



GRAND CANYON
UNIVERSITY™

CST-323 Activity Guide

Contents

Activity 1: Design Cloud Test Application & Cloud Research	2
Activity 2: Build Cloud Test Application and Cloud Research.....	3
Activity 3: Deploy Application to Cloud and Cloud Research Part 1	5
Activity 4: Deploy Application to Cloud and Cloud Research Part 2	6
Activity 5: Cloud Containers and Cloud Research	7
Activity 6: DevOps and Cloud Research	9
Activity 7: Future Cloud Computing and Cloud Research.....	12



Activity 1: Design Cloud Test Application & Cloud Research

Overview

In this activity, students will:

1. Get an Azure Student Account.
2. Complete the initial design and proposal to for building a test application using the Java Spring Framework (or Spring Boot), Angular/React/Express, .NET MVC Core, PHP, or PHP Laravel that will be used in future activities. You **MUST** already know the framework for the chosen web framework used for the test application.
3. Complete cloud computing research on the basic concepts of cloud computing.

Execution

Execute this activity according to the following guidelines:

1. Get a Microsoft Azure account (this will be used in the Topic 3 and 4 activities):
 - a. Request a Student Azure account by using the instructions outlined in the Student Success Center (search for Cloud Hosting, go to the GCU Cloud Hosting Solutions, and select the Microsoft Azure option).
 - b. Complete the Azure Registration process.
 - c. Ensure that you sign into the Azure Portal.
 - d. Take a screenshot of the Azure Portal once you are able to log in to your account.
2. Start cloud test application.
 - a. You will design and build a test application that will be used in future topics to test and validate a number of cloud platforms.
 - b. The goal of the test application is NOT to build a complicated, feature-rich application, but to build an application that can be easily deployed onto, and validated on, a number of cloud platforms.
 - c. The test application must meet the following requirements:
 - i. Designed and built using the test application using the chosen web framework.
 - ii. Designed and built using the MySQL database.
 - iii. Designed and built to include three to four pages that include a combination of pages to enter user data via forms and display user data.
 - iv. Designed to support all the database CRUD methods.
 - v. Designed using Bootstrap.
 - vi. Developed using a GIT repository.
 - vii. Demonstrate functionality in a screencast.



GRAND CANYON UNIVERSITY™

3. Cloud computing research:

- a. Read the required textbook readings assigned for this topic. In 75–100 words, describe the evolution of cloud computing.
- b. Pick one of the case studies from Chapter 1 of the textbook. What are the advantages and disadvantages the company encountered when moving their applications and data to the cloud? Provide three advantages and three disadvantages, along with details regarding any challenges the company faced. In addition, provide a list of three features the company was able to take advantage of in the cloud.
- c. Analyze the Cloud vs. On Premise Server Prototyping Example in Chapter 1 of the textbook. Select either the Cloud or On-Premise solution that could be used to deploy your company's business applications. Provide a detailed (100 words minimum) rationale on your recommendation and justification for your solution choice.

Documentation

Submit the following to the Learning Management System:

An activity report documenting the progress of the completion for the following application components:

1. Cover sheet
2. A screenshot of being logged into the Azure Portal
3. The framework and technology chosen for the test application
4. The progress and status on the database designed for the test application, using an ER diagram, as well as outlining what tables have been built and what tables remain to be built
5. The progress and status on the test application development, outlining what pages and services have been built, as well as what pages and services remain to be built
6. A list of current ongoing issues that are hampering the completion of the test application
7. A screencast URL demonstration of the test application functionality
8. The cloud computing research questions

Activity 2: Build Cloud Test Application and Cloud Research

Overview

In this activity, students will:

1. Design and build a Java Spring Framework (or Spring Boot), Angular/React/Express, .NET MVC Core, PHP, or PHP Laravel test application to be used in future activities. You **MUST** already know the framework for the chosen web framework used for the test application.
2. Complete cloud computing research on the cloud deployment models and cloud service models.



GRAND CANYON UNIVERSITY™

Execution

Execute this activity according to the following guidelines:

1. Finish Cloud Test Application.
 - a. You will design and build a test application that will be used in future topics to test and validate a number of cloud platforms.
 - b. The goal of the test application is NOT to build a complicated, feature-rich application, but to build an application that can be easily deployed onto, and validated on, a number of cloud platforms.
 - c. The test application must meet the following requirements:
 - i. Designed and built using the chosen web framework.
 - ii. Designed and built using the MySQL database.
 - iii. Designed and built to include three to four pages that includes a combination of pages to enter user data into forms and display user data.
 - iv. Designed to support all the database CRUD methods.
 - v. Designed using Bootstrap.
 - vi. Developed using a GIT repository.
 - vii. Demonstrate functionality in a screencast.
2. Cloud Computing Research:
 - a. Analyze each of the cloud deployment models (public cloud model, private cloud model, and the hybrid cloud model). Provide three advantages and three disadvantages for using each cloud deployment model. Explain your rationale.
 - b. Identify two SaaS applications. Provide three advantages and three disadvantages to the identified application when compared to building and hosting those same applications yourself. Explain your rationale.
 - c. Read the required textbook readings assigned for this topic. From an application developer's perspective, what are three primary differences between using a PaaS Cloud Server versus an IaaS Cloud Server for your environment? Explain your rationale.

Documentation

Submit the following to the Learning Management System:

An activity report documenting the progress of the completion for the following application components:

1. Cover sheet
2. Updated test application database and application design
3. A URL of the screencast demonstration of the running test application



4. Cloud computing research questions
5. A zip file of the code for the cloud test application

Activity 3: Deploy Application to Cloud and Cloud Research Part 1

Overview

In this activity, students will:

1. Deploy the test application onto the Microsoft Azure, Heroku, and the Red Hat OpenShift cloud platforms (time permitting).
2. Complete cloud computing research on the cloud deployment best practices and cloud platform feature comparison.

Execution

Execute this activity according to the following guidelines:

1. Cloud platform deployment. Complete the following for the Microsoft Azure, Heroku, and OpenShift cloud platforms. It should be noted that OpenShift will only be used if time permits and based on instructor direction. Refer to Parts 1, 2, and 3 within the "How to Guide," located in the Course Materials, for Cloud Platform deployment instructions. For each cloud platform:
 - a. Create an account on the cloud platform.
 - b. Research the container(s) and service(s) available on the cloud platform that will be required to support your test application. Provision the necessary container(s) and service(s) required to deploy your application and database.
 - c. Configure the database using your DDL script.
 - d. Deploy your application code.
 - e. Test your application and database.
 - f. Prepare a screencast demonstrating all the functionality of your test application running on the cloud platform.
2. Cloud Computing Research:
 - a. Read the required textbook readings assigned for this topic. Identify eight "worst practices" when migrating your application to the cloud. Identify the worst practice in two to three sentences, then provide three to five sentences on how to prevent the worst practice from becoming a risk or issue during an application cloud migration.
 - b. Compare the cloud features of Microsoft Azure and Heroku. Present at least 10 features, explaining how they are similar and/or different. Explain your rationale.
 - c. Compare the cloud features of Microsoft Azure and Red Hat OpenShift. Present at least 10 features, explaining how they are similar and/or different. Explain your rationale. This only needs to be completed if the Red Hat OpenShift Cloud Platform was in scope of the activity based on instructor direction.



GRAND CANYON UNIVERSITY™

Documentation

Submit the following to the Learning Management System:

An activity report that contains the following on Cloud Application Deployment and Cloud Computing:

1. Cover sheet
2. Final database and application design
3. Cloud deployment report for each cloud platform, containing the following information:
 - a. Step-by-step instructions used to configure and deploy the test application
 - b. Challenges encountered during the database and application deployment
 - c. A screencast of the test application being run on the cloud platform
4. Cloud Computing Research questions.

Activity 4: Deploy Application to Cloud and Cloud Research Part 2

Overview

In this activity, students will:

1. Deploy the test application onto the Amazon AWS and the Google Cloud platforms.
2. Complete cloud computing research on REST API solution design and cloud platform feature comparison.

Execution

Execute this activity according to the following guidelines:

1. Cloud platform deployment. Complete the following for the AWS and Google Cloud platforms. Refer to Parts 4 and 5 within the "How to Guide," located in the Course Materials, for cloud platform deployment instructions. For each cloud platform:
 - a. Create an account on the cloud platform.
 - b. Research the container(s) and service(s) available on the cloud platform that will be required to support your test application. Provision the necessary container(s) and service(s) required to deploy your application and database.
 - c. Configure the database using your DDL script.
 - d. Deploy your application code.
 - e. Test your application and database.
 - f. Prepare a screencast demonstrating all the functionality in your test application running on the cloud platform.
2. Cloud Computing Research:
 - a. Read the required textbook readings assigned for this topic. Let's say you decide to leverage (i.e., consume) a REST API that is published from a social media website,



GRAND CANYON UNIVERSITY™

- such as Facebook or Twitter. What are five relevant technical questions related to nonfunctional requirements and security that would you need to ask the vendor?
- Compare the cloud features of Amazon AWS and Google Cloud. Present at least 10 features, explaining how they are similar and/or different. Explain your rationale.
 - Identify five technical or business limitations that could restrict an existing application from being deployed onto any of the Cloud Platforms that were utilized in Topic 3 or Topic 4. Explain your rationale.

Documentation

Submit the following to the Learning Management System:

An activity report that contains the following on cloud application Deployment and cloud computing:

- Cover sheet
- Cloud deployment report for each cloud platform containing the following information:
 - Step-by-step instructions used to configure and deploy the test application
 - Challenges encountered during the database and application deployment
 - A screenshot of the current billing report for the cloud platform
 - A screencast of the test application being run on the cloud platform
- Cloud computing research questions

Activity 5: Cloud Containers and Cloud Research

Overview

In this activity, students will:

- Run a Docker container and Kubernetes.
- Complete cloud computing research on cloud architecture, Docker containers, and cloud availability.

Execution

Execute this activity according to the following guidelines:

- Create an account on Docker Hub, at <https://hub.docker.com>. This will be required to complete the following tutorials.
- Complete the following tutorial on Docker:
 - Go to <https://training.play-with-docker.com> and log in to the training site using your Docker Hub account.
 - Complete the following tutorials:
 - Docker for Beginners – Linux
 - Doing More With Docker Images



GRAND CANYON UNIVERSITY™

- First Alpine Linux Containers
 - Docker Images Deeper Dive
- c. Take a screenshot for each of the completed tutorials.
3. Complete the following tutorial on Kubernetes:
- a. Go to <https://kubernetes.io/docs/tutorials/kubernetes-basics/> and complete the "Learn Kubernetes Basics" tutorial.
 - b. Now complete the following tutorials:
 - Create a Cluster
 - Deploy an App
 - Exploring your App
 - Expose Your App Publicly
 - Scale Your App
 - Update Your App
 - c. Take a screenshot for each completed tutorial.
4. Cloud Computing Research:
- a. Read the required textbook readings assigned for this topic. Provide the following research:
 - Define the business problem statement for your Milestone project.
 - Draw a business architecture diagram for your Milestone project.
 - Identify the business and technical requirements for your Milestone project.
 - b. Describe three elements that could be defined in a DockerFile. Discuss what the elements are used for and provide a brief description for each.
 - c. Research the concepts of high availability (HA), failover, and the number of nines. What is HA, failover, and the number of nines? How does the number of nines help solve HA and failover?

Documentation

Submit the following to the Learning Management System:

An activity report that contains the following related to Docker, Kubernetes, and cloud computing:

1. Cover sheet
2. Docker:
 - a. Include screenshots of completed Docker tutorials.
 - b. What is Docker?
 - c. What is a Docker File?



GRAND CANYON UNIVERSITY™

- d. What is a Docker Image?
 - e. What is a Docker Container?
 - f. What is Docker Hub?
 - g. What is Docker Swarm?
 - h. What are five advantages to using Docker Containers?
 - i. What are five Docker commands you used in the tutorial, and what was the purpose for using each of the comments?
3. Kubernetes:
- a. Include a screenshot of the completed Kubernetes tutorial.
 - b. What is Kubernetes?
 - c. Why would you have a need to use Kubernetes?
 - d. What are five features that you could leverage from Kubernetes?
 - e. What are five differences between Kubernetes and Docker Swarm?
4. Cloud computing research questions.

Activity 6: DevOps and Cloud Research

Overview

In this activity, students will:

1. Integrate DevOps concepts into the test application.
2. Complete cloud computing research on DevOps capabilities and tools.

Execution

Execute this activity according to the following guidelines:

1. DevOps logging integration with the test application.
 - a. Leverage a standard logging framework, such as SL4J, Log4j, Log4n, MonoLog, Winston, or Node-Loggly, into your test application and then add logging statements for all method entry and exit paths, as well as for all exceptions. The logging statements should leverage the proper log levels and be formatted using consistent naming conventions, consistent error messages, contain a date/time stamp, and contain the class name, as well as the method name. To save coding effort it is strongly recommended that you use the interceptor design pattern in your design and implementation.
 - b. Redeploy your test application to the Google, Azure, Heroku, and OpenShift (optional) cloud platforms.
 - c. View the log files for your deployed test application on the following cloud platforms. Take a screenshot of the log trace for each cloud platform.



GRAND CANYON UNIVERSITY™

- i. Google: To view the Application Server log, Application log, or stdout from the Portal, select your Project and then the App Engine->Versions menu and then click the Logs option from the dropdown Tools menu. To view the Application log from the Portal, select your Project and then the App Engine->Instances menu, and then SSH into the server to tail your log file.
 - ii. Azure: To view the Application Server log, Application log, or stdout from the Portal, select your Project and then the Advanced Tools menu, select the Tools->Zip Push Deploy menu options, navigate to the log file, and click the View menu. You must set the logging configuration to log to stdout.
 - iii. Heroku: To view the Application Server log, Application log, or stdout from the Portal, select your application, select the View Logs menu. You must set the logging configuration to log to stdout.
 - iv. OpenShift (if deployed): To view the Application Server log file from the Portal, select your Application from the Application->Deployments menu and then click the View Log menu. To view the Application Logs from the Portal, select your Application from the Application->Pods menu and then click the Terminal menu. From the Terminal tail your log file.
2. DevOps monitoring integration with PHP or Spring Framework (or Spring Boot) test application.
 - a. Create a free trial account on [Loggly](#). It should be noted that the free trial for Loggly only lasts for 14 days. If for some reason your trial expires, simply create another account to finish this activity.
 - b. Heroku has a Drain that will send logs in "real time" to Loggly. Also refer to the [Log Drains](#) article.
 - 1) Log into Heroku CLI.
 - 2) You will need to get your Loggly Customer Token for running the command below. You can get your Loggly Customer Token by accessing your Profile from the Loggly main menu then clicking the API Tokens tab. The Customer Token is retrieved by clicking on the hyperlink at the top of this page.
 - 3) Setup a Heroku Log Drain by running the following command:

```
heroku drains:add http://logs-01.loggly.com/bulk/[LOGGLY_CUSTOMER_TOKEN]/tag/cst323_logfile_heroku_upload --app [APP_NAME]
```
 - 4) To demonstrate a Loggy Tag Query search in Loggly using tag:[TAG NAME] (i.e. cst323_logfile_heroku_upload in this example) query. Take a screenshot of the search results.
 - 5) To demonstrate a Loggy Text Query search in Loggly using a [DESIRED TEXT] query. Take a screenshot of the search results.

NOTE: Refer to Appendix A for implementation notes.
 - c. Create a free account on [Uptime Robot](#).



GRAND CANYON UNIVERSITY™

- 1) Log in to Uptime Robot.
- 2) Set up a New Monitor using the URL of your test application's main index page on one of the cloud platforms. The Monitor Type should be HTTP(s) using the URL of test application's main index page with a monitor interval of five minutes using an Alert Contact for your GCU email address.
- 3) Test your monitor by stopping your test application on the cloud platform. Take a screenshot of your Monitor Alert email.

NOTE: Refer to Appendix B for implementation notes.

3. DevOps CI/CD integration with PHP or Spring Framework (or Spring Boot) test application (this is an optional activity and per directions given by the instructor).
 - a. Explore the build pipeline tools from one the Cloud Platforms where the test application was deployed. Choose one of the Build Pipeline tools OR select from the following tools:
 - 1) Jenkins container on Red Hat OpenShift
 - 2) GitLab
 - 3) Download [Jenkins](#) and deploy locally to JBoss or Tomcat
 - b. Setup a build pipeline using the selected tool from the above list, and deploy your test application to one of desired cloud platforms.
 - c. Complete a screencast of an automated build and deployment.
4. Cloud Computing Research:
 - a. Research the Enterprise Class Logging Analytics and Reporting tool called Splunk (at www.splunk.com). How would this tool be used by DevOps, and what features in this tool help DevOps Engineers be proactive rather than reactive to application or infrastructure issues?
 - b. Read the required textbook readings assigned for this topic. What data information is relevant and should be provided in a log file to support your application in the cloud? Provide three best practices that you should adhere to when adding logging to an application. Provide three issues or risks that could occur if inadequate logging is designed into an application.
 - c. Research three tools that could support a CI/CD build pipeline. What are the tools, and how are they used to support CI/CD?
 - d. From Chapter 10 in the textbook, identify five capabilities that drive the definition of DevOps. What are the five capabilities, and how are these used to help improve application development, testing, and delivery?

Documentation

Submit the following to the Learning Management System:



GRAND CANYON UNIVERSITY™

An activity report that contains the following on DevOps logging, monitoring, CI/CD, and cloud computing:

1. Cover sheet
2. DevOps logging:
 - a. Screenshot of example log file output from the test application deployed on each cloud platform (Google, Azure, AWS, Heroku, OpenShift (optional))
 - b. Why is adding robust logging important for an application deployed to the cloud?
 - c. What are three features of the logging framework that you did not implement, but would be important to implement for a production-level application?
 - d. What are two enterprise class logging products besides Loggly that could be used to search and archive application or system log files?
3. DevOps Monitoring:
 - a. What is the purpose for setting up a log file alert?
 - b. What is the purpose for setting up an application availability alert?
4. DevOps CI/CD:
 - a. What roles do Maven and Composer play when supporting CI/CD?
 - b. What role does a Source Control System play when supporting CI/CD?
 - c. How did your chosen build pipeline tool support CI/CD?
 - d. Besides build and deployment, what are three other features that could be integrated into a build pipeline to support a robust CI/CD?
5. Cloud computing research questions.

Activity 7: Future Cloud Computing and Cloud Research

Overview

In this activity, students will:

1. Research future cloud-based development IDEs.
2. Research future cloud computing features, such as feature toggles and an A/B switch.

Execution

Execute this activity according to the following guidelines:

1. Explore cloud IDE.
 - a. Research the features of the following cloud-based IDEs: VS Code (there are a number of cloud-based online implementations), GitHub Codespaces, Codenvy, Codeanywhere, and Cloud9 (only on AWS).
 - b. Import your test application into your chosen IDE, researched in the previous step.



GRAND CANYON UNIVERSITY™

- c. Identify 10 common day-to-day use cases that you perform using your desktop development IDE that would be required to support the development of your test application. For each use case:
- 1) Analyze your chosen cloud IDE against the 10 use cases, and validate if each of the use cases is supported in the cloud IDE.
 - 2) If the use case is supported, exercise that use case using the features of your chosen cloud IDE. While you are exercising the use case also evaluate if the feature in the cloud IDE is productive and easy to use, such that you could use the cloud IDE to reasonably replace your desktop IDE.
 - 3) If the use case is not supported, summarize why that feature is important to your development, and whether this would be a "must have" feature for you to do development using your cloud IDE.
- d. Write a 300- to 500-word research paper summarizing your observations and findings. Use the following table as a guide. Make sure to address whether you could use this cloud-based environment as a day-to-day development tool to replace your desktop IDE. Justify and rationalize your answer.

Use Case	Supported (Y/N)	Observations on Usability and Productivity	Other Notes

2. Cloud computing research:

- a. Read the required textbook readings assigned for this topic. Answer the following questions:
- 1) What is a feature flag or feature toggle?
 - 2) What is A/B testing?
 - 3) What is continuous delivery?
 - 4) What is continuous integration?
- b. One trend in cloud computing is the ability to develop code in a cloud and browser-based IDE. Research two viable existing cloud and browser-based IDEs on the market. How might these cloud-based IDEs conceptually be used to lower the cost of developing code for a company? What are some disadvantages of, or missing features in, the IDEs you researched when comparing them to a desktop-based IDE, such as Eclipse or Visual Studio?
- c. Complete a Scrum retrospective. Document the following:
- 1) What worked well in this class?
 - 2) What did not work well in this class?
 - 3) What would you like to see improved in the design of the class?



GRAND CANYON
UNIVERSITY™

Documentation

Submit the following to the Learning Management System:

An activity report that contains the following on cloud IDE analysis and cloud computing:

1. Cover sheet
2. Results of your cloud IDE research.
3. Cloud computing research questions



Appendix A – Logging Implementation Notes

To setup a log drain for Loggly in a PHP Laravel application, you must develop a logger that exports its log statements to Loggly. The following illustrations provide an example logger that can be used. Also refer to the article at <https://www.sitepoint.com/effective-php-logging-loggly/>.

```
<?php
namespace App\Services\Utility;

use Monolog\Logger;
use Monolog\Handler\LogglyHandler;

class MyLogger2 implements ILogger
{
    private static $logger = null;

    static function getLogger()
    {
        if (self::$logger == null)
        {
            self::$logger = new Logger('playlaravel');
            self::$logger->pushHandler(new LogglyHandler('[LOGGLY TOKEN]/tag/[LOGGLY TAG]', Logger::DEBUG));
        }
        return self::$logger;
    }

    public static function debug($message, $data=array())
    {
        self::getLogger()->addDebug($message, $data);
    }

    public static function info($message, $data=array())
    {
        self::getLogger()->addInfo($message, $data);
    }

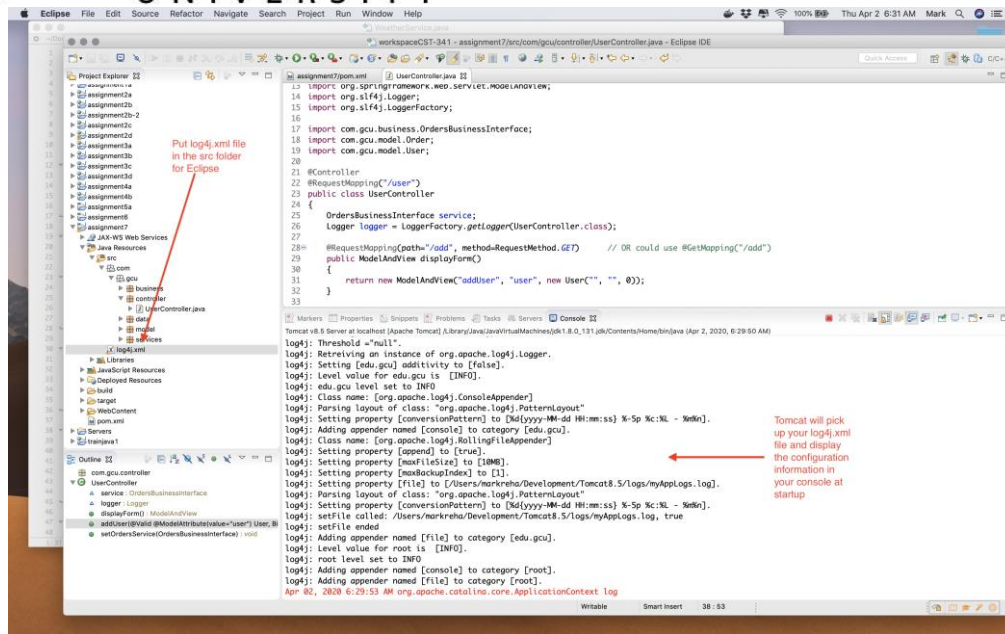
    public static function warning($message, $data=array())
    {
        self::getLogger()->addWarning($message, $data);
    }

    public static function error($message, $data=array())
    {
        self::getLogger()->addError($message, $data);
    }
}
```

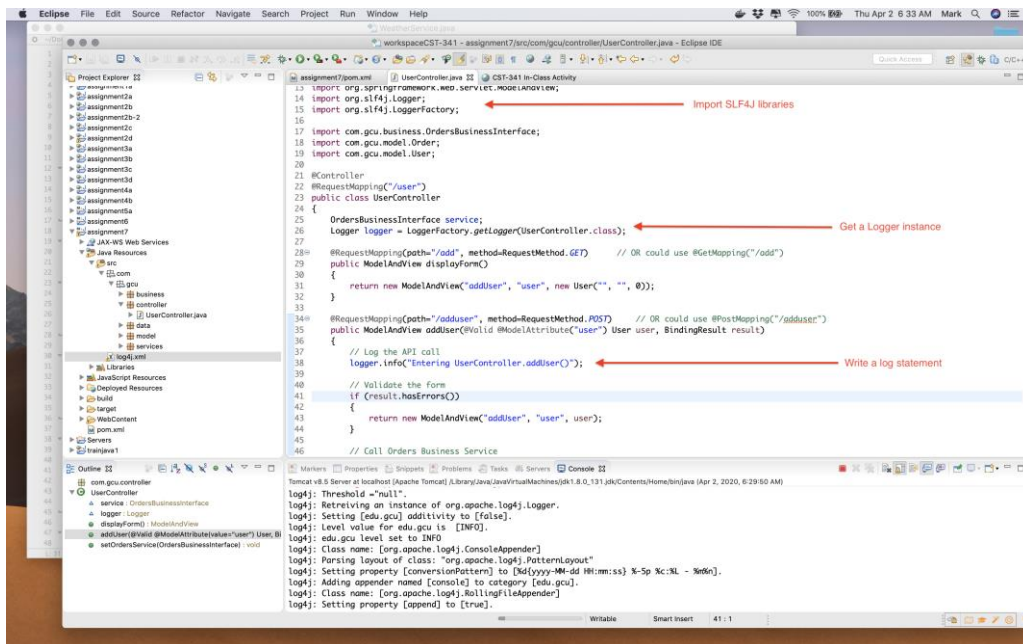
To setup a Java Spring (or Spring Boot) application to support logging, it is recommended to use the SLF4J logging framework with the Log4J log provider. The following screenshots provide some guidance for how to setup SLF4J in a Java Spring (or Spring Boot) application.



GRAND CANYON UNIVERSITY™



The Log4j Configuration file when run in Eclipse should be placed in your project *src* directory.



SLF4J can be imported, a Logger instantiated, and used to write log statements.



GRAND CANYON UNIVERSITY™

The screenshot shows the Eclipse IDE with a web browser displaying the 'CST-341 In-Class Activity' page. The page has a navigation bar with 'GCU', 'Home', 'Shop', 'News and Events', and an 'Add User' button. Below the navigation bar is a table with 10 entries, each with an 'Order Number', 'Product Name', 'Price', and 'Quantity'.

Order Number	Product Name	Price	Quantity
0000000000	This is Product 1	1	1
0000000001	This is Product 2	2	2
0000000002	This is Product 3	3	3
0000000003	This is Product 4	4	4
0000000004	This is Product 5	5	5
0000000005	This is Product 6	6	6
0000000006	This is Product 7	7	7
0000000007	This is Product 8	8	8
0000000008	This is Product 9	9	9
0000000009	This is Product 10	10	10

The console window shows the following log statements:

```
2020-04-02 06:34:00 INFO com.gcu.controller.UserController:38 - Entering UserController.addUser()
```

An arrow points from the log statement in the console to the 'Log statement in the console output' text.

The Tomcat logs directory is also shown, with a file named 'myAppLogs.log' circled. An arrow points from the log statement in the console to the 'Log statement written to a file in the Tomcat logs directory' text.

Your log statements can be written to both the console and the Tomcat logs directory.



```
<project xmlns="http://maven.apache.org/POM/4.0.0" xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance" xsi:schemaLocati
<modelVersion>4.0.0</modelVersion>
<groupId>assignment7</groupId>
<artifactId>assignment7</artifactId>
<version>0.0.1-SNAPSHOT</version>
<packaging>war</packaging>
<properties>
  <warname>assignment7</warname> <!-- WAR File Name -->
</properties>

<build>
  <sourceDirectory>src</sourceDirectory>
  <plugins>
    <plugin>
      <artifactId>maven-compiler-plugin</artifactId>
      <version>3.6.1</version>
      <configuration>
        <source>1.8</source>
        <target>1.8</target>
      </configuration>
    </plugin>
    <plugin>
      <artifactId>maven-war-plugin</artifactId>
      <version>3.0.0</version>
      <configuration>
        <warSourceDirectory>WebContent</warSourceDirectory> <!-- WAR Content Directory -->
        <warName>${warname}</warName> <!-- Final WAR File Name with no SNAPSHOTS -->
      </configuration>
    </plugin>
  </plugins>
  <resources>
    <resource>
      <directory>src</directory>
      <includes>
        <include>log4j.xml</include>
      </includes>
      <targetPath>${project.basedir}/target/classes</targetPath> <!-- Copy Log4j Configuration into Classpath -->
    </resource>
  </resources>
</build>
```

Specify the name of the WAR file in a property

The WAR file source content directory

The WAR file name with no SNAPSHOT

Copy the Log4j Configuration file to the output and in the classpath at runtime

For the Cloud Platforms, you must make sure your Maven POM is setup properly to copy the Log4J Configuration file to the WEB-INF/classes directory so it is in the Java Spring (or Spring Boot) applications classpath at runtime.



GRAND CANYON UNIVERSITY™

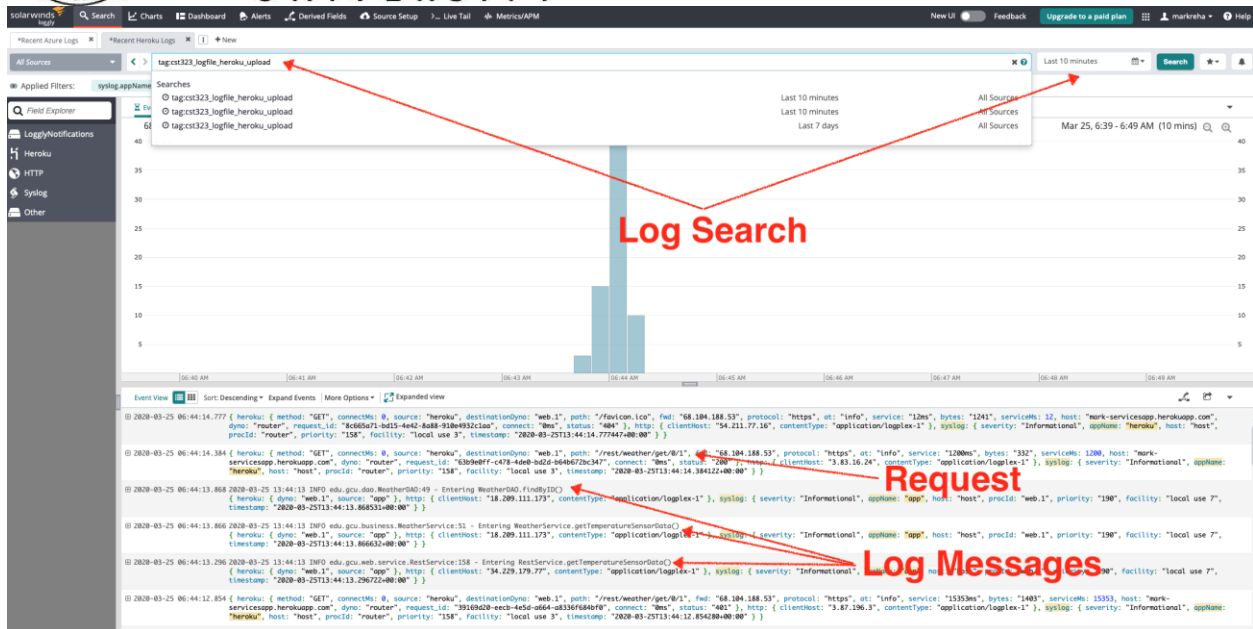
An example Log4j Configuration that logs to the console and a log file within the Tomcat logs directory, and will work with the Cloud Platforms, is shown below:

```
<?xml version="1.0" encoding="UTF-8" ?>
<!DOCTYPE log4j:configuration SYSTEM "log4j.dtd">
<log4j:configuration debug="true" xmlns:log4j="http://jakarta.apache.org/log4j/">
  <appender name="console" class="org.apache.log4j.ConsoleAppender">
    <layout class="org.apache.log4j.PatternLayout">
      <param name="ConversionPattern"
        value="%d{yyyy-MM-dd HH:mm:ss} %-5p %c:%L - %m%n" />
    </layout>
  </appender>
  <appender name="file" class="org.apache.log4j.RollingFileAppender">
    <param name="append" value="true" />
    <param name="maxFileSize" value="10MB" />
    <param name="maxBackupIndex" value="1" />
    <param name="file" value="${catalina.home}/logs/myAppLogs.log" />
    <layout class="org.apache.log4j.PatternLayout">
      <param name="ConversionPattern"
        value="%d{yyyy-MM-dd HH:mm:ss} %-5p %c:%L - %m%n" />
    </layout>
  </appender>

  <logger name="edu.gcu" additivity="false">
    <level value="INFO" />
    <appender-ref ref="console" />
    <appender-ref ref="file" />
  </logger>
  <root>
    <level value="INFO" />
    <appender-ref ref="console" />
    <appender-ref ref="file" />
  </root>
</log4j:configuration>
```



GRAND CANYON UNIVERSITY™



Example Loggly Queries

Refer to the Loggly query cheat sheet at:

<https://www.loggly.com/wp-content/uploads/2017/08/Loggly-Search-Cheat-Sheet.pdf>

Also refer to the following logging resources:

Java Logging Resources:

<https://www.tutorialspoint.com/slf4j/index.htm>

https://www.tutorialspoint.com/log4j/log4j_configuration.htm

<https://www.baeldung.com/slf4j-with-log4j2-logback>

<http://www.slf4j.org/manual.html>

<https://dzone.com/articles/logging-with-slf4j>

<https://mkyong.com/logging/slf4j-logback-tutorial/>

PHP Logging Resources:

<https://stackify.com/php-monolog-tutorial/>

<http://zetcode.com/php/monolog/>

<https://dzone.com/articles/php-monolog-tutorial-a-step-by-step-guide>

<https://seldaek.github.io/monolog/doc/01-usage.html>

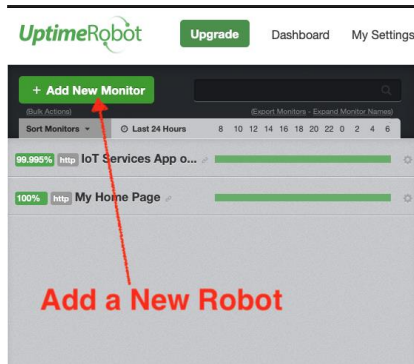
<https://www.sitepoint.com/effective-php-logging-loggly/>



Appendix B – Application Availability Implementation Notes

To monitor an application for availability, one Cloud Service Provider that can be used is Uptime Robot.

To set up a Robot, log into your Uptime Robot account, and then click on the Add New button.



First, set up an Alert Contact in My Settings Setup for your account. Then set up a Robot configuration by selecting the HTTP(s) monitor type, then give your monitor a friendly name (this is the name that will show up on your Uptime Robot Dashboard), set the URL to your HTTP URL main page (or any page that does not require authentication), and finally set the monitor frequency for every 5 minutes, with an alert to your email address.

