```
# Author:           Martin Solum
# Date:             18DEC2024Wed
# Course:           CIS129 Prog & Problem Solv I
# Instructor:       Brittany Griwzow
# Assignment:       Final Project
```

Personal Information Manager (PIM) Productivity tool…

Table of Contents

Motivation

```
* The is my Final Project proposal for Fall2024 CIS129 Programming &
Problem Solving I at Pima Community college.  The assignment is not
to develop a software system or even a Software Requirements
Specification.  Rather, the assignment is to do research to select a
target software problem, design a solution, and produce & submit a
presentation.

* The target area I have chosen is to create a Personal Information
Manager (PIM) Productivity tool.  My primary motivation is actually
that I want such a tool.  I have been keeping track of tasks, time,
appointments, goals, habits, & such in computers for decades.  Over
that time I have often encountered tools that I found very helpful
in one way or another.  But I've never come across a system that
```

does exactly the things in this space I find critical. The system needs to be lightweight, quick, & and effective.
Critical Terminology

* Wikipedia #1: defines a PIM as:
“As an information management tool, a PIM tool’s purpose is to facilitate the recording, tracking, and management of certain types of "personal information".

* The CheckPeople.com#2 blog article "What is a Personal Information Manager (PIM)?" by Michelle W ilson - January 20, 2023, defines a PIM this way:
"By definition, a personal information management (PIM) system is a tool or set of tools that help individuals organize and manage their data, including tasks, events, contacts, notes, and other information".

* Michelle Wilson states further:
[...] PIMs [have a] task or to-do list function, which allows users to create and prioritize tasks, set deadlines, and track progress. These features can help manage daily responsibilities, long-term projects, and goals. Other PIMs may also offer calendar and event management tools, which allow users to schedule appointments, meetings, and other events and receive reminders and notifications.

Existing Solutions in this Space

Calendar Applications

* The above definitions depict well the general PIM software application space I am focusing on. It t
urns out that a wide variety of tools have been used to accomplish such tasks. Most notably to begin with is that Google, Microsoft, & Mozilla, to mention just a few, offer calendar applications that provide at least some PIM functionality. On the Linux platform there is a very straight forward calendar program called CalCurse. See Calcurse reference. See reference to CalCurse below under “Command Line Interfaced based applications".

Note-taking Applications

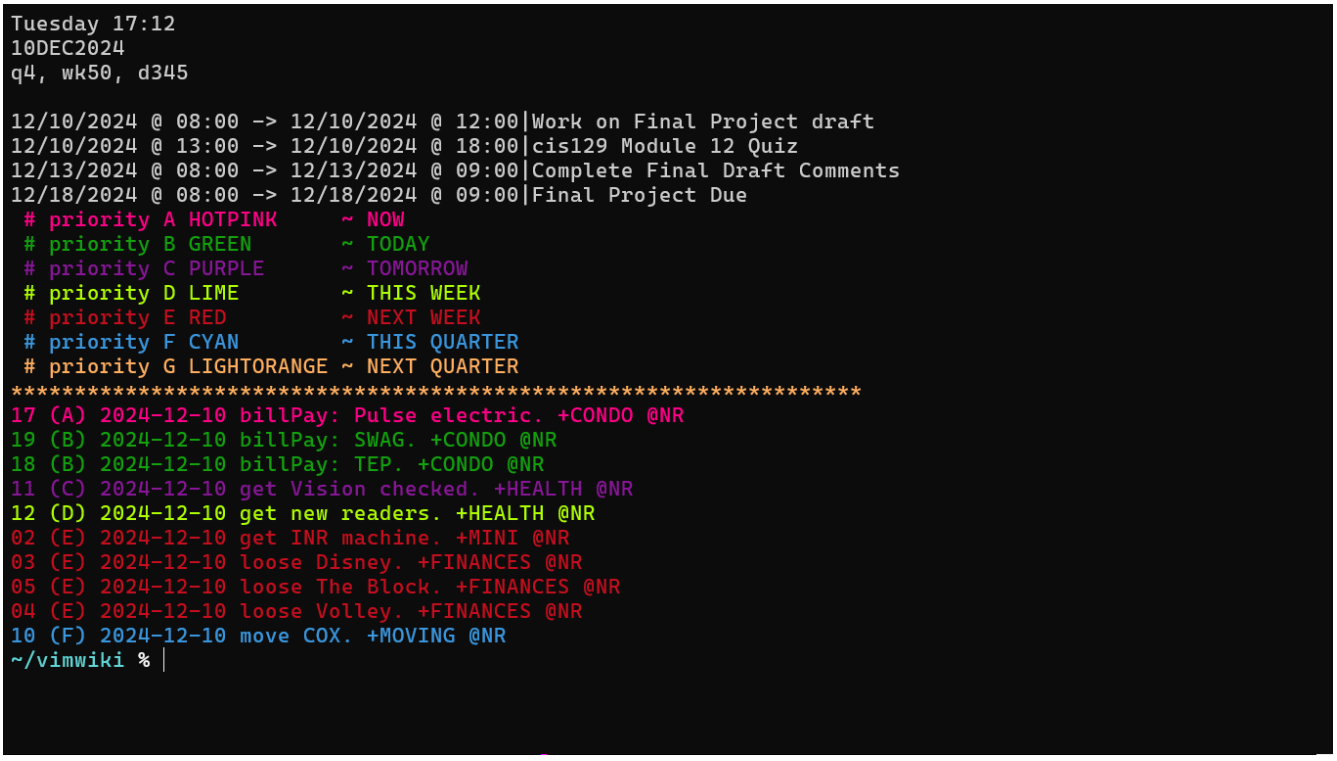
* There is a huge new movement bursting on the scene called Note-Taking Systems, as defined by Tiago
Fiorte in his seminal works "Building a Second Brain, & "The Para Method". Notable tools in this space include "Obsidian", "Notion", "Microsoft One-Note", & "Joplin". Precursors to these technologies that existed for some time include "CherryTree", & "Zim" and many others. Although these systems have a much broader charter than the PIM scope I have identified, they all can be used to provide some PIM functionality and often have specific calendar plugins to provide PIM functionality.

Text-Editor based Applications

* The two major Unix\Linux platform editors each have various plugins and approaches for facilitating
PIM processes. Probably most comprehensive of these is the Org-Mode extention to emacs text editor. Another approach is to use a Wiki style interface such as the VimWiki plugin for the vim text editor. Back in 1995 Ward Cunningham developed the Wiki system for user edited websites which spawned a plethora of specialized website editing software packages such as MoinMoin & DocuWiki. Although, to my knowledge, none of these week systems were primarily focused on the area I've identified as the PIM software space, all have them have had features that could be used to quickly track many kinds of information including traditional PIM features (tracking tasks, time, appointments, etc).

Command Line Interface based Applications

* There are some interesting old school text based tasks management systems on Linux as well. Most noteably ToDo-txt (which I will discuss in some depth later in this presentation) & TaskWarrior. One of the most important experiments in the preparation of this document was to integrate the CalCurse calendar application and the ToDo-txt applications. A combined display of calendar data exported from CalCurse & ToDo data exported from ToDo-txt. see figure 1.



Project Management Applications

* Task management can also be a precursor to more ambitious project management systems e.g. Microsoft Project & Primivera.

Other related Applications

* There are many other solutions that touch on the PIM problem space including Time Card tracking systems, contact management systems, email based workflow systems, generic workflow systems etc.

I don't have space here to drill down into all of the strengths and weaknesses of these various applications; but it is important to note that experience with all of these products has substantially influenced my thinking on this project.

Specifically: What the System Shall Do

Because of the wide range of features in the various systems above, it is critical to identify which are the critical features as opposed to the merely the "nice-to-haves". I have identified these specific features as critical.

* The system shall provide an interface for setting appointments, tasks, reminders, time logging, & comprehensive reporting functionality.

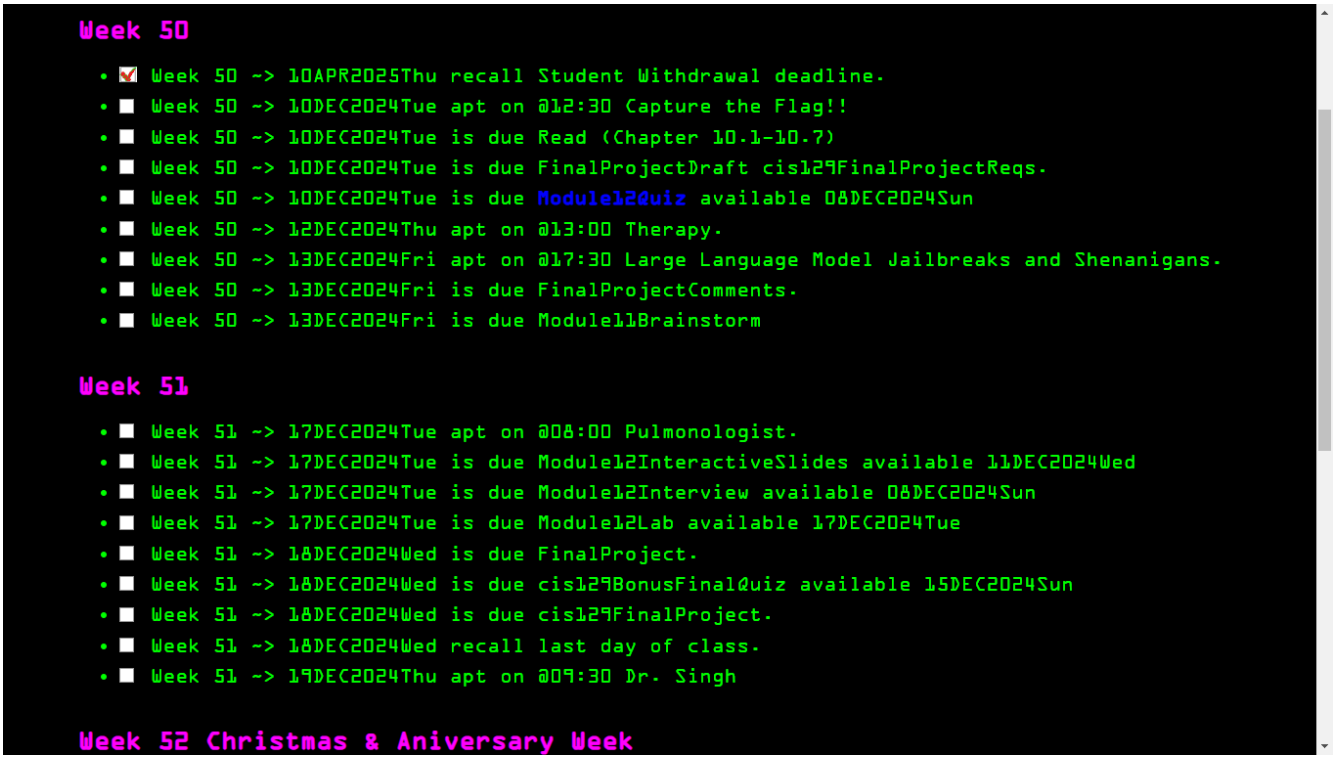
Unique Value Proposition (UVP)

Obliviously many of the applications listed above provide functionality for setting appointments, tasks, reminders, time logging, & comprehensive reporting functionality. How will this PIM Productivity system be any better than all the rest?

Weekly-based PIM System

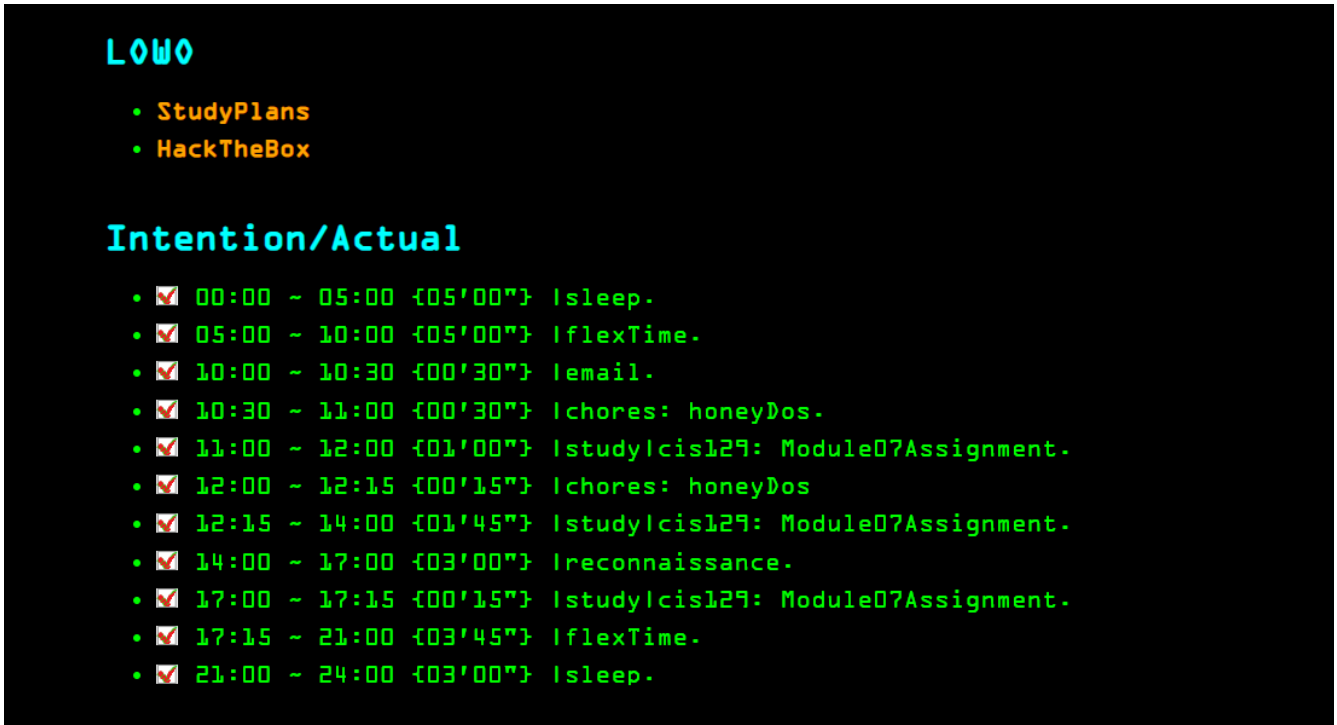
This is a specialized solution for people who value ease of data entry, week|quarterly|annual planning periods, & ambitious time tracking with a minimal amount of data entry.

ISO numbers weeks. Steven Covey, in his highly lauded book "7 Habits of Highly Effective People" emphasized strongly the need to stop thinking in a one day at a time manner and start planning & doing things with a week-based orientation. Not everyone believes in week-based planning. This application is not intended for everybody. It is intended for people who have already committed to organizing their life on along the lines of week-based planning employing ISO week numbers. This application shall support week-based planning. Figure 2 displays this sort of approach…

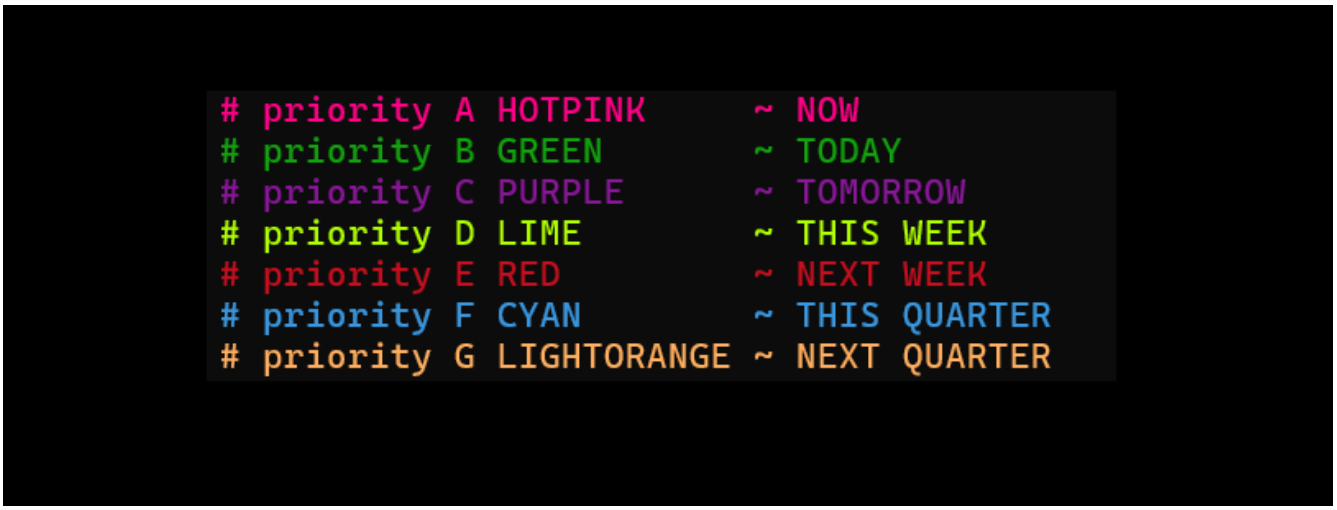


Application Tracks both Task Time and other Time usage

When tracking time, it is nice to be able to track how much time a given sub-project or feature required. However, some other activities, like processing email, are done every day and a more granular break-down of the time would not be beneficial. This application shall support both tracking time by tasks and time doing general activities.
Here's some task tracking as an example:



Provides a unique, minimalist, yet effective prioritization system.



Basic todo tasks prioritization often delineate Urgent & Important tasks as A's or B's and Important but not Urgent tasks as C's or D's. This sort of prioritization doesn't help scheduling that much. A better approach is to set rough time guidelines for tasks. This makes it possible to categorize what will be done next, today, tomorrow, this week, next week, etc. This application shall use this sort of system.

Additional Functionality

* Provides a unique, minimalist, yet effective time logging system.

* Flexible report configuration & generation.

* Hierarchical task management.

* Also tracks billable hours.

* A customized date formatting system.
Perhaps somewhat idiosyncratic, I like a terse informative date format, i.e. 10DEC2024Tue.

* Ambitious emphasis on minimizing the required number of keystrokes required.

* Coordinated support for alternate device interfaces.

* {later release} LapTop, phone, or Server based system storage.

* {later release} Off-line storage support for Google drive, Microsoft OneDrive, DropBox.

A note on Interface

* I need to make a few comments on the user interface. The early prototyping was done in text mode, specifically in NeoVim/VimWiki & using the command line interface for ToDo-txt. Is such an old school command line interface the right user interface for the PIM Productivity System? Maybe, maybe not. The use of an old school command line interface provides some advantages. Sometimes such an interface can be the most expedient. This is especially true when setting up a new system from scratch. However, many applications, like this one, are best deployed on a mobile system such as a cell phone or a tablet. In such cases, old school command line interface is a non-starter. Even on a laptop or desktop, having a GUI can make the system much easier to navigate & use. For that reason I've been looking at GUI technologies for Python e.g. BeeWare & Kivy.

* Python is not the most mature language for the cell/phone | tablet platforms (e.g. Android & iPhone). It might be better to use a proven Android or iPhone development technology such as Java, Kotlin, or Swift. TBD.

* Regardless of what GUI technology is adopted in the future. I intend to begin with an old school command line interface. Part of the reason for this is that it might turn out to be and adequate interface for all or part of the application. But the real reason to build a text interface is that a text interface greatly facilitates testing and by starting out with a text interface, early GUI decisions can be post-poned until the application design has matured a bit. By developing the text interface first we can remain GUI agnostic, possibly developing more than one GUI interface for the application. Also, some of the tools mentioned in the initial discussion of this project above, e.g. Obsidian & VimWiki, might to be excellent platforms for the PIM Productivity system. In such a case, having started with a command-line text interface will facilitate the development of the system as an Obsidian or VimWiki plugin component.

A note on Artificial Intelligence.

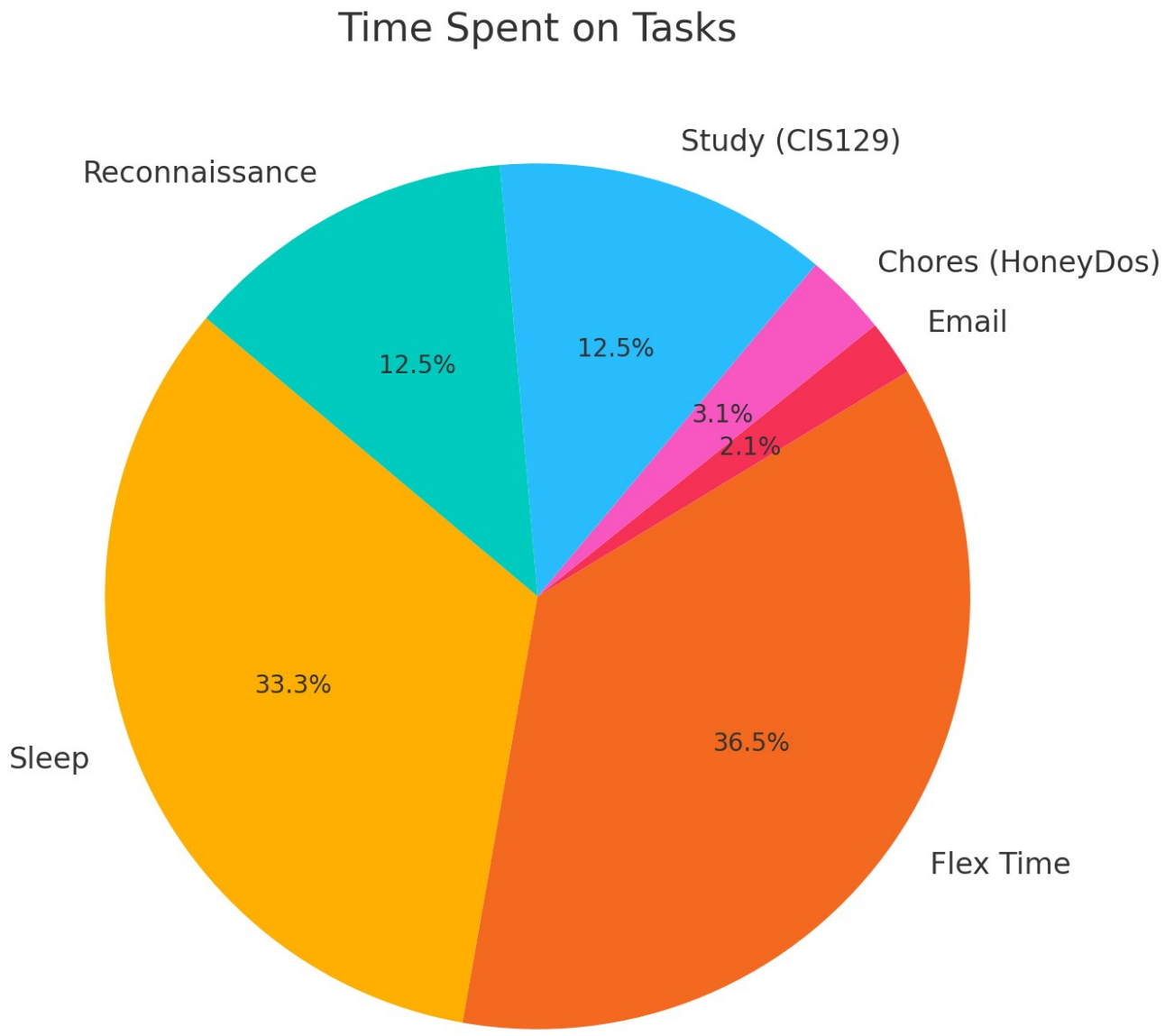
* I need to make a few comments on the how the introduction of Artificial Intelligence is impacting many software applications. Some people claim that the integration of Large Language Models(LLMs) will massively change modern automation. Others claim that AI is massively over-hyped and a devastating bubble burst is just around the corner. Unfortunately, both are true. Customer service in the US is far worse today than it has been since the introduction of the telephone, largely due to improper application of large language model technology. At the same time, many workers will become massively more productive by augmenting there work with assistance from large language models. One of the areas likely to greatly benefit from the application of LLM techology is software development. Microsoft|GitHub co-pilot, CrewAI, & PyCharm, to name

a few, are already making a massive contribution to developer productivity. In this project, I dumped some of my raw data into ChatGPT just to see, using this prompt:

Hello, please analyze the following data and provide me a pie chart depicting how much time was spent on each task...

```
* [X] 00:00 ~ 05:00 {05'00"} |sleep.
* [X] 05:00 ~ 10:00 {05'00"} |flexTime.
* [X] 10:00 ~ 10:30 {00'30"} |email.
….
```

ChatGPT produced this response:



Use case brain storming

* TBD

Footnotes

1. https://en.wikipedia.org/~rmation_manager
2. https://checkpeople.com/blog/personal-information-manager-pim/