**Exam Question Paper Template**

**Course: Software Engineering Lab Exam: Internal– Practical**

**Duration: 5hrs Max Marks: 100Q1. Problem Statement Analysis – 10 Marks**

**Task:** Students will be given a problem statement. Based on it, write the following:

1. **Abstract** – Summarize the problem in your own words
2. **Functional Requirements** –
3. **Non-Functional Requirements** –
4. **users**

**Q2. Maven Project Building -- 30 Marks**

**Task:** You will be provided with a project folder. Perform the following steps:

1. Import the project into Maven environment

|  |
| --- |
| Will give steps how import it  **Open Eclipse**   * Launch **Eclipse IDE** (preferably Eclipse IDE for Enterprise Java Developers).   **2. Clone Repository**  Go to the menu: **File → Import → Git → Projects from Git (with smart import)** → Next.  Choose **Clone URI** → Next.  Enter the **GitHub repository URL** (SSH or HTTPS):   * + Example (HTTPS):   + https://github.com/username/repository.git   Select the **branches** you want (usually main or master) → Next.  Choose the **directory** where Eclipse will store the repo → Finish. |

1. Resolve dependencies using pom.xml
2. Build the project to generate the **WAR/JAR file**
3. Verify the generated artifact in the target/ folder

**Q3. Git and GitHub -- 30 Marks**

**Task:** Work with Git and GitHub as follows:

1. Initialize a Git repository and add your project files
2. Set global config
3. Solve the given **SQB (Short Question Based)** Git tasks (e.g., branch, merge, revert) Push your Maven project to GitHub

**Q4. Docker – 20 Marks**

**Task:** Containerize your Maven project using Docker.

1. **Dockerfile Creation**

|  |
| --- |
| * + Refer the given link/documentation to write a Dockerfile for the Maven project.   + <https://github.com/archanareddyse/labinternal-1.git> |

* + Ensure it copies the WAR/JAR and runs on Tomcat (or relevant base image).

1. **Image Building** 
   * Build the Docker image using docker build -t <image\_name> .
2. **Push to Docker Hub** 
   * Tag and push the created image to your Docker Hub account.
3. **SQB** 
   * Answer short questions related to Docker commands, image vs. container, etc.

**Q5. Docker Compose -- 10 Marks**

**Task:** Write a docker-compose.yml file for a multi-container setup.

1. **Container 1** – Use the Docker image you pushed to Docker Hub
2. **Container 2** – Configure a database container (MySQL/PostgreSQL/MongoDBEnsure both containers are running together
3. Demonstrate service startup with docker-compose up

**Note to Students:**

* All steps must be executed **practically**.
* Marks will be awarded for correctness, clarity, and proper usage of tools.