**Project Phase 1:**

* Initialized git repository
* Created a root directory called career-nexus and added a pom.xml file using the command: *mvn archetype:generate -DgroupId=com.careernexus -DartifactId=career-nexus -DarchetypeArtifactId=maven-archetype-quickstart -DinteractiveMode=false*
* Deleted the src folder as it is not needed.
* Modified pom.xml to include java version-21, spring boot version 3.5.6
* Created spring boot projects for each microservice:
  1. gateway-service
  2. user-service
  3. resume-service
  4. job-matcher-service
  5. learning-suggester-service
  6. common-lib
* Made the pom.xml file in career-nesus (root) the parent so the other microservices inherit the spring boot version, and java jdk version and other configurations.
* common-lib is a blueprint for DTOs, utils and exceptions so it can be used throughout the project.
* Reload maven only in root pom.xml, even if you make any changes in pom.xml files in any of the microservices, it’ll automatically reload all the microservices.
* Created a Dockerfile in each microservice each assigned a specific port, and the port 8080 is assigned for gateway-service.
* Created a docker-compose.yml file at the root directory and laid down the configurations for MySQL and all other services.
* **Suggestion in docker-compose.yml:** 
  1. Add startup retry inside services via Spring Retry or connection pool config to handle transient DB readiness issues beyond Docker healthcheck.
  2. Single DB for dev/testing is common; for production, ideally separate DB/schema per microservice for better autonomy and scalability.
* Created environment variables for docker-compose.yml, in the root directory.
* Created environment variables for each service specifically.
* Shifting the containerization (docker) to the end of the project for faster development, shifting to local machine instead of docker containers.
* Created a compound run in run configurations so all services can be run at once.