Finalized Project Topic Proposal:

Seed Identifier*:* A Python-Based; AI Driven Object Recognition Application

Summary:

This project involves developing an application designed to accurately identify various types of seeds using image recognition techniques. Python, SQL, and local image directories will be used. At the advisement of the professor, PyTorch will be employed to deliver the ID engine. Additional training processes would be required to further narrow down and improve search functionality. The application will be equipped with a user-friendly GUI that allows users to upload images of seeds from their local computers. The core functionality of the application will revolve around identifying the seeds and providing a weighted identification label based on trained models.

Key Features:

Image Processing and Object Recognition:

- Utilize machine learning models trained on a dataset of seed images to recognize and classify different types of seeds.

- Implement algorithms to enhance image quality and preprocessing to improve identification accuracy.

Weighted Identification:

- Assign weights of accuracy, reflecting the confidence level of the model's prediction, for each item for the user to determine if its dependable information.

Graphical User Interface (GUI):

- Design a simple and intuitive GUI using a Python framework –utilizing Tkinter to enable the users to easily upload images and view identification results.

Local Computation:

- The application will be designed to run on a local computer, leveraging local processing power to handle image recognition tasks. A potential challenge would be to utilize GPU over local memory.

Optional Online Hosting (Time Permitting):

- Explore the possibility of hosting the application online, allowing users to upload images through a web interface and receive identification results remotely.

Challenges and Considerations:

- **Processing Speed:** Optimize the application to ensure that identification is performed efficiently on local machines, considering possible hardware limitations. If run on a local machine GPU would offer more power and better speeds that local memory.
- **Training Methods:** Select appropriate models and training techniques that balance accuracy and computational efficiency.
- **Error Handling:** Implement strategies to manage and reduce identification errors, such as improving model training, preprocessing techniques, or providing fallback options in the GUI.