

Planning

Scenario:

My “client” is **Piyush Chandra**. He is currently a sales manager at Amazon who worked there for four years, and he has valuable experience with Artificial Intelligence software and Amazon based tools. As a worker at Amazon, he can provide me with specific guidance regarding the organization of different elements in my application. His advice will aid me in cleaning up my program to make it more efficient. He can also help me in providing recommendations for countries to cover with the naturally broad scope of my project. Since programming an app is a first-time experience for me, his advice will be especially valuable for this and other coding endeavors.

Personally, I have always had the issue of tracking which opportunities I have applied to, specifically the ones via LinkedIn. I also find it cumbersome to constantly update resumes to match the required experience; some projects may seem more valuable than others. I have also found value in learning Amazon Web Services (AWS) tools as well as PyTorch and other utilities that work well with AI. The overall problem comes from the fact that there is no efficient way of tracking the above, and the end goal is to create a project that can manage that.

Solution:

I will utilize AWS and PyTorch to create an application/site that analyses LinkedIn job postings and creates resumes based on both the user’s levels of experience and the preferred demands of the posting. It may also involve generative AI to this end such as ChatGPT to make the resume tailoring more efficient.

Rationale:

Not only does the use of AWS and PyTorch prove to be a learning experience over the summer, it also builds upon my previous experiences in programming and backend implementation. In today’s world, AI conquers the world of efficiency, so the possibility of integrating AWS and other generative formats could prove beneficial for my development process as well as user convenience. Brushing up on PyTorch is also valuable in more ways than not; it allows me to seamlessly add integral functionality to AI-based technology. Therefore, with these three primary tools in mind, it is sufficient to say that this project will be a learning experience and an efficient end product.

Success Criteria:

1. The application must work on a desktop.
2. The user must be able to import LinkedIn job postings via URL.
3. The user must be able to import their experiences, whether it be through importing their resumes or individually adding them.
4. With 2 and 3, the application must create an “ideal” resume that takes the user’s experiences and formats the most preferred ones according to the posting.
5. If a user is lacking in required experience, the application should suggest a possible addition the user could work on (i.e. if experience in Java is required, then learning Java would be the suggestion).
6. As a stretch goal, selected information should be used to remind them of the precautions they must take at their destinations when they arrive.
7. In the event of an invalid link or an error, the application should return with appropriate error handling. It should be apparent that the application supports LinkedIn job postings only.
8. If ChatGPT is used, it should be used to generate descriptions for experiences and summaries for the linked job postings.
9. As a stretch goal, the application should critique the user’s given experiences; if they do not have what is required for the job, then it would respond as such.
10. The application must have a clean user interface, utilizing many key tenets of good UI design such as controlled element placement and efficient menu navigation (if present).
11. The application must be held in a repository in GitHub. This will imply version control and deployment.
12. The project must be done by September 2025. All design concerns and planning must be done by 6/20/25.