Evangeline Kim

CS6620: Cloud Computing

Final Project Proposal Idea

My final project idea is a cloud-based drug interaction checker that can help doctors identify which medication combinations may be dangerous or react differently when taken together. I have a brother with epilepsy, and he currently takes multiple neurological medications daily, which is what inspired this project after I saw firsthand how complex it can be for doctors to quickly verify drug interactions across numerous prescriptions (usually they forget or just have to look up the interactions one at a time). For example, one of the meds my brother was on recently made another one he was on less effective, so not adjusting the other meds properly risked adverse dermatological reactions. The application would ideally allow clinicians to input full patient medication lists and receive real-time interaction warnings with things like severity scores and alternative drug recommendations. The core infrastructure requirements will probably include something for secure API endpoints (AWS API Gateway), Lambda functions running Python code for the interaction analysis, database storage for drug data from sources such as DrugBank (this is a widely used drug database), and maybe S3 for caching frequently accessed combinations. This addresses a real-world issue where many doctors (at least the ones I've experienced) rely on slow or outdated systems that can fail during high-demand periods, potentially missing interactions that contribute to numerous medication-related accidents. The proof-of-concept should demonstrate basic two-drug interaction checking, while the MVP will hopefully support full medication list analysis with some sort of web interface for clinical use (I don't have much front-end experience though), all deployed using AWS and Terraform. I am still not quite sure what is technical feasible at this point (there is still a lot I need to learn), but overall, this is the direction I would like to go for the semester! I do realize that drug interaction checkers already exist, so I am also leaving room for flexibility and scope changes.