**2024-06-13 Compressed Gantt**

1. Agenda

| - share outline  - this presentation is 2 parts : is about project management, but also AI and what still needs our insight and problem solving  - show how i used this tool to better manage projects and timelines  - show table  - show both before and after and how this helped me shift timelines  - talk about what i used AI for: brokenbarh  - but this didnt solve the problem  - talk about the Python, what AI did , what i created, and how i gelled them  - show animation. Also say i used a similar process for creating this although i didnt need to create a hesvy duth algorithm: this was supposed to show this |
| --- |

* 1. Purpose of Project
     1. Rose out of frustration with figuring out how much to work on what, when given various external and internal timelines
     2. Really going from a high level project plan to a weekly and daily plan
     3. Gantt does this but doesn’t give a great visual understanding of demand and effort for each project I thought
     4. So: decided to make something that put “nesting blocks” so I can easily see given project start dates, end dates, and “level\_of\_effort” aka how much effort i’m applying at one time at a given time, roughly hours per day for that time,
     5. How things “look”
     6. And maybe rebalance
     7. I show this to you in case you might find it useful for your purposes, but of course I realize time and task management is individual
  2. Purpose of presentation
     1. I’m going to show you both how I use this
     2. And also the process of creating this from a development perspective
        1. google/stackoverflow
        2. ChatGPT
        3. My own development/insight
     3. The last part to me shows how AI is really still a tool that may increase efficiency, but is not a replacement for us!
  3. Demo and use case
     1. Start by showing spreadsheet
        1. My summer projects
        2. In style of FY25 projects
        3. But with additional “level\_of\_effort”
     2. **Show first sheet**
     3. This is the sheet I started with
        1. Separate lines for each project
        2. Start
        3. End
        4. level
     4. Now I’ll run the script I wrote on this and you can see what this would have done to my schedule!
     5. **Run script on 7A**
     6. You can see as often happens I kind of stacked things up in the beginning, which would have made for a hectic May and June
     7. So I adjusted timelines a little given this, and have this adjusted sheet
     8. **Run script on 7B**
     9. This to me makes a lot more sense
  4. Caveats and takeaways
     1. It’s certainly not perfect
     2. You can see that hole in the first part of the summer
     3. I could adjust things around with more fine tuning to future that out
     4. But the goal was general guidance on what to focus on when given what needed to be accomplished
     5. With as you can see my vacation built in
     6. And now I feel I had a more reasonable, albeit still busy way to approach what I had to do both before and after my vacation this summer
  5. Code - Step 1 - google
     1. Came up with idea
     2. Google solutions
     3. Found a good approach with “broken bar charts” in stackoverflow using matplotlib, which I’m familiar with in general, but did not know abou this structure
     4. **show code 1** “brokenbarh”
     5. Aka a bar chart where each bar may not touch the axis and may be stacked on top of each other, and may not continue where the last left off , unlike a stacked bar chart
  6. Code Step 2 - mockup in ChatGPT
     1. I made an initial mockup, and then pushed it into shape with my requirements in ChatGPT
     2. That gave me something to work with, but it still didn’t *work*
  7. Code step 3 - Step 3 algorithm
     1. But it didn’t work!
     2. Didn’t stack or nest
     3. So that’s where I thought of how to use Python pandas, which is a module I’m familiar working with, for how to do this.
     4. The key was adding each task to a master dataframe in the date range specified, and then “sliding” it up until it “fit” with what was there before
     5. Had to sort first
     6. **Show code 2**
     7. There’s a lot on this screen, but the basic idea is that it sets up a master dataframe to collect all the projects
     8. And then it loops through each of the projects “tasks” successively
     9. Each project is a smaller dataframe that keeps inching up the master dataframe until it “fits” aka without hitting any of the others
     10. Here is an animation that shows roughly how that algorithm works
     11. **Run animate.py**
     12. You can see how this generally works
     13. The point is ChatGPT *could not* get this. Or at least come to it with any possible prompt I could come up with
     14. It points to the idea that since LLMs are predicting what the best completion of a query is, the probabilistic part is guesswork and it feels very much like a one pass first draft in this kind of application
  8. Colors
     1. One more thing I’d note is I find out of the box colors in matplotlib are a little rought to look at
     2. So I used the colormap module of the pyplot matplotlib graphing library
     3. It allows you to specify color schemes by name and pick from them randomly or in a gradient sequence
     4. **Show code 3**
     5. I used this to write some boilerplate, and put
  9. Questions? thoughts