**Report**

**Framework Review:**

* I have used CrewAI for building AI agents although the framework fastens the development process it is slow at inference.
* Needs to be replaced by faster frameworks if we need to develop an application ready for production. Can be used for testing and faster development.
* CrewAI has very less customizability as it has built in frameworks. Custom tools can be introduced though.
* I wanted to incorporate a web search through RAG but keeping inference time in mind I did not carry forward with the implementation.
* *While parallelizing certain processes can improve execution time, I also want to run the resource collection process in parallel. By utilizing manager agents, resource collection can be interconnected across platforms to gather better resources and datasets for a specific use case.*
* *Langraph, Autogen can be the next best choices requiring a bit more development time for customizing tools and agents.*

**Answer Relevance and Quality** :

* The use case generation works like a charm and a correct process (a step-b-step guide) for generation might improve the process. But they are still at ideation stage (Primary).
* As deeper search through website content was not implemented, it’s implementation might increase the analysis and provide a detailed report in all perspectives to implement the use case .
* This would result in a detailed search for repositories and datasets within a single use case and for implementing by allowing delegation around resource collection tasks.
* The links and datasets provided are mostly related to the proposed use case, but they are direct search results. Sometimes a search query might pick up an irrelevant link due to confusion caused due to the keywords.
* There must be more intention to searching resources rather than keywords, usecase generated can be broken down to further tasks as a general AI project. (This will in turn increase calls and increase inference time, parallelism might help here).

Agent Framework