**Project - XXX**

**Created By - YYY**

**Date - ZZZ**

Contents

[1. Objectives of this document 1](#_Toc18589520)

[2. “How to build Java Full Stack Applications” 1](#_Toc18589521)

[2.1 Overview 1](#_Toc18589522)

[2.2 Know your workspace 2](#_Toc18589523)

[3. Project Overview 4](#_Toc18589524)

[4. Scope 4](#_Toc18589525)

[5. Not in Scope 4](#_Toc18589526)

[6. Technology Stack 4](#_Toc18589527)

[7. Data model 4](#_Toc18589528)

[8. Design Choices 4](#_Toc18589529)

[9. Other Details on Solution 4](#_Toc18589530)

[10. Screen Snapshots 4](#_Toc18589531)

[11. How to Start the App from My Workspace 4](#_Toc18589532)

[12. Assumptions 4](#_Toc18589533)

## Objectives of this document

1. **Section 2 of this document would help the developer using the Cloud based IDE (opened from the exam journey screen) understand the workspace environment that has the IDE spawned. It further would tell the Developer about the configuration details on Tomcat 8 & MySQL and enable him to know how to deploy the full stack app into Tomcat while pointing it to the MySQL database.**
2. Rest of the sections in this document are included to stand as a documentation template on the project case study. Developer is expected to fill the relevant sections after completing the assignment. Once filled, the developer can upload the same document to be used by Reviewer for further verification.

## “How to build Java Full Stack Applications”

### 2.1 Overview

A sample template code for Java applications is already available in the workspace in the form of a Maven based project structure.

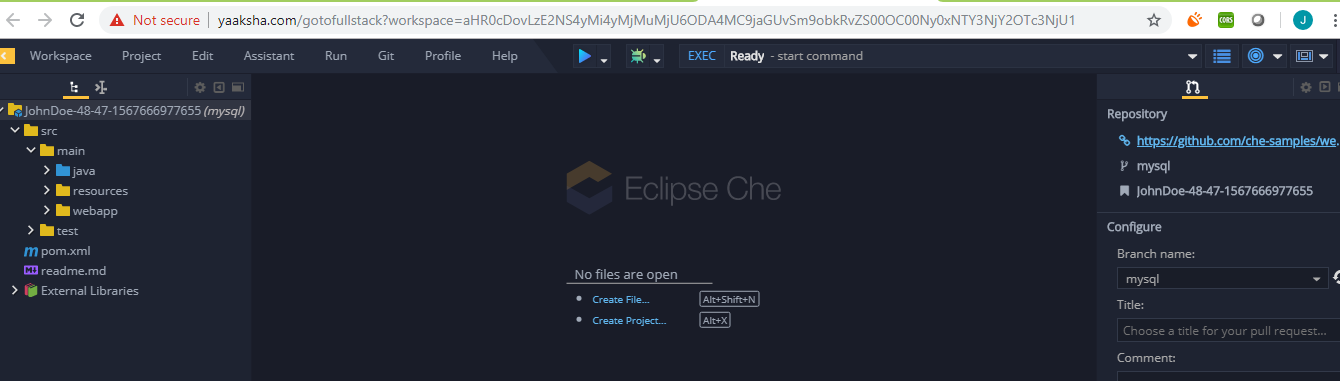
One can build a full stack Java application by coding -

1. Html(s)/Css/JavaScript/JQuery (or for that matter any front end library including Angular jS or ReactJS. Typically a front end technology stack to be used for the application may be provided as a part of project specifications. If yes, one may use the same.
2. Spring/Google Guice/JPA/JAXRS based middleware. Typically a middleware technology stack to be used for the application may be provided as a part of project specifications. If yes, one may use the same.
3. Use MySQL as a database

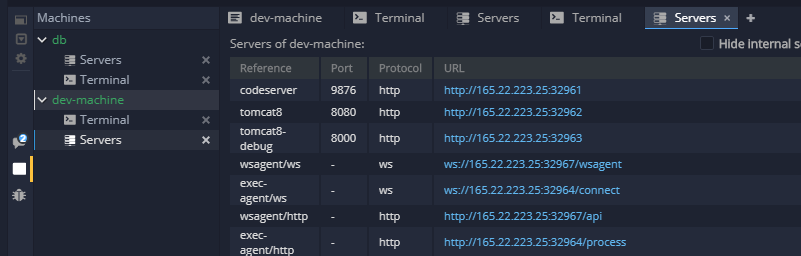
### 2.2 Know your workspace

1. Click on the link ‘Open Code IDE in new Window’ from your exam screen.

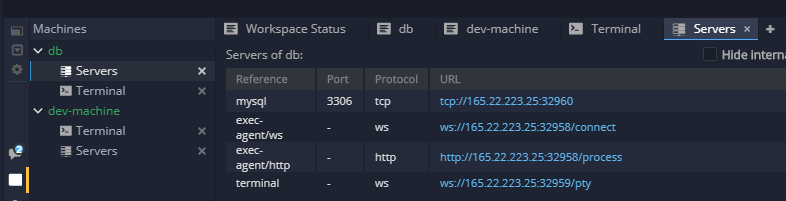
You will see a cloud based IDE open up in your browser – something like this –



1. A private workspace corresponding to a user is spawned at a run time. There are 2 virtual machines contained within this workspace – a DB machine and a Dev Machine.



1. Dev Machine contains an embedded Tomcat server while the DB machine contains an embedded MySQL server**. From above –32962 is the port where your web server will listen for http requests when started. For more details see point ‘h’ below**
2. You can right click on ‘db’, as seen above, and click on Servers to the ports that are available for the MySQL database.



From above the MySQL port is 32960. This is a port that was available and open at the time of spawning the workspace. **It is quite possible that if you spawn this workspace later you may get a different port that may be available then.**

1. Here are the default MySQL db access details

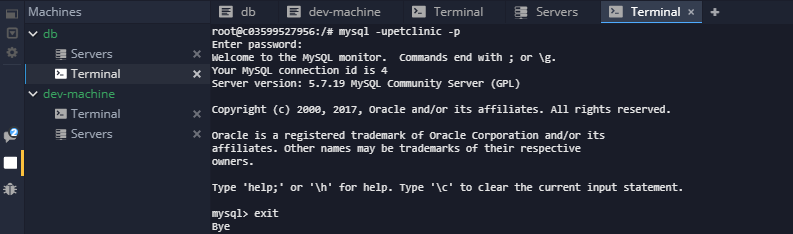
**Database – petclinic**

**User – petclinic**

**Password – password**

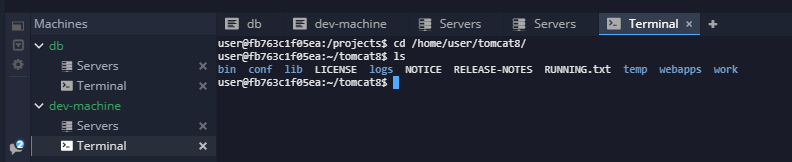
**(From above the MySQL url is - mysql://** **165.22.223.25:** 32960**/petclinic )**

1. To connect to database from command prompt, right click on ‘db’ as seen above and click on Terminal
2. Then you can use the regular MySQL commands to connect to the database



You can connect to the database as well from outside using your favourite MySQL editor. – Just remember that the port may change every time you launch a new session.

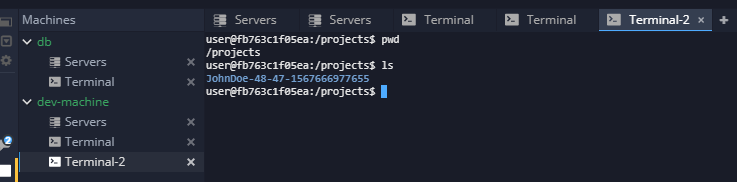
1. To access the location of your web server, right click ‘dev-machine’ (See above) and then click on Terminal. Use a simple change directory command to go to following location - **/home/user/tomcat8**



You can manually deploy a war file into the webapps folder or configure the location of Tomcat in your Maven Build and Deploy script – that’s up to you.

Once you start your tomcat, it can be accessed by following url - [**http://165.22.223.25:32962/**](http://165.22.223.25:32962/)**${ProjectName} (See point ‘c’ above)**

1. Your code base – Open a terminal in your Dev Machine. –



You are by default In your parent project location. Change directory into your project folder and you can start compiling using Maven

Note – Steps for “How to auto-deploy “ into Tomcat after a Maven build has to be added by the developer, if the need be.

## Project Overview

## Scope

## Not in Scope

## Technology Stack

## Data model

## Design Choices

## Other Details on Solution

## Screen Snapshots

## How to Start the App from My Workspace

## Assumptions