Backend code plan:
Iterate through the following steps to capture all requirements, and corner cases. Try to follow first things first principle. First, focus on core simple use case and then keep on adding common variations from simple use case you are likely to see.

1. schema
   Think about the state you want to capture and store. This could include maintaining information about raw data received from a source, data that has been assigned for processing, ie work-in-progress, data that has been processed

2. write the process flow in words
   Example,
   if value of variable v is x
   do some process
   if value of variable w is z
   do some process
   ….
   else
   do some process
   next, do some process
   …
   else if value of v is y
   do some process
   …
   else
   …

3. identify process names,
   Use descriptive names for processes you identified earlier.

4. Define data inputs to processes, and data outputs of processes
   When specifying data only specify key names in data object eg:
   {users:[{_id, name, …}]}

`https://api.github.com/search/repositories?q=${language} language:python created:${from}..${to}&sort=stars&order=desc&per_page=100&page=1`

Job roles: Data Engineer, Cloud Engineer, DevOps, ML Engineer, Data Scientist, Data Analyst, Business Intelligence

Repo Types:

- Foundation libraries/frameworks - Widely used, community maintained reusable code providing core functionality. Examples are React, Express, jQuery.
- Focused utilities/modules - key attributes are focused on solving one specific problem, small number of maintainers, and promotion of integration into other codebases. Examples are Lodash, Moment, Chalk, Dotenv, PM2
- End-user applications - Applications focused on specific user needs, problems or workflows rather than provide general purpose functionality.
- Tutorials/learning resources - Example code meant for teaching/learning rather than production use.
- Research paper code - Code meant to share and validate research results rather than use in production.
- Datasets - Repositories holding datasets rather than code.
- Templates/Boilerplates - Provides a starting point, scaffolds or structure for new projects.
- Miscellaneous