Meeting 1:

TODO: This should be simple, since we were just going over the very basics of the project.

Meeting 2 (February 4):

Most of us were able to get vllm running, with some experiencing some technical difficulties. There appeared to be some special complications when trying to run the GPU version.

* Need permissions for HumanSignal
  + <https://humansignal.com/>
  + Label studio needs some kind of “organization” authorization, which we do not currently have. This made it impossible to experiment with the technology.
    - This can be resolved *after* installing it?
    - Carter created a “team”:
    - Having troubles having team members to join the group created by Dr. Phan.
* The next time we need to make a round of validating the data (around 90 translation pairs).
  + Look at input code and output code provided by the different models.
  + No outside documentation needed, just provide your explanation in the “Explanation of Human Judgment” section
* The main goal of this project is to build the principles for each programming language. This “play around” section should help us setting the tone for the rest of the project.
* We need to define a principle for our programming language

Final deliverable is defining principle that improves LLM. Also figure out which LLM is best for different type of judgment (different coding principles)?

Meeting 3 (February 12th):

* Input code with prompt and the application will return multiple code segments
* Code search will compare with vector comparison to find closest code candidate to natural language (using existing tool to generate code), need to perform traditional code search
* Output of traditional code search will have 10 code candidates, how do you integrate LLM?
* The most straightforward approach to integrate LLM is to provide context and ask LLM to evaluate short list of code to me and find best code candidate, will rank list of code
* Define criteria for best candidate (principle), still have to define principles
* Final output to end user is re-ranked list of code
* No technical challenge because you can easily replicate, he already ran code search process, using linked UniXcoder
  + run.py
* Main contribution is integration of LLM
* First round of application, ask model to re-rank given short list of code
* CodeSearchNet dataset to make vector database?
  + So it is pulling code snippets from here and not generating the code
* Next 2 weeks, must have first version of application ready
  + Don’t have to have GUI ready right away
* Demo will cover (mostly) everything from chart