

Tutorial 10

AWS Deployment

CS551 Advanced Software Engineering

UMKC

Topics to cover

Amazon AWS

AWS Elastic Beanstalk

AWS Introduction

Amazon Web Services (AWS), is a collection of cloud computing services, also called web services, that make up a cloud-computing platform offered by Amazon.com.

AWS Introduction

- AWS is a collection of services in the cloud, as the definition says.
- AWS provides fast computing resources online (for example, you need 10 minutes to set up a Linux server).
- AWS offers affordable fees. AWS provides easy-to-use services out of the box, which is saves lots of time manually setting up a database, cache, storage, network and other infrastructure services.
- AWS is always available and is highly scalable.



Account Creation

AWS - Console

Open the URL : <https://aws.amazon.com/>



PRODUCTS & SERVICES

[AWS Console](#) >

[AWS Console Mobile App](#) >

[FAQs](#) >

RELATED LINKS

AWS Management Console

Access and manage Amazon Web Services through a simple and intuitive web-based user interface. You can also use the [AWS Console mobile app](#) to quickly view resources on the go.

Features

[Sign in to the AWS Console](#)




Sign In or Create an AWS Account

What is your email (phone for mobile accounts)?

E-mail or mobile number:

☐ I am a new user.

☒ I am a returning user
and my password is:

Sign in using our secure server 

[Forgot your password?](#)



Amazon Aurora | Enterprise-class database at 1/10th the cost.

“10 times faster than our MySQL environment. It just works!”
-Alfresco

[Learn more](#)

The advertisement features a blue background with a network of white lines and nodes. It includes several circular icons: a clock, a shield, a crossed-out dollar sign, and a speedometer. The speedometer is the largest and most prominent, with a needle pointing towards the top.

Learn more about [AWS Identity and Access Management](#) and [AWS Multi-Factor Authentication](#), features that provide additional security for your AWS Account. View full [AWS Free Usage Tier](#) offer terms.



Login Credentials

Use the form below to create login credentials that can be used for AWS as well as Amazon.com.

My name is:

My e-mail address is:

Type it again:

note: this is the e-mail address that we will use to contact you about your account

Enter a new password:

Type it again:

Create account

Fill in the details to create account.

Contact Information

☒ **Company Account**☐ **Personal Account**

** Required Fields*

Full Name*

Company Name*

Country*

United States ▼

Address*

Street, P.O. Box, Company Name, c/o

Apartment, suite, unit, building, floor, etc.


City*

State / Province or Region*

Postal Code*

Phone Number*

Phone Number*

Security Check 

3TE37X

[Refresh Image](#)

Please type the characters as shown above

AWS Customer Agreement

☐

Check here to indicate that you have read and agree to the terms of the [AWS Customer Agreement](#)

Create Account and Continue

You will receive a
signup confirmation
after creating account

AWS Console-Signup



Payment Information

Please enter your payment information below. You will be able to try a broad set of AWS products for free via the Free Usage Tier. We will only bill your credit or debit card for usage that is not covered by our Free Usage Tier.

AWS Free Usage Tier <i>free for 1 year</i>	Compute Amazon EC2 750hrs/month*	Storage Amazon S3 5GB	Database Amazon RDS 750hrs/month*
--	--	---	---

[*View full offer details »](#)

Credit/Debit Card Number

Expiration Date

01 ▼

2016 ▼

Cardholder's Name

☒ Use my contact address

Make sure you have
the Free Usage Tier

Contact Information

Payment Information

Identity Verification

Support Plan

Confirmation

Identity Verification

You will be called immediately by an automated system and prompted to enter the PIN number provided.

1. Provide a telephone number

Please enter your information below and click the "Call Me Now" button.

Country Code

United States (+1)

Phone Number

Ext

Call Me Now

2. Call in progress

3. Identity verification complete



Contact Information



Payment Information



Identity Verification



Support Plan



Confirmation

Identity Verification

You will be called immediately by an automated system and prompted to enter the PIN number provided.

1. Provide a telephone number ✓

2. Call in progress ✓

3. Identity verification complete

Your identity has been verified successfully

Continue to select your Support Plan

Support Plan

All customers receive free support. Choosing a paid support plan will allow you to receive one-on-one technical assistance from experienced engineers and access many other support features. Please see below.

Please Select One

☒ **Basic (Free)**

Contact Customer Service for account and billing questions, receive help for resources that don't pass system health checks, and access the AWS Community Forums.

☐ **Developer (\$49/month)**

Get started on AWS - ask technical questions and get a response to your web case within 12 hours during local business hours.

☐ **Business (Starting at \$100/month - Pricing Example) - Recommended**

24/7/365 real-time assistance by phone and chat, a 1 hour response to web cases, and help with 3rd party software. Access AWS Trusted Advisor to increase performance, fault tolerance, security, and potentially save money. ?

☐ **Enterprise**

15 minute response to web cases, an assigned technical account manager (TAM) who is an expert in your use case, and white-glove case handling that notifies your TAM and the service engineering team of a critical issue.

If you select this option, you will not be charged immediately. We will contact you to discuss your needs and finalize the sign-up.

AWS Support Features

	Basic	Developer	Business	Enterprise
Customer Service - 24x7x365	●	●	●	●
Support Forums	●	●	●	●
Documentation, White Papers, Best Practice Guides	●	●	●	●
AWS Trusted Advisor ⓘ	4 checks	4 checks	37 checks	37 checks
Access to Technical Support	Support for Health Checks	Email (local business hours)	Phone, Chat, Email, Live Screen Sharing (24/7)	Phone, Chat, Email, Live Screen Sharing (24/7)
Primary Case Handling	Technical Customer Service Associate	Cloud Support Associate	Cloud Support Engineer	Sr. Cloud Support Engineer
Users who can create Technical Support		1 (account credentials only)	Unlimited (IAM supported)	Unlimited (IAM supported)
Response Time		<12 hours	<1 hour	<15 minutes
Architecture Support		Building Blocks	Use Case Guidance	Application Architecture
Best Practice Guidance		●	●	●

The service gets activated after 24 hours and a confirmation will be sent to your email.



AWS Elastic Beanstalk




Elastic Beanstalk



Open the URL : <https://aws.amazon.com/elasticbeanstalk/>

Menu



Products

Solutions

Pricing

Software

Support

More ▾

English ▾

My Account ▾

Complete Sign Up

PRODUCTS & SERVICES

AWS Elastic Beanstalk >

Product Details >

Pricing >

Getting Started >

Developer Resources >

FAQs >

RELATED LINKS

Documentation

Release Notes

Discussion Forum

AWS DevOps Blog

AWS Elastic Beanstalk

Easy to begin, Impossible to outgrow

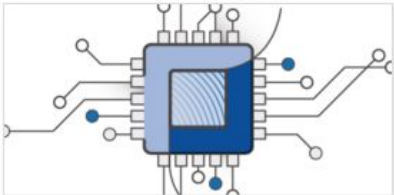
AWS Elastic Beanstalk is an easy-to-use service for deploying and scaling web applications and services developed with Java, .NET, PHP, Node.js, Python, Ruby, Go, and Docker on familiar servers such as Apache, Nginx, Passenger, and IIS.

You can simply upload your code and Elastic Beanstalk automatically handles the deployment, from capacity provisioning, load balancing, auto-scaling to application health monitoring. At the same time, you retain full control over the AWS resources powering your application and can access the underlying resources at any time.

There is no additional charge for Elastic Beanstalk - you pay only for the AWS resources needed to store and run your applications.

Manage Your AWS Resources

Sign in to the Console



AWS WEBINAR | MAR 29, 9AM PT

Learn how to fully utilize Amazon EC2 and related services including creating an

AWS Console - Elastic Beanstalk

AWS Services [SHOW ALL SERVICES](#)



Compute



EC2

Virtual Servers in the Cloud

Elastic Beanstalk

Run and Manage Web Apps

EC2 Container Service

Run and Manage Docker Containers

Lambda

Run Code in Response to Events



Storage & Content
Delivery



Database



Networking



Developer Tools



Management
Tools



Security & Identity



Analytics



Internet of Things



Mobile Services



Application
Services



Enterprise
Applications



Game
Development

JAVA on Elastic Beanstalk

When you use the AWS Elastic Beanstalk console to deploy a new application or an application version, you'll need to upload a source bundle. Your source bundle must meet the following requirements:

- Consist of a single `ZIP` file or `WAR` file (you can include multiple `WAR` files inside your `ZIP` file)
- Not exceed 512 MB
- Not include a parent folder or top-level directory (subdirectories are fine)

If you want to deploy a worker application that processes periodic background tasks, your application source bundle must also include a `cron.yaml` file. For more information, see [Periodic Tasks](#).

If you are deploying your application with the Elastic Beanstalk Command Line Interface (EB CLI), the AWS Toolkit for Eclipse, or the AWS Toolkit for Visual Studio, the ZIP or WAR file will automatically be structured correctly. For more information, see [The Elastic Beanstalk Command Line Interface \(EB CLI\)](#), [Creating and Deploying Java Applications on AWS Elastic Beanstalk](#), and [AWS Toolkit for Visual Studio](#).

Supported Platforms

Java with Tomcat

Elastic Beanstalk supports the following Tomcat configurations:

Configuration and <i>Solution Stack Name</i>	AMI	Language	Application Server	Web Server
Java 8 with Tomcat 8 version 2.0.8 <i>64bit Amazon Linux 2015.09 v2.0.8 running Tomcat 8 Java 8</i>	2015.09	Java 1.8.0_71	Tomcat 8.0.30	Apache 2.2.31
Java 7 with Tomcat 7 version 2.0.8 <i>64bit Amazon Linux 2015.09 v2.0.8 running Tomcat 7 Java 7</i>	2015.09	Java 1.7.0_95	Tomcat 7.0.67	Apache 2.2.31
Java 6 with Tomcat 7 version 2.0.8 <i>64bit Amazon Linux 2015.09 v2.0.8 running Tomcat 7 Java 6</i>	2015.09	Java 1.6.0_38	Tomcat 7.0.67	Apache 2.2.31

Structuring your JAVA application

```
~/workspace/my-app/  
|-- build.sh           - Build script that compiles classes and creates a WAR  
|-- README.MD         - Readme file with information about your project, notes  
|-- ROOT.war          - Source bundle artifact created by build.sh  
`-- src               - Source code folder  
    |-- WEB-INF       - Folder for private supporting files  
    |   |-- classes   - Compiled classes  
    |   |-- lib        - JAR libraries  
    |   |-- tags      - Tag files  
    |   |-- tlds       - Tag Library Descriptor files  
    |   |-- web.xml    - Deployment Descriptor  
    |-- com           - Uncompiled classes  
    |-- css           - Stylesheets  
    |-- images        - Image files  
    |-- js            - JavaScript files  
    `-- default.jsp   - JSP (JavaServer Pages) web page
```

Deploying an application to Elastic Beanstalk

1. Open the [Elastic Beanstalk console](#)
2. On the Elastic Beanstalk application navigation bar, click **Create New Application**.



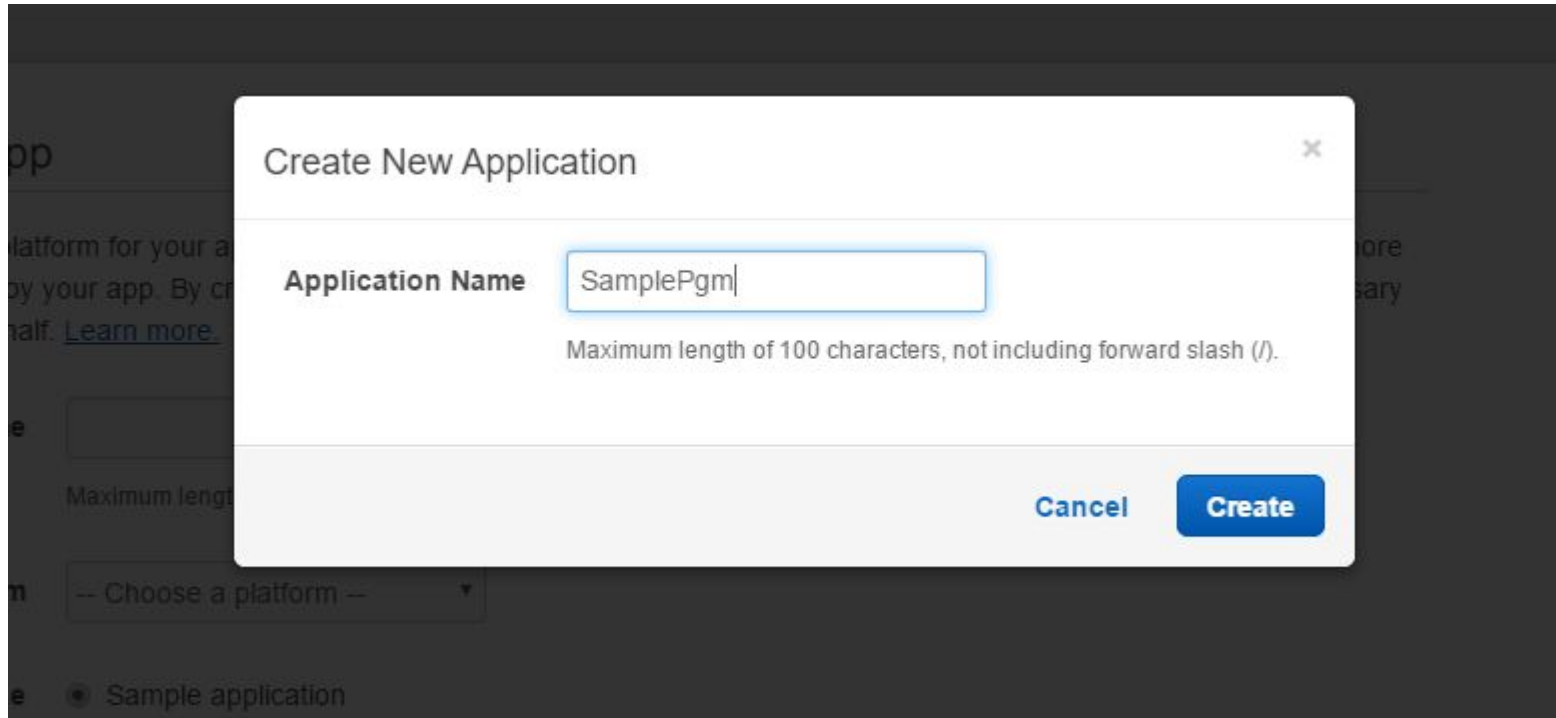
Create a web app

Choose a name and a platform for your app. You can start with a sample app or upload your own code. Then, you can configure more options before you deploy your app. By creating an app, you allow AWS Elastic Beanstalk to administer AWS resources and necessary permissions on your behalf. [Learn more.](#)

Application name

Maximum length of 100 characters, not including forward slash (/).

3. Enter the name of the application and, Then click **Create**.



The image shows a 'Create New Application' dialog box with a close button (X) in the top right corner. The 'Application Name' field contains the text 'SamplePgm|' and is highlighted with a blue border. Below the field, a note states: 'Maximum length of 100 characters, not including forward slash (/)'. At the bottom right, there are two buttons: 'Cancel' and 'Create'.

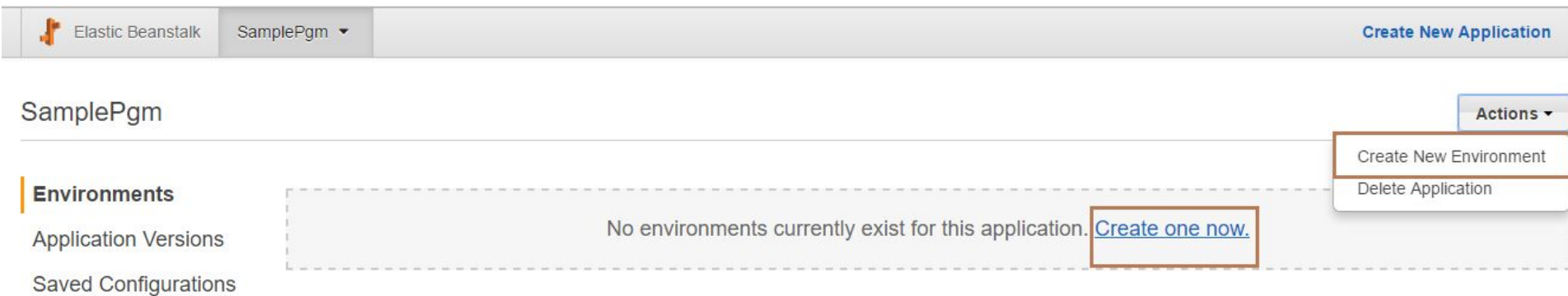
Create New Application

Application Name

Maximum length of 100 characters, not including forward slash (/).

Cancel Create

4. In the upper right corner, choose **Create a New Environment**.



The screenshot displays the AWS Elastic Beanstalk console interface. At the top, the header shows the Elastic Beanstalk logo, the application name 'SamplePgm' with a dropdown arrow, and a 'Create New Application' button. Below the header, the main content area is titled 'SamplePgm'. On the left, there is a sidebar with a vertical orange bar and the following links: 'Environments' (highlighted with an orange bar), 'Application Versions', and 'Saved Configurations'. The main content area shows a message: 'No environments currently exist for this application.' followed by a blue link 'Create one now.' which is enclosed in a red rectangular box. To the right of this message, an 'Actions' dropdown menu is open, displaying two options: 'Create New Environment' and 'Delete Application', both of which are also enclosed in red rectangular boxes.

Elastic Beanstalk SamplePgm ▼ Create New Application

SamplePgm

Environments

Application Versions

Saved Configurations

No environments currently exist for this application. [Create one now.](#)

Actions ▼

- Create New Environment
- Delete Application

5. From the drop down menu select **Tomcat** under platform and select the option **Upload your own code** where the WAR or ZIP is to be uploaded.



Create a new environment

Choose a platform for your new environment. You can start with a sample app, use an existing app or upload your own code. Then, you can configure more options or deploy your environment. By creating an environment, you allow AWS Elastic Beanstalk to administer AWS resources and necessary permissions on your behalf. [Learn more.](#)

Application name SamplePgm

Platform

Tomcat ▼

-- Choose a platform -- (after initial setup)

Docker

Go

Java

Node.js

PHP

Python

Ruby

Tomcat

.NET

App code

configure your application. You can upload a new source code for this app later.

ons which have already been uploaded for **SamplePgm**.

-- Choose a version -- ▼

☒ Upload your own code

You can upload a file or provide a URL to your app code in Amazon S3.

Upload

ZIP or WAR

6. Choose **Sample Application** or **Upload your own** and upload an application source bundle.

Create a new environment


Choose a platform for your new environment. You can start with a sample app, use an existing app, or upload your own code. After you choose a platform, you can configure more options or deploy your environment. By creating an environment, AWS creates the necessary AWS resources and necessary permissions on your behalf. [Learn more.](#)

Application name SamplePgm

Platform Tomcat
Tomcat 8 Java 8 (this can be updated after initial setup)

App code

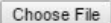
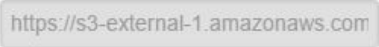
- ☒ Sample application
Comes with instructions on how to configure your application.
- ☐ Use existing version
The following are application versions which have already been deployed to this environment.
-- Choose a version --
- ☐ Upload your own code
You can upload a file or provide a URL to your app code in Amazon S3.


 **Upload** ZIP or WAR

Upload your own code

You can provide the location of an application in S3 or upload source code from your local computer.

Source code origin

- ☒ Local file
(Maximum size 512 MB)
 No file chosen
- ☐ Public S3 URL
 https://s3-external-1.amazonaws.com

Version label  samplepgm-source
Useful when multiple uploads have the same filename.

[Cancel](#) [Upload](#)

[SamplePgm](#) ▶ samplepgm (Environment ID: e-3yb9j7fgvi, URL:)



Creating samplepgm

This will take a few minutes..

12:26am Successfully launched environment: samplepgm

12:26am Environment health has transitioned from Pending to Ok.

12:25am Added instance [i-01765cd3a796ca6fc] to your environment.

12:24am Waiting for EC2 instances to launch. This may take a few minutes.

12:23am Environment health has transitioned to Pending. There are no instances.

12:22am Created security group named:
awseb-e-3yb9j7fgvi-stack-AWSEBSecurityGroup-1WAIUUEJCPM8A

12:22am Created EIP: 52.87.17.173

12:22am Using elasticbeanstalk-us-east-1-037654936661 as Amazon S3 storage bucket for environment data.

12:22am createEnvironment is starting.

[SamplePgm](#) ▸ samplepgm (Environment ID: e-3yb9j7fgvi, URL: samplepgm.uq2sf3jmee.us-east-1.elasticbeanstalk.com)

Actions ▾

Dashboard

[Configuration](#)[Logs](#)[Health](#) NEW[Monitoring](#)[Alarms](#)[Events](#)[Tags](#)

Overview

[Refresh](#)

Health

Ok

[Causes](#)

Running Version

samplepgm-source

[Upload and Deploy](#)

Configuration

64bit Amazon Linux 2015.09
v2.0.8 running Tomcat 8 Java 8[Change](#)

Recent Events

[Show All](#)

Time	Type	Details
2016-03-22 00:26:27 UTC-0500	INFO	Successfully launched environment: samplepgm
2016-03-22 00:26:10 UTC-0500	INFO	Environment health has transitioned from Pending to Ok.
2016-03-22 00:25:11 UTC-0500	INFO	Added instance [i-01765cd3a796ca6fc] to your environment.

Generating and Uploading war file

Download the source code from <http://www.mkyong.com/webservices/jax-rs/reteasy-hello-world-example/>

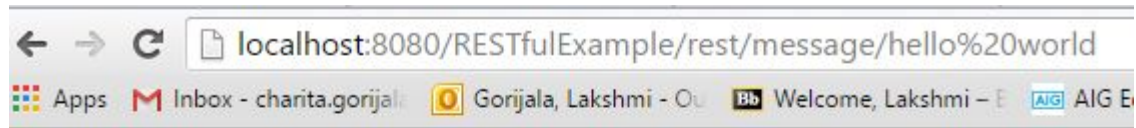
Sample Java REST App

Java EE - RESTfulExample/src/main/java/com/mkyong/rest/MessageRestService.java - Eclipse

File Edit Source Refactor Navigate Search Project Run Window Help

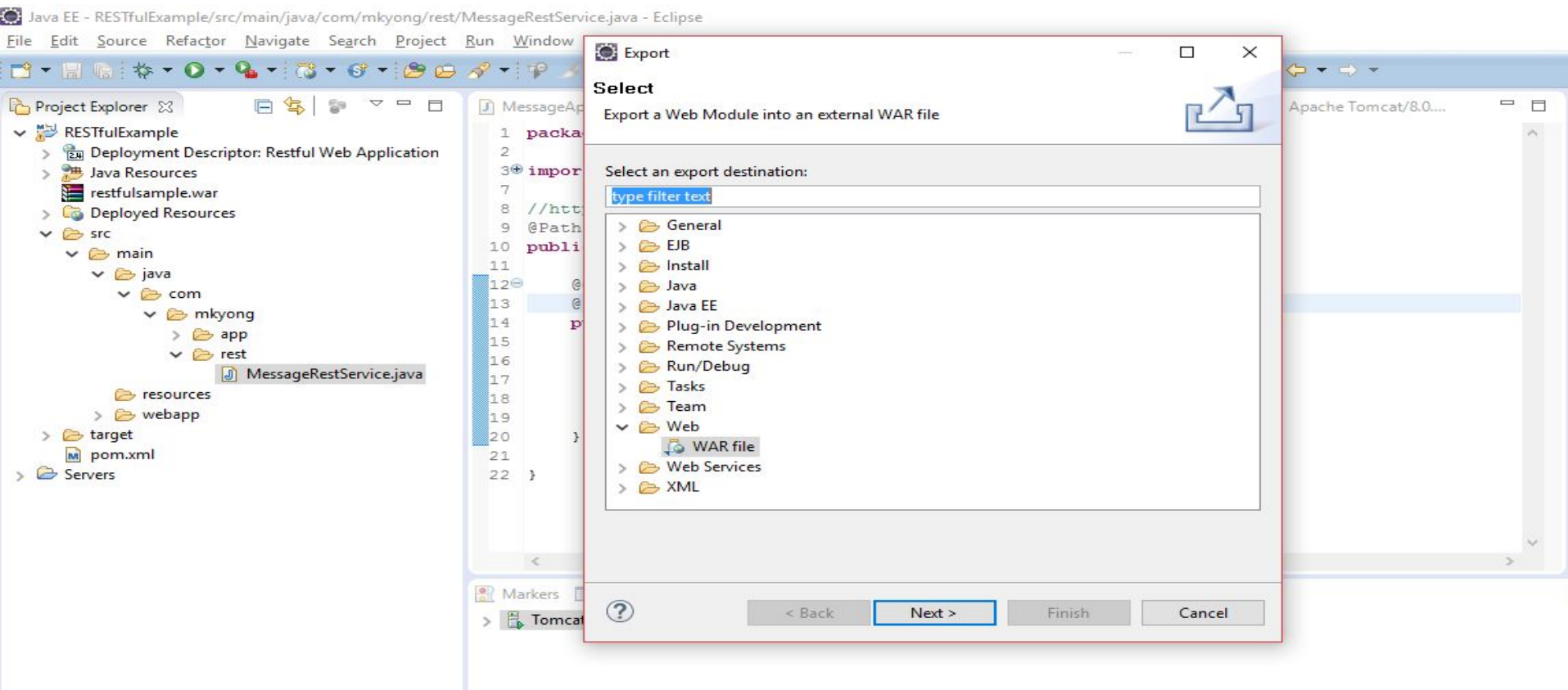
```
1 package com.mkyong.rest;
2
3 import javax.ws.rs.GET;
4
5
6 //http://localhost:8080/RESTfulExample/rest/message/hello%20world
7 @Path("/message")
8 public class MessageRestService {
9
10     @GET
11     @Path("/{param}")
12     public Response printMessage(@PathParam("param") String msg) {
13
14         String result = "Message : " + msg;
15
16         return Response.status(200).entity(result).build();
17     }
18 }
19
20
21
22 }
```

Local Run



Message : hello world

Generate WAR file from Eclipse





Project Explorer

- RESTfulExample
 - Deployment Descriptor: Restful Web Application
 - Java Resources
 - restfulsample.war
 - Deployed Resources
 - src
 - main
 - java
 - com
 - mkyong
 - app
 - rest
 - MessageRestService.java
 - resources
 - webapp
 - target
 - pom.xml
 - Servers

MessageAp

```
1 packa
2
3+ impor
4
5
6
7
8 //htt
9 @Path
10 publi
11
12 @
13 @
14 p
15
16
17
18
19
20 }
21
22 }
```

Markers

> Tomcat

Export

WAR Export

Resource C:\Users\ssr\Desktop\AWS\RESTfulExample\restful.war already exists. If you wish to overwrite select the "Overwrite Existing file" option.

Web project: RESTfulExample

Destination: C:\Users\ssr\Desktop\AWS\RESTfulExample\restful.war

Browse...

Target runtime

☐ Optimize for a specific server runtime

Apache Tomcat v8.0

☐ Export source files☐ Overwrite existing file

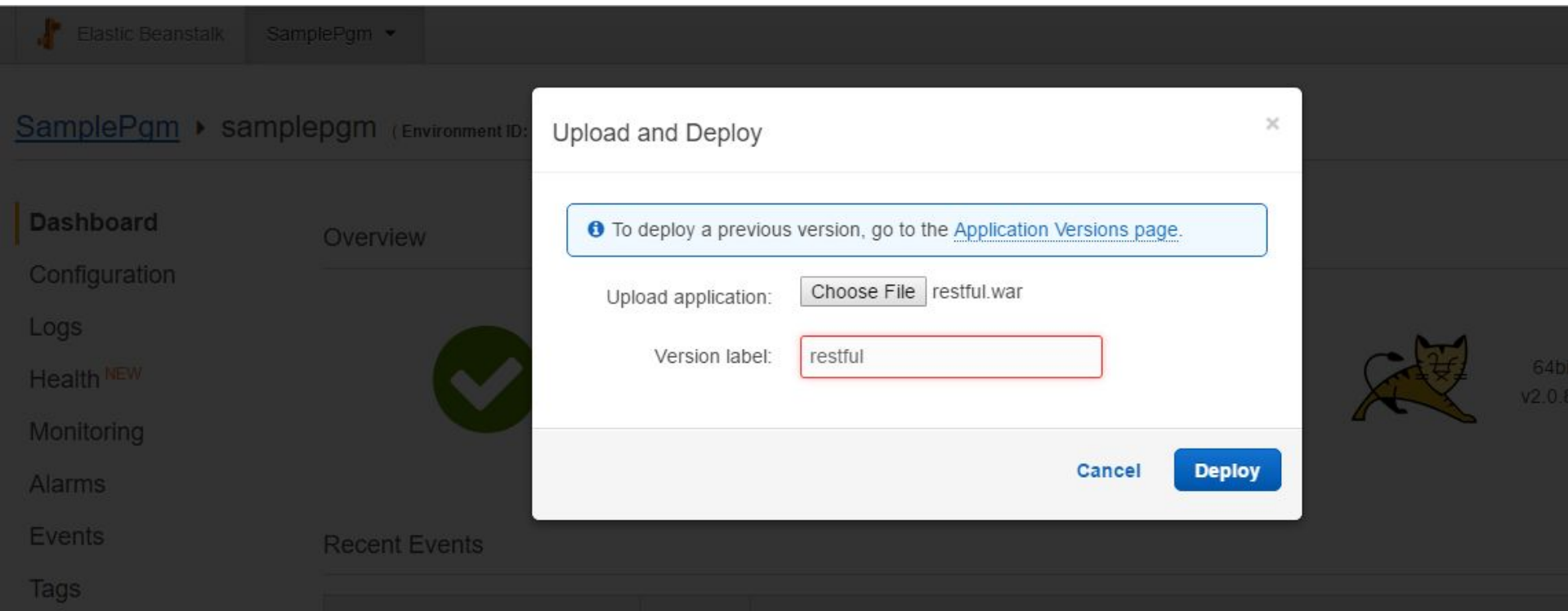
< Back

Next >

Finish

Cancel

Upload and Deploy the WAR file



The screenshot shows the Elastic Beanstalk console interface. At the top, there's a header with the Elastic Beanstalk logo and a dropdown menu for 'SamplePgm'. Below this, the breadcrumb navigation shows 'SamplePgm' > 'samplepgm' (Environment ID: ...). The left sidebar contains a navigation menu with 'Dashboard' (selected), 'Configuration', 'Logs', 'Health' (with a 'NEW' badge), 'Monitoring', 'Alarms', 'Events', and 'Tags'. The main content area is titled 'Overview' and shows a large green checkmark icon. A modal dialog box titled 'Upload and Deploy' is open in the center. It contains an information message: 'To deploy a previous version, go to the [Application Versions page](#).' Below this, there's a section for 'Upload application:' with a 'Choose File' button and the text 'restful.war'. Underneath, the 'Version label:' is set to 'restful'. At the bottom right of the dialog are 'Cancel' and 'Deploy' buttons.

Elastic Beanstalk SamplePgm

SamplePgm ▶ samplepgm (Environment ID: ...)

Dashboard Overview

Configuration

Logs

Health ^{NEW}

Monitoring

Alarms

Events

Tags

Recent Events

Upload and Deploy

To deploy a previous version, go to the [Application Versions page](#).

Upload application: Choose File restful.war

Version label: restful

Cancel Deploy

Deployment Successful

[SamplePgm](#) ▶ samplepgm (Environment ID: e-3yb9j7fgvi, URL: samplepgm.uq2sf3imee.us-east-1.elasticbeanstalk.com)

Actions ▾

Dashboard

Configuration

Logs

Health **NEW**

Monitoring

Alarms

Events

Tags

Overview

Refresh



Health

Ok

Causes

Running Version

restful

Upload and Deploy



Configuration

64bit Amazon Linux 2015.09
v2.0.8 running Tomcat 8 Java 8

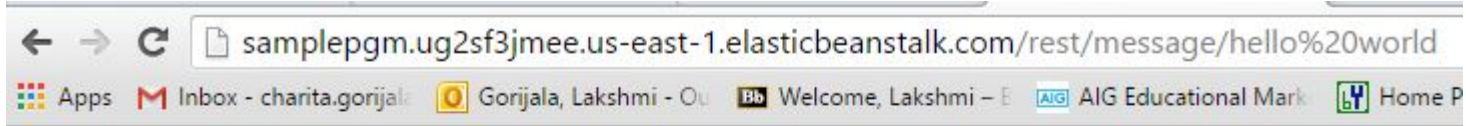
Change

Recent Events

Show All

Time	Type	Details
2016-03-22 01:43:27 UTC-0500	INFO	Environment update completed successfully.
2016-03-22 01:43:27 UTC-0500	INFO	New application version was deployed to running EC2 instances.

Test the url



Message : hello world

Task

Deploy your applications onto your Amazon instances and share the URL.

Submit the code under the Source folder in Github.

References

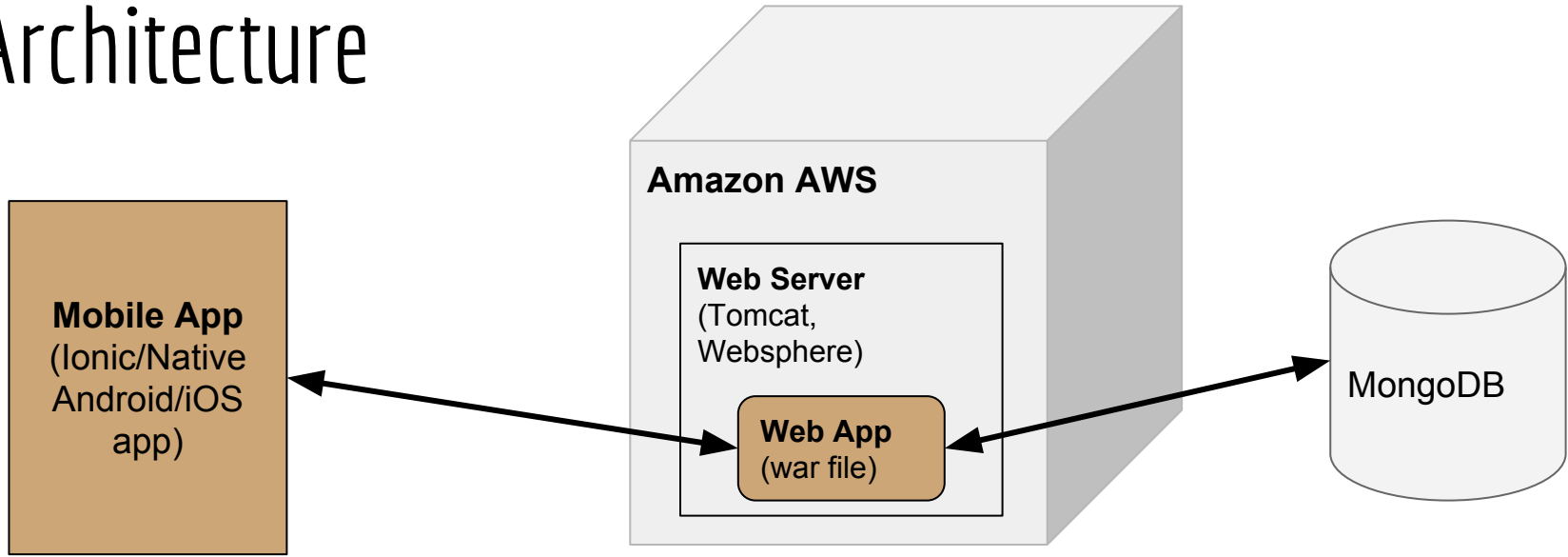
<https://www.toptal.com/aws/boost-your-productivity-with-aws>

<https://aws.amazon.com/elasticbeanstalk/>

<http://www.mkyong.com/webservices/jax-rs/resteasy-hello-world-example/>

Review : All Tutorials

Architecture



Everything together makes more sense..

Review

1. Git + ZenHub + Visio + Creately
2. Android (Android Studio, Android SDK, AVD, Gradle, Jargon like Activity, Intent, Fragments, layouts, drawable etc)
3. HTML5, CSS, Javascript
4. Angular JS
5. Servlets, WebSphere, Unit testing
6. REST : Servlets with MongoDB
7. Hybrid app : Ionic basics and unit testing
8. Ionic plugins and performance testing
9. Amazon WS



Log.w(“TAs Message”, “THANK YOU”);

