Questions for PEAK

1. Can we record meetings - yep
2. When is good to meet for you
   1. Every 2 weeks ideal
   2. They’re free Wednesday afternoons
   3. Boyd Orr 720 at 3pm
3. How long and frequent should meetings be
4. How often should we report back outside of meetings? Frequent communication will be mutually beneficial
   1. ‘probably worthwhile’
   2. want to be told what they actually *need*  to know
   3. ask questions (teams, email) when we need to, don’t wait til next meeting
5. Have you worked with students before, how did it go? Any recommendations?
   1. No
6. How to communicate and who is allocated emailer
   1. Emails
   2. GitLab
   3. Teams; prefer to be informal – ok to ask lots of little questions, we would prefer teams for informality
7. Add to GitLab
8. Need to decide if we are doing public or private repo
9. Minimum viable product
   1. At least: storage aspect in standalone container, the predictions in standalone container
   2. Loading far more structures automatically, scaling it all up
   3. Priority runs: Protein storage, alphafold, stretch work/scalability
   4. Focus on the storage first
   5. Gonna be complex, may need helper services eg databases for the storage making scaling easier later
   6. Eg want protein x, doesn’t exist so get a prediction, request it again don’t want to wait for another prediction (can take 24 hours), prevent duplicate queueing
   7. External data will be updated outside of our control, and changes will have to be updated into our own software.
10. How should it looks in terms of UI and branding, examples would be good
    1. No UI required, just API backend, maybe make a small python script to create command lines to provide input.
    2. Picking the structures currently done by hand (10s of proteins or random selection)
    3. Scale to run virtual screening, select more specifically and far more (1000s, 10s of 1000s)
11. What is the front end going to look like? How are we receiving the protein requests? How are people adding proteins?
12. No front end needed
13. Input is just a text string with the protein sequence
14. Can you give us some user stories for the product, who is going to be using it
    1. Someone already reasonably competent programmer interfacing it for their own computation experiments
    2. More non-technical people can use it, may use command line; first one primary user, would be happy writing their own scripts
15. Approx. how large is the database going to be
16. We have the database links on the handout.
17. A Kubernetes cluster appears to be a difficult thing to set up and maintain, can you explain how you expect us to use this with your system
    1. Easier to add other stuff to one we create; no pre-existing one
    2. Everything in containers avoiding conflicts when working with scientific data/code
    3. Start with docker containers, happy if we get to running those, maintenance not massive until transitioning to Kubernetes; if we get to that point will be provided with a cluster to work with/practice on
    4. **Start with docker, if can scale to above one copy then can scale to many,** kubernetes then for abstracting
    5. Not expected to start from scratch
    6. Command line tool is available to transition Docker containers to Kubernetes cluster
    7. Keep microservices stateless so that multiple can be deployed to increase scalability
18. What do you expect to see (prototype) by the next time we see you, what should we accomplish in the first sprint?
    1. User requirements specification
       1. Can be sent in advance
       2. Can be working copy as things clarified
       3. Everything on GitLab wiki for marking
    2. If we decide on fast API (they’re most comfortable with), we’ve done the intro for that and docker (‘hello world’ type stuff); a lot of automation already in it eg condition checking, need to learn
    3. Play with returning a file, serving the file content to the request
    4. Expecting/fine with questions about fastAPI, docs are good
    5. Worst case can store docker containers using git clone, can push to public docker hub; need to look in o what we have access to for storing the containers
19. How do we filter out the ‘junk’ proteins from protein data bank without knowledge of that field
    1. Based on criteria provided
    2. Eg sum waiting to be put on the method
       1. Required ‘good’ values for xray/resolution quality etc, return result and decide from there
    3. Extract from API with some parameters and adjust
20. How do we decide which is the ‘best’ protein prediction to select from the Alpha Fold Database
    1. Weighting proteins based on factors like method, coverage, and resolution
    2. Have a program that spits out a number and selects the best one.
21. Can you elaborate on what happens when a protein structure needs to be predicted. You mention this is resource intensive but would this be handled by us? Mentioned a request goes to another group
    1. Running on one VM
    2. Protein sequences are strings, one letter = one amino acid
    3. Passed to alphafold service which then makes the prediction; min viable on our end, dig into how afold works and see if it can be split out/speed up (currently can take a week)
    4. 25 predictions one after the other, queues up ranks them spits out the best
    5. Can possibly look into AlphaFold technicalities to find out how to make it run on more threads, etc.
22. Do you want to search database for individual structures or search based on criteria to return multiple results
    1. Always going to be exact match expected
    2. Once you get a result store it in so you don’t need to predict for next time.
23. Any monetary incentive?
    1. No :)
24. Licensing of the app
    1. Open source; MIT simplest (they prefer)
    2. Copy on our private that we release at the end **or** public github that are showed on the go, show updates and progress and mirror onto gitlab; up to us

GENERAL MINUTES:

* PEAK – support company, biotech research, lots of diff companies mainly startups
* Mainly REST API microservices; NO WEB UI EXPECTED
  + Need API docs to interact; FastAPI good resource
* Public API; no logins, security, no personal data just protein structures
* PSS –
  + Protein databank
  + alphaFold protein structure database
  + other microservices
  + user upload
  + no transformations needed to be done to content by us
* UniProt
  + If someone looking for certain protein need to find structure to download and stash properly
  + Wanting to ‘pick the best one’
    - Certain methods more specific than others; eg lower resolution better, structure method, sequence coverage; EXPECTED TO BE AUTOMATED TO CHOOSE BEST DEPENDING ON HEURISTICS OF INDIVIDUAL PROTEINS
  + Expected to return URL for the protein you want
* When protein doesn’t exist, current prediction takes a while; long proteins take up a lot of memory to run predictions depending on speed
* No preference for in person or online just need to book room and organise ourselves Wednesday afternoon
* **3pm 18/10 next meeting**