Important Instructions

1. **Documentation:**

The project you choose must have good documentation throughout. In the initial stages, this is very useful.

2. **Should be in your skill set:**

You should be good with the technologies that the project uses - language, frameworks etc. If not all, at least you should be familiar with the language. The rest, you can learn as you work on the project.

3. **Well established support system:**

The process of working with the code should be easy and not be a bumpy ride. It means the entire process - compiling, modifying, reproducing bugs, testing and deploying - should be easy.

4. **Easy deployment:**

Select something that is easy to deploy after making code changes. That is, it should have an output that you can directly see (instead of having to check logs). You should be able to see the changes you make directly (as opposed to performance improvement changes that you can directly see as an output)

5. **Contribute to something you use regularly:**

Choose a project whose software you regularly use and care about. This will make it easier to reproduce bugs since you know the way around the system. During your regular usage, you might notice new bugs and/or think of new features and you'll have the motivation to build it since you need it.