GIT HUB DEMO OUTLINE

**Summary** – I am working on understanding the core of GitHub. My experience has been previously mainly around Azure DevOps and formally known as VSTFS

As part of my current role, I have been tasked with starting to understand how to replace the MS git with GitHub Enterprise.

I decided to be super transparent about this and use commits to see my awful thinking process.

I need this to be MVP (Minimal Viable Product), define the scope and make sure yes, it is perfect, you can make it perfect if you get the job 😊

Using VSCode for my “Editor” it seems to work just like Azure DevOps Repro’s go figure they are based in GIT, well unless you remember TFVC (yes, I am that old)

So, this work does both.

Mural Link – This is my idea board (love this tool) chaos mind organized!! <https://app.mural.co/t/becu7324/m/becu7324/1653438022088/8eb3686617b8d78b99442a5d1b9f5de12177622a?sender=uaf3a2c608df1ed4236795066>

**Tools used**

* VS Code
* Word
* GitHub Desktop

**Customer Key Point** –Refer to mural but the mission statement should be what the customer wants. reason it is top.

* Scale is the issue
* Rolling out in “Bulk” seems to be concern
* Setting up structure
  + Branch policies
  + I would assume that the standard GIT Flow (insert link)
  + Event notifications? Are they required and at what level?
    - New branch
    - PR into main or other constructions (development)
* Scanning – Code Analysis
  + What happens?
    - Event trigger or policy that says “hey I won't commit until that passes X”
    - Email notification
  + Protected branches (Non shall pass)

**GitHub Points:**

* I feel like they want me to understand and demonstrate understanding of organization.
* Coder Owners? - policies around best practices
  + How do you apply that to the organization?
    - Per repro?
* Organization Events
  + How does that relate to the check in etc (WebHooks)
  + Pull requests
  + Issues tracking?
    - Define that like actions fixed verses reference (feel like this isnt MVP)
* Authorization – how does this link to other tools 😊
  + This might be more of my current roles as azure active directory and github last I read was clunk – how do others do this?
* API (Application Programming Interface) – stress the power of API (we all know that API are the king)
  + Organization events was the email stress point
  + Others seem as important (explore that and list)
* Documentation - I like two types of detailed and quick engineering references (make both)
* Core Values – I read them, and I like them they align with “Product” first – make sure reference are inserted especially to customer presentation -
* I feel like extra points would be using GIT hub actions explore that option.