Creating your own micro-applications

Steps for running your own app library

Step 1: Identify the problem

Use deep research to identify what are the pain areas in the selected use case. Use experience and research to define how to address these challenges within the scope of the use case.

Step 2: Define the requirement

Define the requirement – both functional and technical. Context is important.
Functional: What features are required
Technical: HTML, css, javascript, tailwind, Node.js, redis, etc. (you can query GenAi for what technologies are required). This will also cover UI features like you want to activate file upload, charts, other UI/UX.

Step 3: Generate the prompt

Use GenAI to generate a detailed prompt to create an application for the defined requirement. Use key words like professional, comprehensive, relevant, etc.
Define the technical things it should use – css for styling, use Deloitte color theme, javascript, Three.js, etc.

Step 4: Generate the code and Review

Feed the prompt in GenAI to generate the code – Python back end, HTML front end. Depending on the GenAI model that you use, the context window may need a few iterations if the code is complex.
Ask GenAI to generate Readme.MD, .env, Requirements.txt, file / folder structure, etc.

Review the code – apply your Python code experience, to see if this is what you want.

Step 5: Run the code

Create a folder structure, setup environment, save the code files. Install the requirements. Run the code. If the code does not run the first time, read the error messages – these can be of two types

- dependent package / library missing – install the package and run again
- some other error – Feed it back to the GenAI, update the code with the suggested change and run again

Step 6: Test, iterate and optimize

Sample data for testing:

- Option 1: Search on the internet for available datasets (Kaggle, etc.)
- Option 2: Generate your own data through steps 2,3,4,5

Test the code, whether it serves Step 1 and 2 as intended.
Optimize the code by adding sophistication to parts of the code. Ask GenAI itself about ways in which you can improve the code and then iterate.