Project Name: AI Vulnerability Scanner

* Roles
  + Ian => Backend
  + Alex => Backend
  + Asher => Frontend / DB config
* Tech Stack: React (Tailwind CSS), NestJS, Electron, Postgres
  + DB Tables: Users, Scans (maps to users and a list of vulnerabilities), Vulnerabilities, if its returned as a JSON consider MongoDB
* **To be TRANSFERRED TO GITHUB PROJECTS**
* Week 1 Deadlines
  + Tech stack with hello world
  + Remaining:
    - Login in Backend(Authentication, Saved pass, session mgmt), DB Config (Postgres USE TYPEORM)
  + Asher:
    - Project structure, Home page UI(INCL Navbar), Login in Frontend, Begin UI for input Page
  + Ian: Electron Setup, conditional loading, and building vulnerability scanner functionality
* Week 2 Deadlines
  + Remaining:
    - Create storing system for all data, create fetching class for data
  + Asher:
    - Loading Screen, Front end for results page and main page, AI interpreter start
  + Ian: Creating vulnerability cache, finishing vulnerability scanner (multithreading if possible), link to web app and electron merge
* Week 3 Deadlines
  + Remaining:
    - Chat ui for recommendations, host website and database(ALL), CL/CI(ALL)
  + Asher:
    - Front end for history page, AI interpreter finish / Chat
  + Ian: Interpreter for charts and raw data
* Week 4 everything done **END**
* Login BackendX
* Login Front end and UIX
* Home pageX
* Electron Setup doneX,
* DB Config X
* Project structureX
* Frond ui for drop down menu / NavBarX
* Input fields and front end with suspense for Main page
* Loading screen since it'll take a bit
* Scanner algorithm / API
* Interpreter for data to be find to charts n stuff
* Interpreter for AI recommendations
* Chat UI for something for AI recommendations
* 2X Merge with desktop app on electron
* Create a vulnerability cache
* Link interpreter to scan the linked web application
  + Returning the selected vulnerability test results
* Implement history page, might need caching here since would be a lot of data