Lab 1 Schedule Puzzle Product

Kayla Pineda

Old Dominion University

CS 411W, Fall 2023

Professor Thomas Kennedy

November 17, 2023

Version 2

Table of Contents

Table of Figures

Table of Tables

1.   Introduction

People have many tasks and comparatively little time. In a 2022 study of 500 employees

across several industries, less than 18% have a proper time management system, with the other

82% using a list or their email inbox as a time management tool (Richardson, 2022). From the

same study, 12.5% of people felt as if they never had control over their work (Richardson, 2022).

Scheduling is a solution to this. Scheduling has benefits such as improved structure and

increased productivity (Prabhu, 2022). By scheduling tasks, people have more control and

understanding. The problem is scheduling does not come naturally to all. The current approach to

scheduling is all done by the user. As shown in Figure 1, the typical approach is to first schedule

tasks with a specific time frame known, then schedule tasks that do not have a specific time

frame known. However, a person may create a schedule only to realize the schedule does not

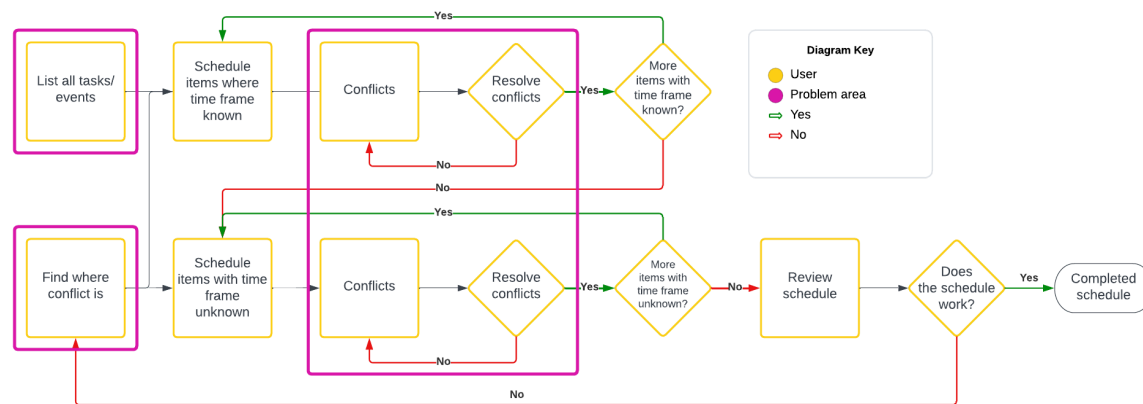work for them, resulting in that person starting over.

Figure 1: Current Process Flow

The solution is automated scheduling. The system and the user work together to create a

schedule. The user inputs their tasks, and the system prioritizes and schedules tasks based on

user given tasks and events. This is the goal of Schedule Puzzle.

2.   Schedule Puzzle Product Description

Schedule Puzzle is a semi-automatic schedule creator that uses user-supplied tasks. In addition to having basic calendar functionalities, features such as importing and exporting, labeling tasks, and sending reminders, Schedule Puzzle aids the user in creating a custom schedule. Figure 2 illustrates the flow of Schedule Puzzle. The user supplies the basic information of their tasks, Schedule Puzzle ranks the task based on priority, and a schedule is created for the user based on the user's inputs.
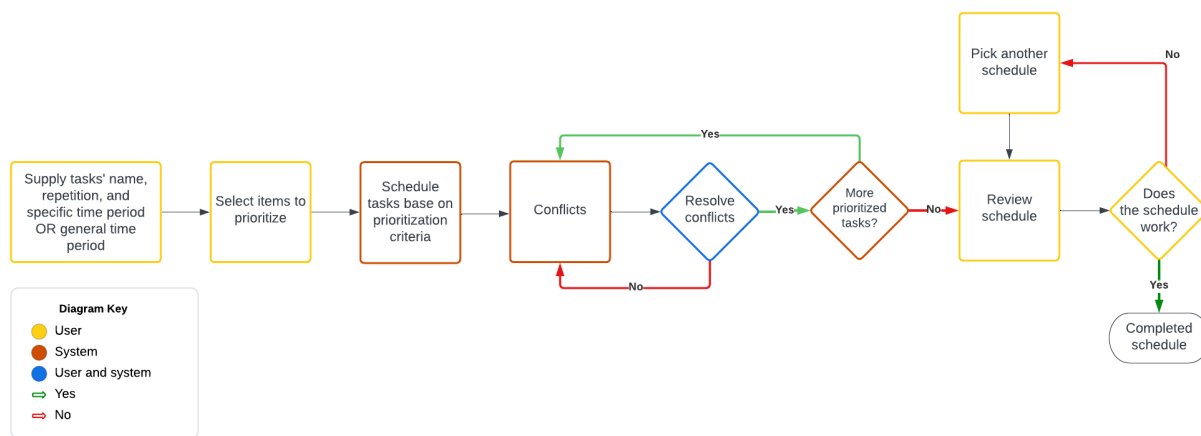

Figure 2: Solution Process Flow

2.1.   Key Product Features and Capabilities

The key feature of Schedule Puzzle is its ability to semi-automatically create a schedule based on user-supplied inputs. At the start, users input their tasks. Users must supply the name of the task, its duration, and number of repetitions. Users can also supply the priority level or the category of a task. When entering the fields, users can manually input task data or use Natural Language Processing (NLP) to populate the fields. Schedule Puzzle uses NLP to rank the tasks based on priority. Tasks with higher priority will be scheduled first. If conflicts between tasks arise, the user is notified about the conflict and is asked how to resolve it. This semi-automatic

approach to scheduling alleviates user stress by allowing users to work together with the system to create a schedule.

2.2.   Major Components (Hardware/Software)

Schedule Puzzle is a web-based program, requiring no special hardware for the user. The hardware required to use Schedule Puzzle is a personal computer, a desktop or laptop, a cellular device, or a tablet. Any device that can access the internet is enough to access Schedule Puzzle.

Schedule Puzzle uses HTML, JavaScript, and CSS on the frontend. Python is the main language Schedule Puzzle is coded in and uses the Django framework. Schedule Puzzle uses PostgreSQL as its database. The IDE used to code Schedule Puzzle is Visual Studio Code. Amazon Web Services (AWS) is used to host the website, and GitHub is used to store the repository.

3.   Identification of Case Study

Schedule Puzzle is aimed towards individuals who struggle with time management and those who already use calendar applications. Individuals who struggle with time management would benefit from Schedule Puzzle by providing them the scheduling help they need. Users work together with the system to create a schedule that suits their needs. Individuals who already use calendar services would also benefit from Schedule Puzzle as it may help them refine their existing schedules.

Possible users in the future are students, administration clerks, organization leaderships, and starting professionals. Schedule Puzzle would aid these users in organizing the various tasks they have.

4.   Schedule Puzzle Product Prototype Description

The aim of the Schedule Puzzle prototype is to have most of the basic calendar functionalities and some automation and prioritization features implemented. Table 1 displays all of the main features in Schedule Puzzle, if they appear in the real world product, and if they appear in the prototype.

Feature	Real World Product	Prototype
Basic Calendar Functionalities		
Log in to profile	Fully functional	Fully function
Import existing schedules (.ics, .csv)	Fully functional	Fully functional
Export existing schedules (.ics, .csv)	Fully functional	Fully functional
Has daily/weekly/monthly calendar interface	Fully functional	Fully functional
Modify tasks	Fully functional	Fully functional
Create notes inside of tasks	Fully functional	Fully functional
Send reminders/notifications (push, text, email)	Fully functional	Partially-Email only
Automation, Customization, and Prioritization		
Automatic schedule creation	Fully functional	Fully functional
Semi-automatic conflict resolution	Fully functional	Fully functional
Custom prioritization	Fully functional	Partially functional
Natural language processing	Fully functional	Partially functional
Behavioral suggestions	Fully functional	Eliminated

Table 1: Real World Product vs. Prototype

4.1.   Prototype Architecture (Hardware/Software)

The architecture for the prototype is mostly the same as listed in Section 2.2. The first

exception is the prototype will primarily focus on using a personal computer, such as a laptop or

desktop, for the hardware. The other exception is the prototype will not use AWS to host the site.

The prototype will be hosted on Old Dominion University's Computer Science servers. Figure 3

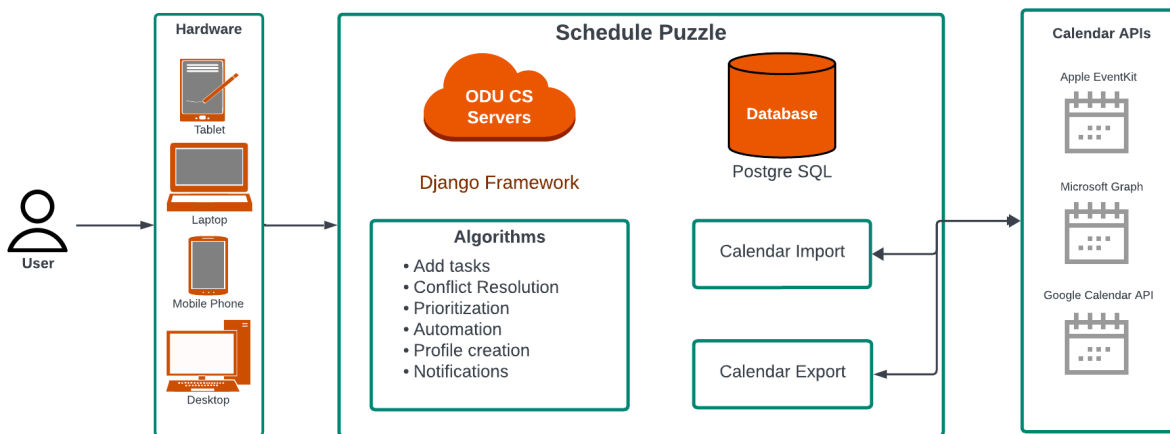illustrates the prototype architecture for Schedule Puzzle.

Figure 3: Prototype Major Functional Component Diagram

4.2.   Prototype Features and Capabilities

For the prototype, basic calendar functions and some automation and prioritization

features will be implemented.

Starting with the basic calendar functions, the first function planned to be included in the

prototype is profile creation. Users will need to create an account to store, edit, and view their

schedules. Since sensitive data such as emails and passwords will be stored, industry standard

encryption and safety protocols will be followed. Figure 4 illustrates how users will create their
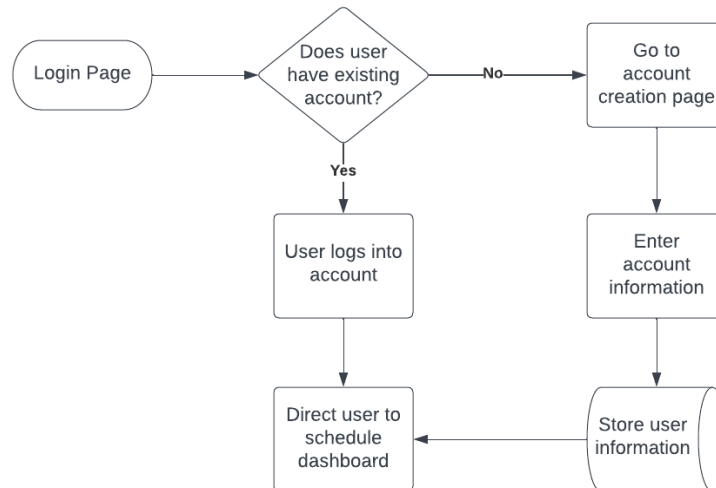
account.

Figure 4: Profile Creation Algorithm

The next two functionalities are importing and exporting. As shown in Figures 5 and 6, users will also have the ability to import and export from external calendar applications, such as Apple Calendar, Microsoft Outlook, and Google Calendar.
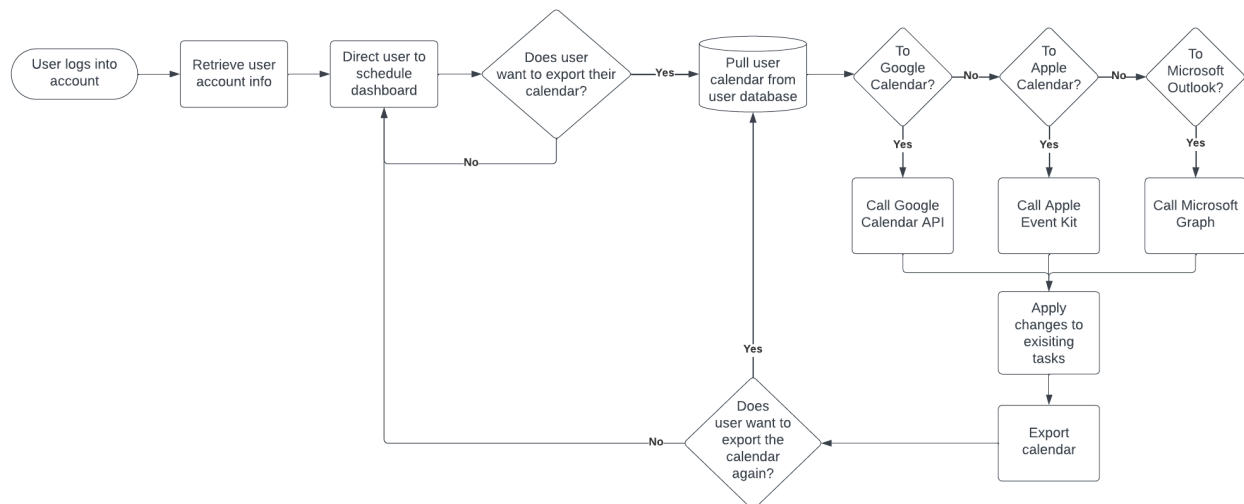

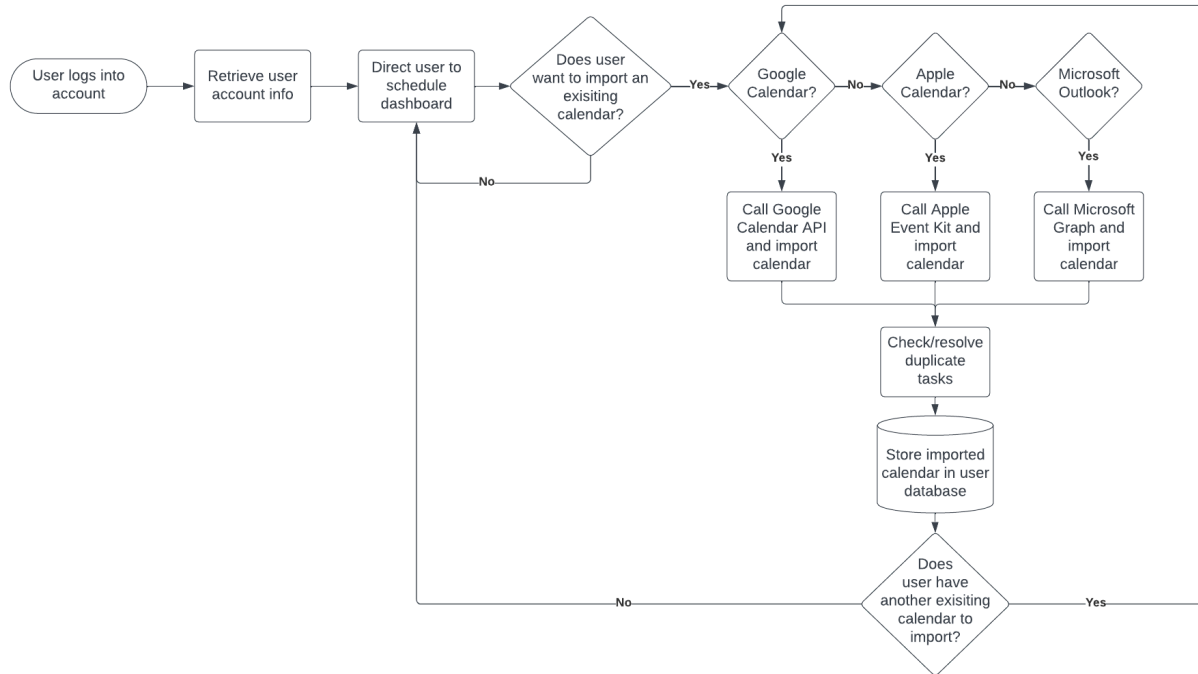*Figure 5: Import Calendar Algorithm*

Figure 6: Export Calendar Algorithm

Another basic function to be included in the prototype is the ability to add or modify

tasks. Figure 7 shows how users will have the ability to add tasks by entering in the task fields

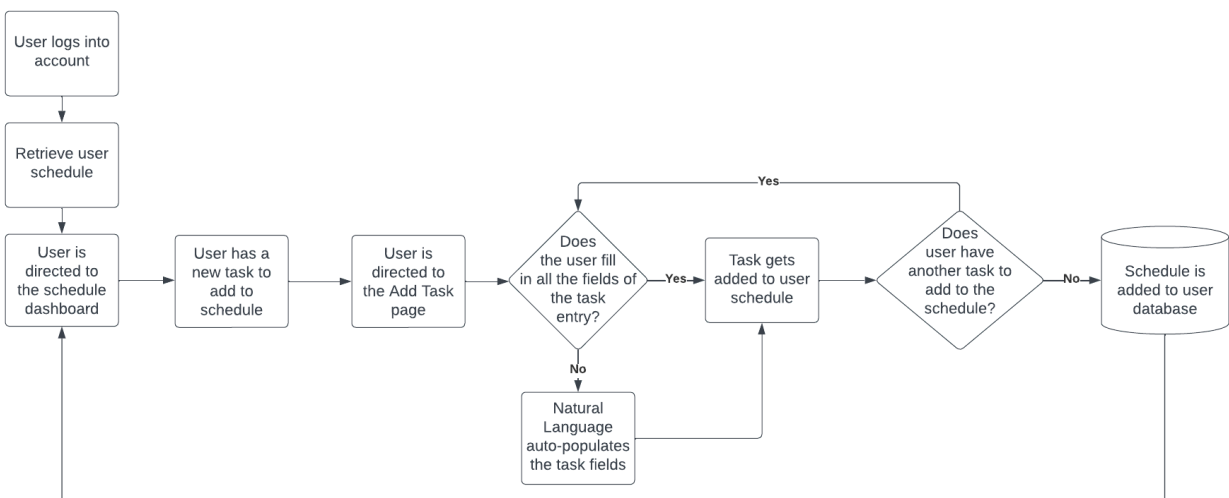manually or by utilizing NLP to populate certain fields.

Figure 7: Add Task Algorithm

The final basic calendar function that will be in the prototype is the ability to receive

notifications. Users will have the ability to opt for notifications via email, text, or push
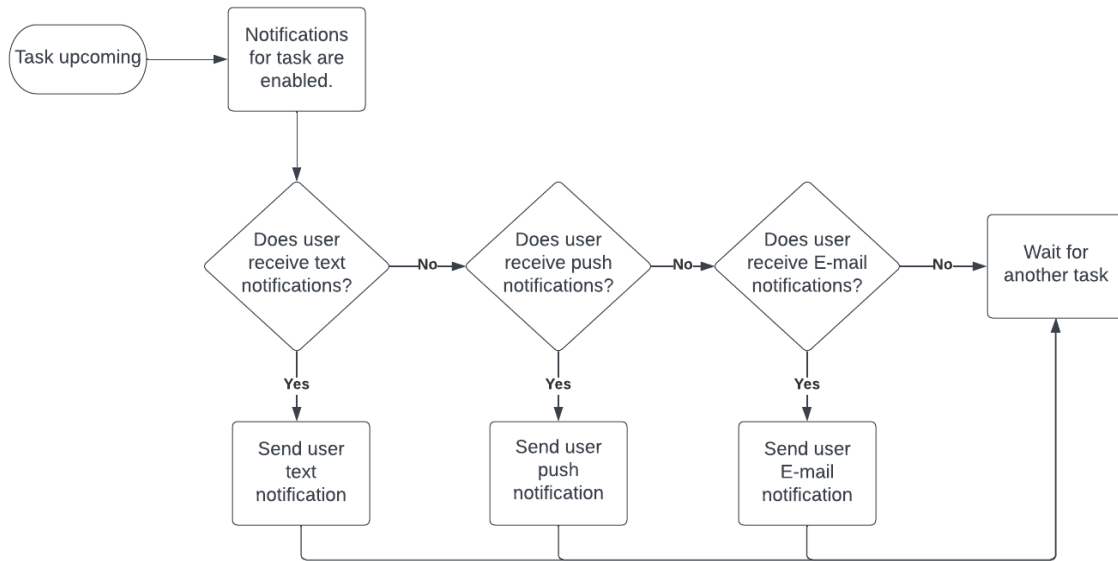
notifications. Figure 8 illustrates this.

Figure 8: Notification Algorithm

The main focus of the prototype is the automated schedule creation algorithm. As shown

in Figure 9, this algorithm will take the user's tasks, order them by priority, and output a
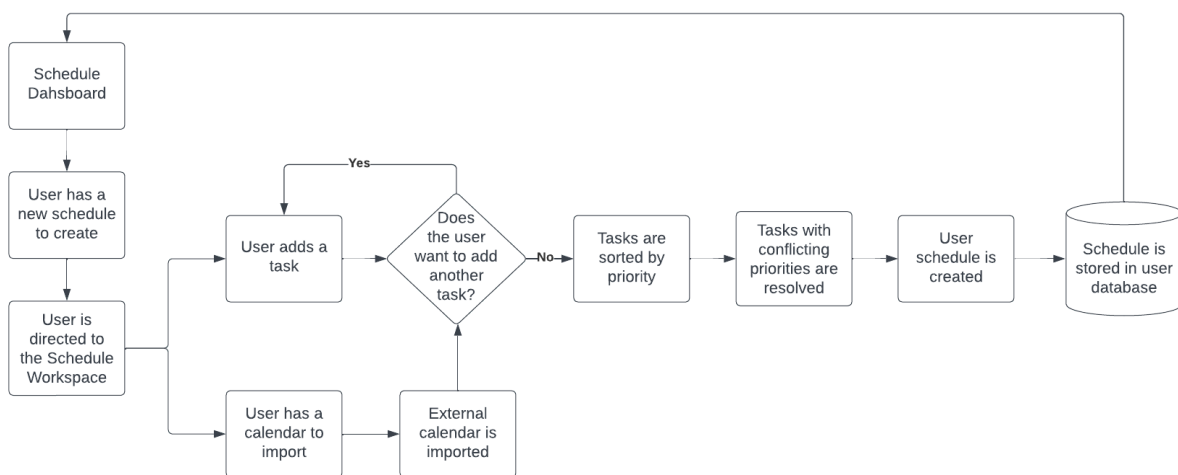
schedule for the user.

Figure 9: Automated Schedule Creation Algorithm

Users will aid in the process of creating their schedule through semi-automatic conflict resolution. Figure 10 illustrates how the algorithm will identify and resolve conflicts. If the tasks have the same priority, the user is prompted to select which task to schedule.
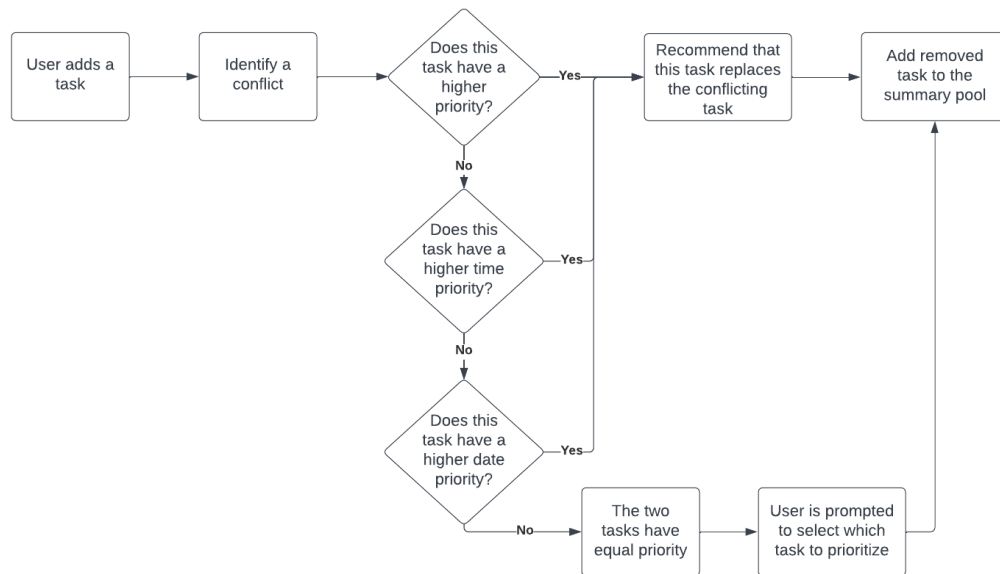

Figure 10: Semi-Automatic Conflict Resolution Algorithm

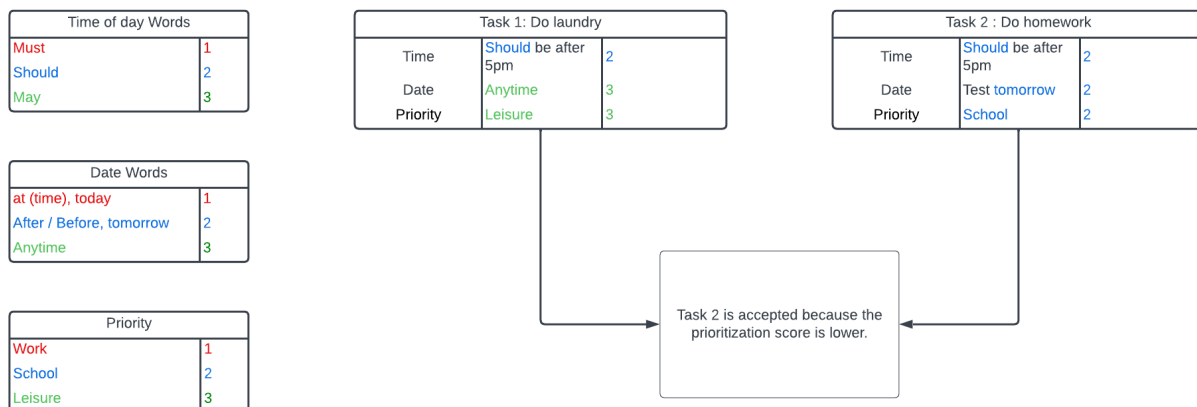Figure 11 provides an example of how NLP will score and rank each task according to priority.

Figure 11: Prioritization Example

4.3.   Prototype Development Challenges

There are three major challenges during prototype development:

1. The first challenge is learning the Django framework.

2. The second challenge is learning Python and the APIs to be used.

3. The third challenge is implementing NLP.

For each challenge, the team plans to use online tutorials, online documentation, and practice programs to learn how to use each tool. Every member of the team will be getting experience with each tool, regardless of if their role demands it. This is so each member gains experience and to help other team members when needed.

5.   Glossary

Application Programming Interface (API): Software that allows two or more computer programs to communicate.

Amazon Web Services (AWS): Service that provides on-demand cloud computing and APIs to individuals and organizations.

Cascading Style Sheets (CSS): Style sheet language used to format markup languages such as HTML.

Django: Python based framework.

HyperText Markup Language (HTML): Markup language used to display web pages.

Integrated Development Environment (IDE): Software application used for software development

Natural Language Processing (NLP): Machine learning used to interpret human language.

One time task: Tasks that do not repeat on a normal basis such as appointments or meetings.

PostgreSQL: Relational database management system.

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

Recurring task: Tasks that do repeat on a normal basis such as school or work.

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/career-development/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-to-do-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-scheduling-tasks-and-its-ben

efits/

Richardson, B. (2022, October 26). *Time Management Statistics & Facts (New 2022 Research)*.

Acuity Training. Retrieved from

https://www.acuitytraining.co.uk/news-tips/time-management-statistics-2022-research/