label and notify users of tasks and prioritize them based on categories, deadlines, days, and times of day. Schedule Puzzle not only allows for schedule creation within the system, but users can also export their calendars from other applications as well. Users also have the flexibility to export their schedule into other calendars and spreadsheets. Schedule Puzzle also lets users create their own categories using labels; for instance, one could categorize their tasks into work, home, recreation, and school based on what they are associated with. Schedule Puzzle also features multiple calendar interfaces, including the Google Calendar API, Apple EventKit, and Microsoft Graph. These are used for integrating the Google, Apple, and Outlook calendars, respectively.

2.1. Key Product Features and Capabilities

The key product features of Schedule Puzzle are its capability to automate schedule creation based on the tasks, duration, and events supplied to it by the user. Schedule Puzzle is unique among its competitors because it allows for automated schedule creation based on user inputs. Schedule Puzzle will allow users to input tasks at the beginning. At the very least, the user will have to provide the name of the task, the duration, and how often it will repeat. Schedule Puzzle also allocates preferences when it comes to scheduling tasks that the user flags for prioritization. Tasks are labeled based on categories, and users are notified of tasks and time conflicts. If time

conflicts are found, the user is notified and decides on the resolution. After conflict resolution, the

next prioritized task is scheduled, and the process repeats. This aids in conflict resolution and

alleviates the problem of rigid schedule creation by allowing users to work in conjunction with

the system.

2.2.   Major Components (Hardware/Software)

Schedule Puzzle can work on personal computers (desktops and laptops), cellular devices, and

tablets that have internet access. It has multiple servers, with the frontend being implemented in

JavaScript, HTML, and CSS and the backend being implemented in Python. It is hosted on two

databases: Amazon Web Services and PostgreSQL. The framework used is Django, and the

repository is available on GitHub.

3.   Identification of Case Study

Schedule Puzzle is for individuals who struggle with time management, organization, and

prioritization of tasks. It is also for individuals who already use calendar applications. Planned

tasks can be too rigid to follow through with sometimes, and opportunities to be productive can

be missed, and important tasks can be neglected. Schedule Puzzle will alleviate the stress that

comes with schedule creation. Students, organization leaders, and starting professionals are

examples of groups who will find the automated schedule process Schedule Puzzle implements

useful.

[*This space is intentionally left blank.*]

4.   Schedule Puzzle Product Prototype Description

Schedule Puzzle's main goal is to aid in schedule creation and reduce possible scheduling

conflicts by automating the process based on user inputs. Due to time and resource constraints,

only features essential to proof of concept will be implemented in the prototype. Due to the

complex nature of certain features, they will only be partially functional in the prototype.

Prioritized features will include semi-automatic conflict resolution, custom prioritization,

behavior suggestions, importing and exporting calendars, and multiple calendar views.

4.1. Prototype Architecture (Hardware/Software)

The prototype will have a web application and a mobile application and, as such, will run on

computers, laptops, tablets, and mobile devices. The software utilized on the prototype will be

HTML, JavaScript, and CSS for the frontend. The prototype backend will be implemented in

Python. The prototype intends to prove that automated schedule creation based on user inputs will

alleviate the stress and conflicts associated with schedule creation. The prototype's automated

features will be prioritized. Features that are not essential to the proof of concept will only be

partially functional in the prototype. As shown in Figure 2, the prototype will be a Django

application that uses the PostgreSQL database management system.

[This space is intentionally left blank.]
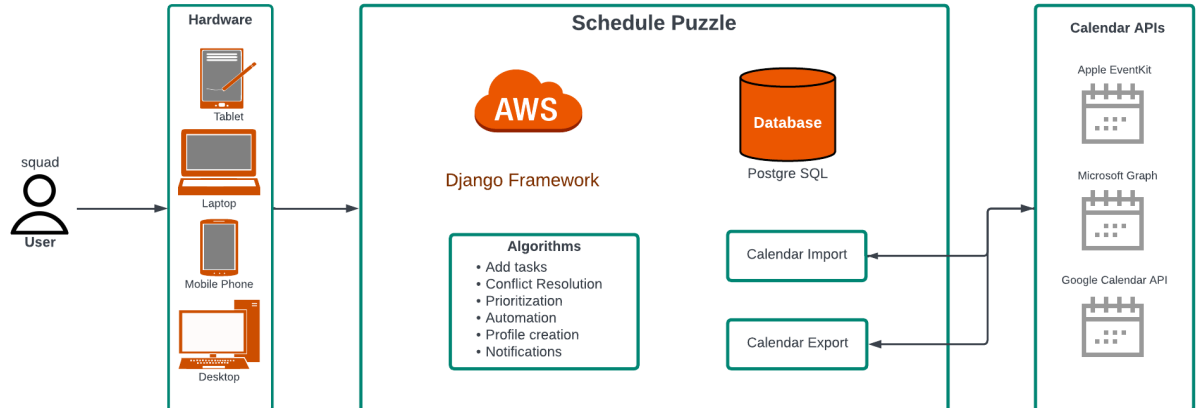
Figure 2: Major Functional Component Diagram

Calendar APIs will help in the importing and exporting of calendars from external applications.

The application will allow the user either add tasks or import calendars, and afterwards they are

sorted by priority. Tasks with conflicting priorities are resolved with the help of the user, and then

the schedule is created. Only the essential features for the prototype will be fully implemented or

partially due to time and resource constraints.

4.2. Prototype Features and Capabilities

The features implemented in the Schedule Puzzle program show how schedules can be generated

quickly and uniquely to meet an individual's needs. Users will be able to create an account to

store, edit, and view their various schedules. As shown in Figure 5, when the user is at the Login

Page and has an existing account, they will log into their account and be directed to the schedule

dashboard. If the user does not have an existing account, they will be directed to create one by
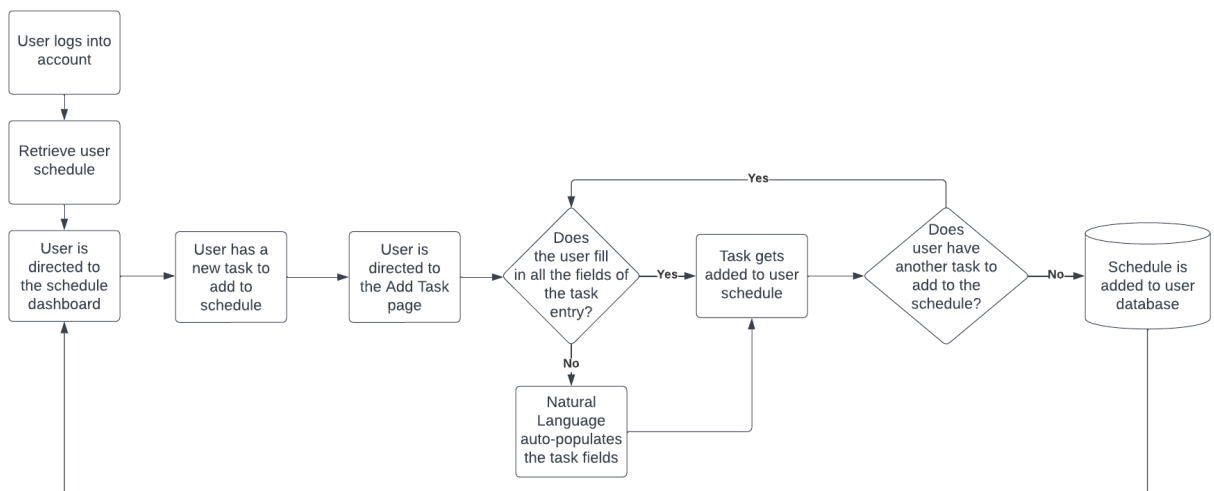
entering their account information.

*[This space is intentionally left blank.]*
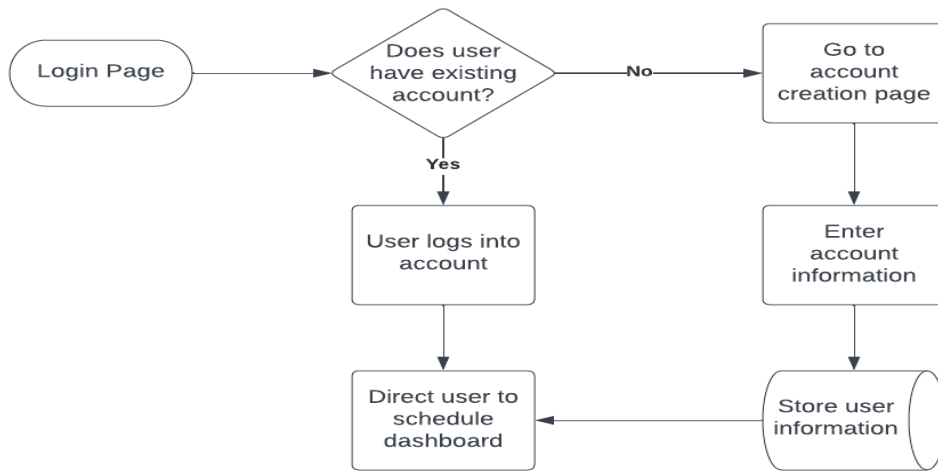
Figure 3: Profile Creation Algorithm

Users will be able to add tasks by entering in the task fields, using labels, or using the Natural

Language Processing feature implemented in Schedule Puzzle. Figure 6 shows the procedure that

the 'Add Task Algorithm' will take.


Figure 4: Add Task Algorithm

Additionally, the prototype will address the risks involved with scheduling conflicts by carefully

considering all potential hazards and tackling them with industry best practices. Schedule Puzzle

will aid users in schedule creation and alleviate the stress and errors that are associated with the

process. The prototype aims to bring automatic schedule creation to individuals. As shown in

Figure 7, the automated schedule creation process starts at the schedule dashboard when the user

has a new schedule to create. The process concludes with the user's schedule being created and

stored.
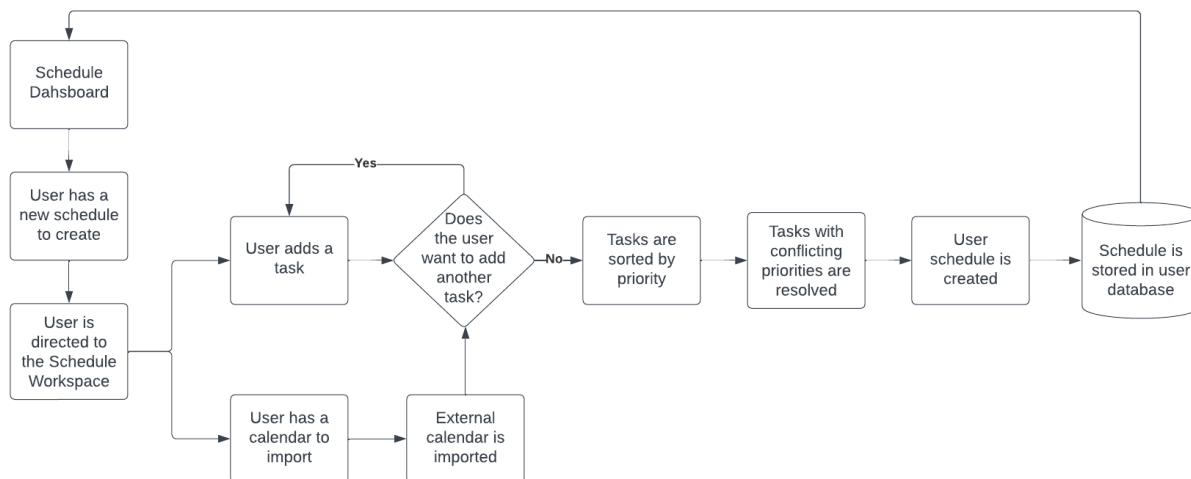
Figure 5: Automated Schedule Creation Algorithm

The prototype is also equipped with features that allow users to choose whether they want

notifications via text, email, or push notifications. The prototype will also allow users to export

and import files to and from external applications such as Apple Calendar, Microsoft Outlook,

and Google Calendar in the form of ics files to Schedule Puzzle. Additionally, it will allow users

to export their schedules from Schedule Puzzle into CSV format..Figures 9 and 10 display what

the process of calendar importation and exportation from and to an external application would

look like.
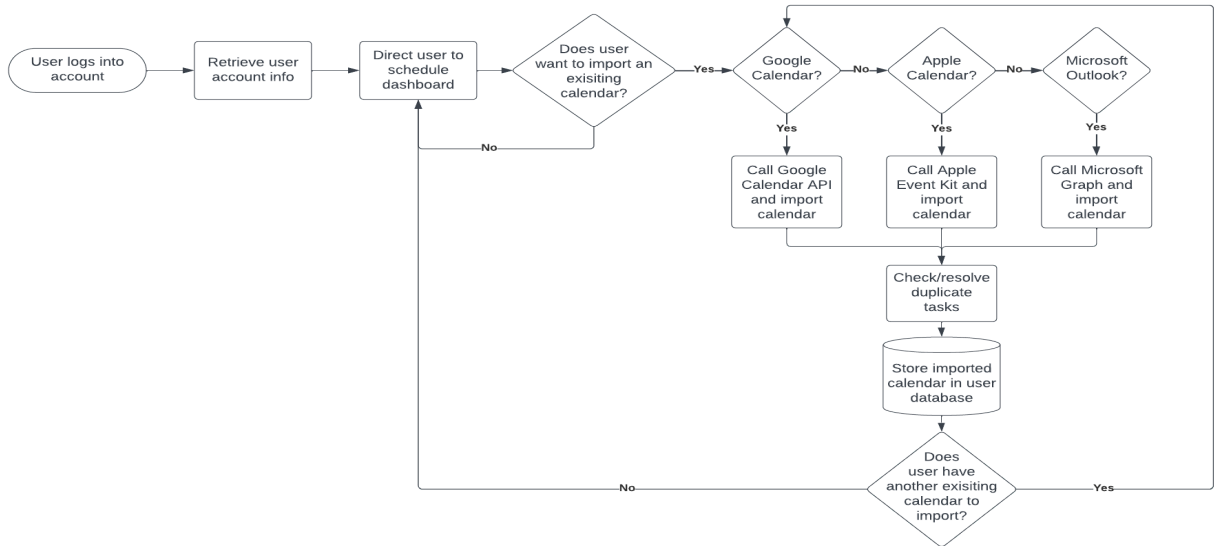
Figure 6: Import Calendar Algorithm

[*This space is intentionally left blank.*]
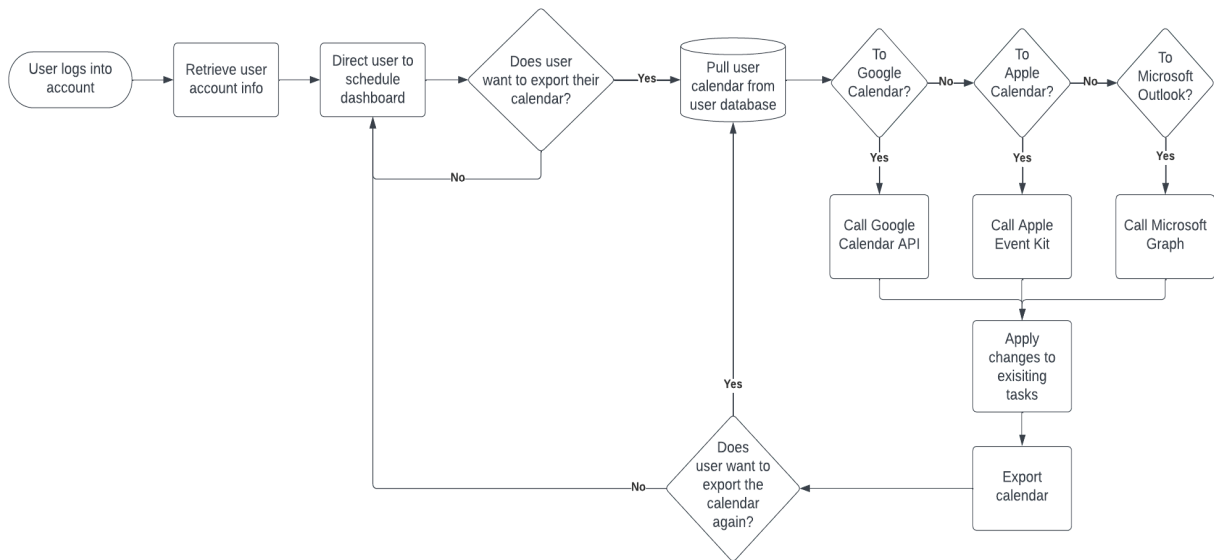
Figure 7: Export Calendar Algorithm

The prototype will also feature semi-automatic Conflict resolution, but the functionality will be limited because the natural language processing feature will be partially functional in the prototype. In addition, the prototype will use a numbering system, which will improve the rate at which people accomplish their high-priority tasks. The numbering system ranges from 1 to 3:

1. tasks with the highest priority

2. tasks with the second highest priority

3. tasks that are the least prioritized

[*This space is intentionally left blank.*]

4.3. Prototype Development Challenges

One of the first development challenges will include becoming adept with some of the chosen programming languages, APIs, tools, and libraries. All team members do not have advanced knowledge in Javascript, PostgreSQL, AWS tools and services, databases, web application development concepts, or natural language processing tools. Additionally, after learning how to use these databases, APIs, cloud services, and concepts, implementing them in a way where they best complement each other will be another challenge on its own. Not all group members have experience integrating a significant number of libraries, tools, and services into an application. Strides will be made to identify resources, share these resources amongst the team, and guide each other.

[*This space is intentionally left blank.*]

5.   Glossary

Application Programming Interface (API): mechanisms that enable two software components to communicate with each other using a set of definitions and protocols

Automatic: a device or process working by itself with little or no human control over it

Amazon Web Services: cloud service that is a subsidiary of Amazon that provides reliable,scalable, inexpensive, and on-demand cloud computing services and platforms to individuals, companies, and organizations.

Backend: parts of a computer application or a program's code that allow it to operate and that cannot be accessed by a user.

Cascading Style Sheets (CSS): A style sheet language used to style or describe the presentation of a document written in a markup language such as HTML or XML.

Categorize: to place in a particular class or group

Database: a collection of structured information, or data, typically stored electronically in a computer system.

Frontend: layer above the back end that includes all software or hardware that is part of a user interface.

GitHub: a platform for hosting software development projects and version control using Git, allowing developers and users to store and manage their code.

Javascript: an object-oriented programming language commonly used to create interactivity on a webpage.

Label: assign to a category

Natural Language Processing: refers to the branch of artificial intelligence (AI) that deals with giving computers the ability to understand and process text and spoken words in much the same way that human beings can.

Python: an interpreted, object-oriented programming language with high semantics that prioritizes readability and clarity of code.

Server: a computer or system that provides resources, data, or services to other computers known as clients over a network.

Task: catch-term for all things that need to be completed by the user.

6.   References

Indeed Editorial Team. (2021, February 22). *12 Time Management Problems (and How To Fix Them)*.   Indeed.   Retrieved   from https://www.indeed.com/career-advice/careerdevelopment/time-management-problems

Nemko, M. (2021, December 3). *4 Causes of Poor Time Management | Psychology Today*. Psychology   Today.   Retrieved   from https://www.psychologytoday.com/us/blog/how-todo-life/202112/4-causes-of-poor-time-management

Prabhu, A. (2022, November 25). *Importance of scheduling tasks and its benefits*. Profit.co. Retrieved   from https://www.profit.co/blog/task-management/importance-of-schedulingtasks-and-its-benefits/

Richardson, B. (2022, October 26). *Time Management Statistics & Facts (New 2022 Research)*. Acuity   Training.   Retrieved   from https://www.acuitytraining.co.uk/news-tips/timemanagement-   statistics-2022-research/

Profit.co. (n.d.). *Importance of Scheduling Tasks and its Benefits*. Retrieved   from https://www.profit.co/blog/task-management/importance-of-scheduling-tasks-and-itsbenefits/#:~:text=Benefits%20of%20Scheduling&text=A%20schedule%20establishes%20a%20predictable,productivity%20planning%20and%20goal%20management.

Oracle. (n.d.). *What Is a Database?*.Retrieved   from https://www.oracle.com/database/what-is-database/