



## Distributed Systems

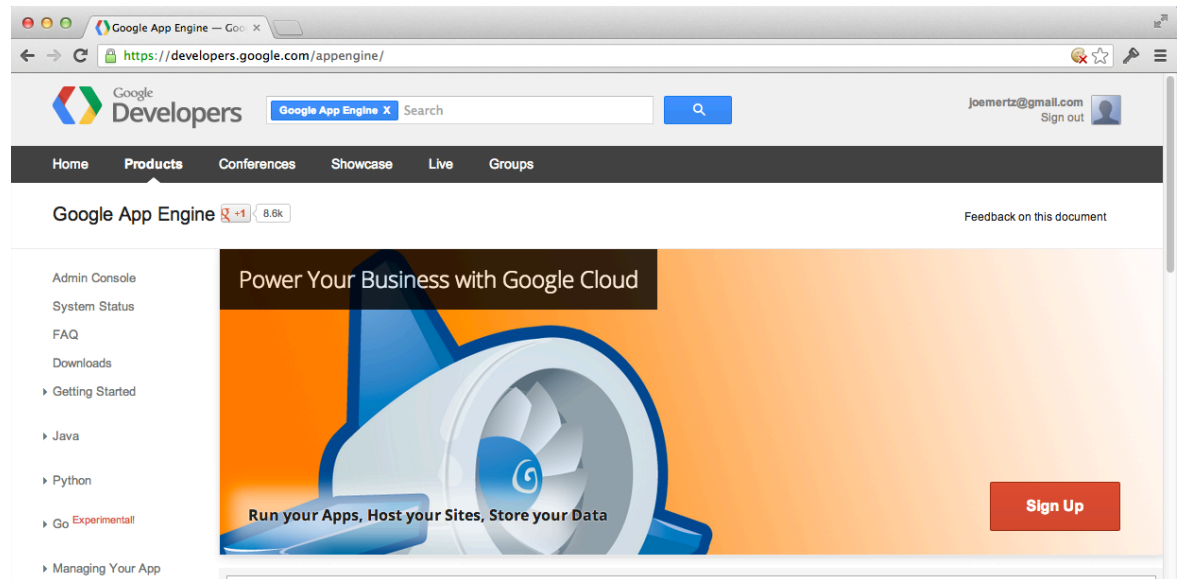
# Cloud Computing via Google App Engine

# Google App Engine

- Free
  - To some threshold, for development
- Provides a Java Web Container
- Provides a simple development environment
  - As an Eclipse plugin
- It is a good example of a PaaS to experiment with
  - And use in a homework project.

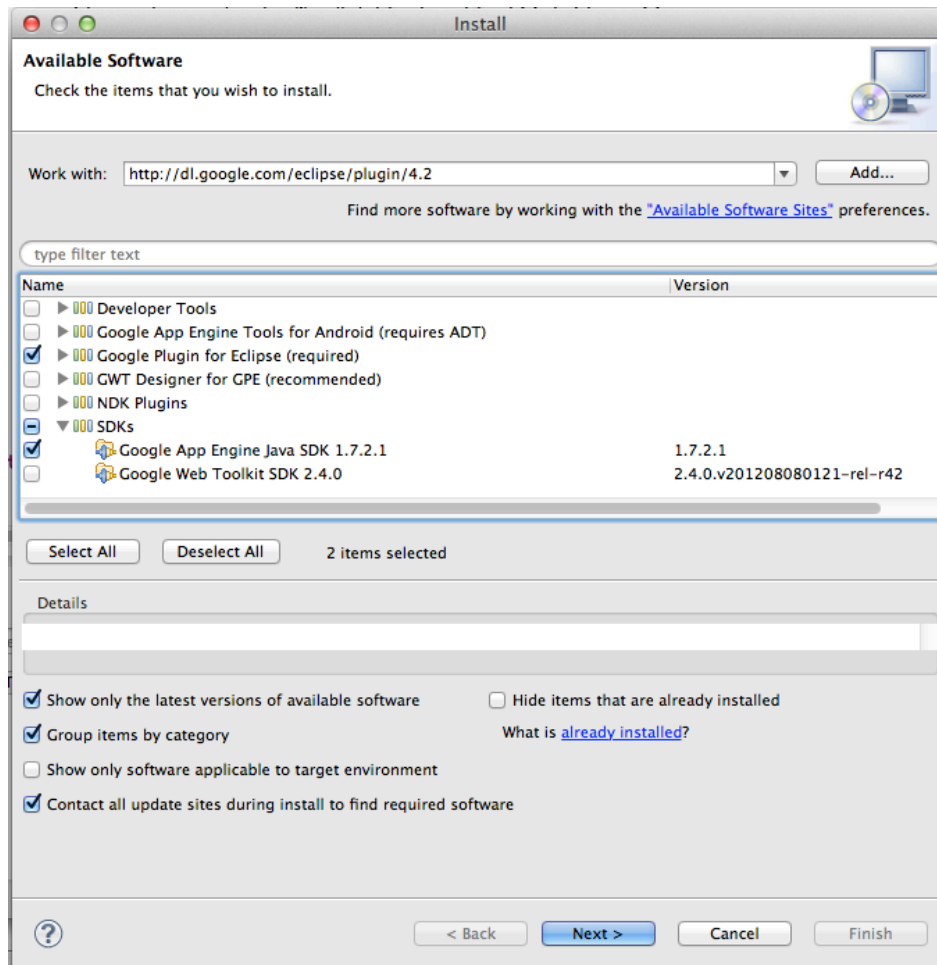
# Before today\* you should have

- Gone to <https://developers.google.com/appengine/docs/java/gettingstarted/introduction>
- Signed up for a GAE account
- Downloaded and installed the App Engine plugin for Eclipse



\*See the email I sent Saturday.

# Installing Eclipse GAE Plugin



When installing,  
choose:

- Google Plugin for Eclipse
- Google App Engine Java SDK



# Hints: Installing under Windows

- Make sure to run Eclipse.exe as an Administrator before installing.
- If installation hangs after a certain point (such as 48% of the process), disable Anti-virus software (such as AVG) temporarily and then proceed.

# App Engine Development Environment

- I will walk through a simple development cycle
- Show working on localhost
- Identify the deploy button
- Show working on Google App Engine

# My Applications

Applications Overview

https://appengine.google.com

Applications Overview

Google app engine

joemertz@gmail.com | [My Account](#) | [Help](#) | [Sign out](#)

## My Applications

« Prev 20 1-7 of 7 Next 20 »

Application	Title	Billing Administrator	Storage Scheme	Current Version
<a href="#">alflickrbet</a>	AlFlickrBet		Master/Slave ( <a href="#">Migrate to High Replication</a> )	1
<a href="#">ds95702</a>	Example DS Application		High Replication	None Deployed
<a href="#">newalflickrbet</a>	New AlFlickrBet		Master/Slave ( <a href="#">Migrate to High Replication</a> )	1
<a href="#">proxyflickr</a>	Proxy Flickr		Master/Slave ( <a href="#">Migrate to High Replication</a> )	1
<a href="#">tang-1</a>	Test application Tang-1		Master/Slave ( <a href="#">Migrate to High Replication</a> )	1
<a href="#">tang-2</a>	Another test		Master/Slave ( <a href="#">Migrate to High Replication</a> )	1
<a href="#">tang-3</a>	Another test		Master/Slave ( <a href="#">Migrate to High Replication</a> )	1

Create Application

You have 3 applications remaining.

« Prev 20 1-7 of 7 Next 20 »

# Create an Application

Create an Application

https://appengine.google.com/start/createapp? Reader

Create an Application

Google app engine joemertz@gmail.com | [My Account](#) | [Help](#) | [Sign out](#)

## Create an Application

You have 4 applications remaining.

**Application Identifier:**  
ds95702 .appspot.com [Check Availability](#) Yes, "ds95702" is available!  
All Google account names and certain offensive or trademarked names may not be used as Application Identifiers. You can map this application to your own domain later. [Learn more](#)

**Application Title:**  
Example DS Application  
Displayed when users access your application.

**Authentication Options (Advanced):** [Learn more](#)  
Google App Engine provides an API for authenticating your users, including Google Accounts, Google Apps, and OpenID. If you choose to use this feature for some parts of your site, you'll need to specify now what type of users can sign in to your application:

- ☒ **Open to all Google Accounts users (default)**  
If your application uses authentication, anyone with a valid Google Account may sign in.
- ☐ **Restricted to the following Google Apps domain:**  
  
e.g. foo.com  
If your application uses authentication, only members of this Google Apps domain may sign in. If your organization uses Google Apps, use this option to create an application (e.g. an HR tracking tool) that is only accessible to accounts on your Google Apps domain. This option cannot be changed once it has been set.
- ☐ **(Experimental) Open to all users with an OpenID Provider**  
If your application uses authentication, anyone who has an account with an OpenID Provider may sign in.

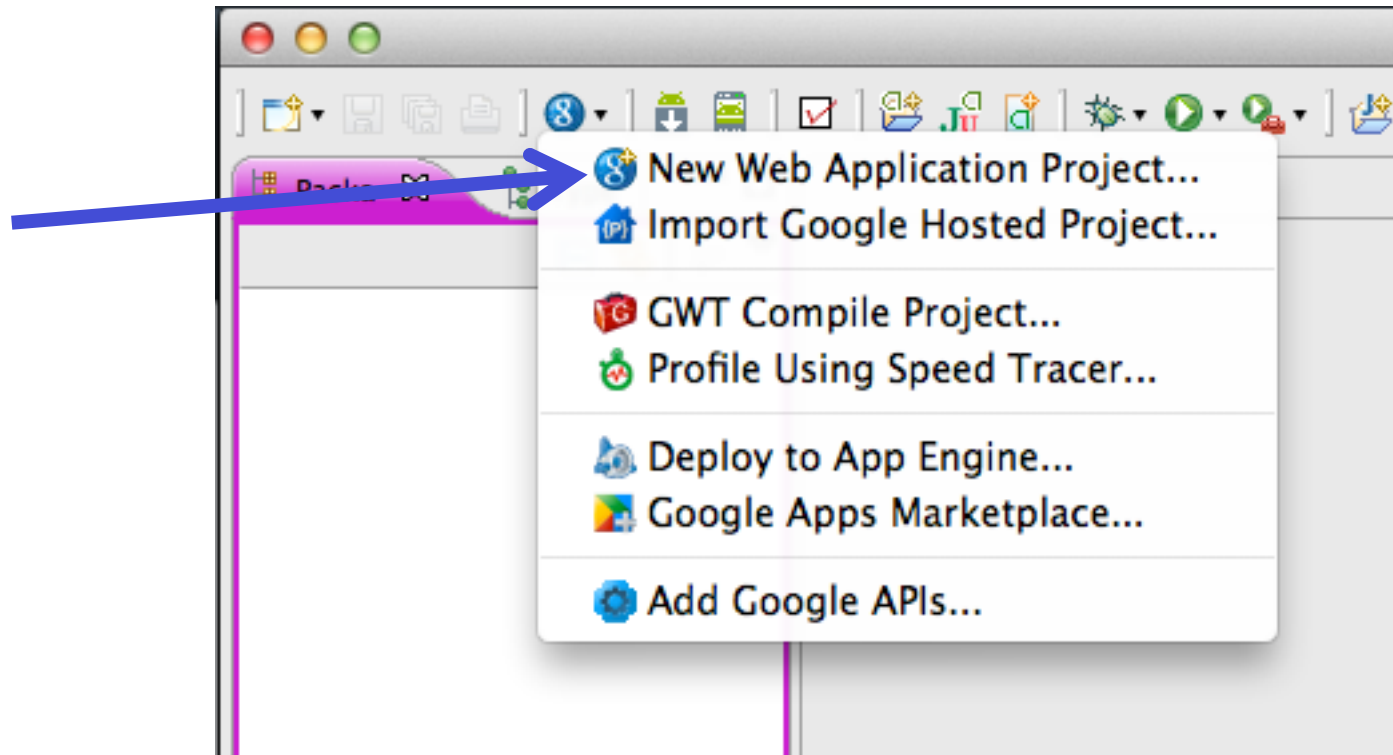
**Storage Options (Advanced):**  
Google App Engine datastore options.

- ☒ **High Replication (default)**  
Uses a more highly replicated Datastore that makes use of a system based on the Paxos algorithm to synchronously replicate data across multiple locations simultaneously. Offers the highest level of availability for reads and writes at the cost of eventual consistency for some queries.
- ☐ **Master/Slave ~~Deprecated~~**  
Uses a master-slave replication system, which asynchronously replicates data as you write it to another physical datacenter. Since only one datacenter is the master for writing at any given time, this option offers strong consistency for all reads and queries, at the cost of periods of temporary unavailability during datacenter issues or moves.  
⚠ The Python 2.7 and Go runtimes are not supported for applications that use the Master/Slave Datastore.

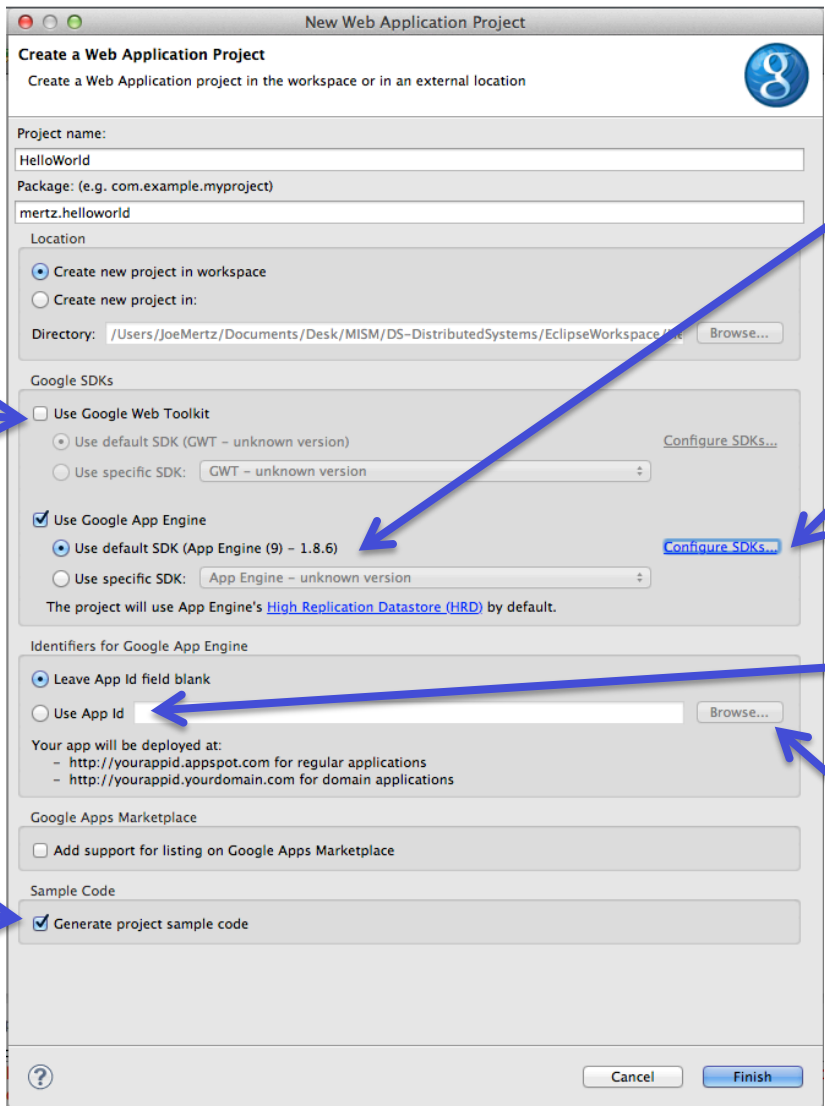
[Create Application](#) [Cancel](#)



# Eclipse: New Web Application Project



# Create a Web Application Project



The screenshot shows the 'New Web Application Project' dialog in the Eclipse IDE. The dialog is titled 'New Web Application Project' and has a Google logo in the top right corner. It contains several sections for configuring the project:

- Project name:** 'HelloWorld'
- Package:** 'mertz.helloworld'
- Location:** 'Create new project in workspace' (selected), 'Create new project in: /Users/JoeMertz/Documents/Desktop/MISM/DS-DistributedSystems/EclipseWorkspace/...' (with a 'Browse...' button).
- Google SDKs:**
  - ☐ Use Google Web Toolkit
    - ☐ Use default SDK (GWT - unknown version) (with a 'Configure SDKs...' button)
    - ☐ Use specific SDK: 'GWT - unknown version' (with a dropdown arrow)
  - ☒ Use Google App Engine
    - ☒ Use default SDK (App Engine (9) - 1.8.6) (with a 'Configure SDKs...' button)
    - ☐ Use specific SDK: 'App Engine - unknown version' (with a dropdown arrow)

The project will use App Engine's [High Replication Datastore \(HRD\)](#) by default.
- Identifiers for Google App Engine:**
  - ☒ Leave App Id field blank
  - ☐ Use App Id (with a 'Browse...' button)

Your app will be deployed at:  
- <http://yourappid.appspot.com> for regular applications  
- <http://yourappid.yourdomain.com> for domain applications
- Google Apps Marketplace:**
  - ☐ Add support for listing on Google Apps Marketplace
- Sample Code:**
  - ☒ Generate project sample code

At the bottom, there are 'Cancel' and 'Finish' buttons.

Annotations with blue arrows point to specific elements:

- An arrow points to the 'Unselect for we are not using Google Web Toolkit' text, which points to the 'Use Google Web Toolkit' checkbox.
- An arrow points to the 'Select (Provides a good starting place)' text, which points to the 'Generate project sample code' checkbox.
- An arrow points to the 'If default is not 1.9.0 (the one you just installed)' text, which points to the 'Configure SDKs...' button next to the 'Use default SDK (App Engine (9) - 1.8.6)' option.
- An arrow points to the 'Click on Configure SDKs and choose 1.9.0' text, which points to the 'Configure SDKs...' button next to the 'Use specific SDK: App Engine - unknown version' option.
- An arrow points to the 'Select "Use App ID" and click on Browse... to set the Application ID you created earlier.' text, which points to the 'Use App ID' radio button and its associated 'Browse...' button.

(See next slide)

# Create a Web Application Project

If not  
logged in,  
click here  
to log in

New Web Application Project

Create a Web Application Project

Enter an App Id

Project name:  
HelloWorld

Package: (e.g. com.example.myproject)  
mertz.helloworld

Location

☒ Create new project in workspace  
☐ Create new project in:

Directory: /Users/JoeMertz/Documents/Desktop/MISM/DS-DistributedSystems/EclipseWorkspace/HelloWorld  
Browse...

Google ST

☐ Use Google App Id  
☒ Use Existing App Id  
☐ Use Google App Id

Click the link below to log in to view and/or create app ids.

App Ids

Click [here](#) to log in.

Create App Id

Cancel OK

Identifiers

☐ Leave App Id field blank  
☒ Use App Id

Your app will be deployed at:

- http://yourappid.appspot.com for regular applications
- http://yourappid.yourdomain.com for domain applications

Google Apps Marketplace

☐ Add support for listing on Google Apps Marketplace

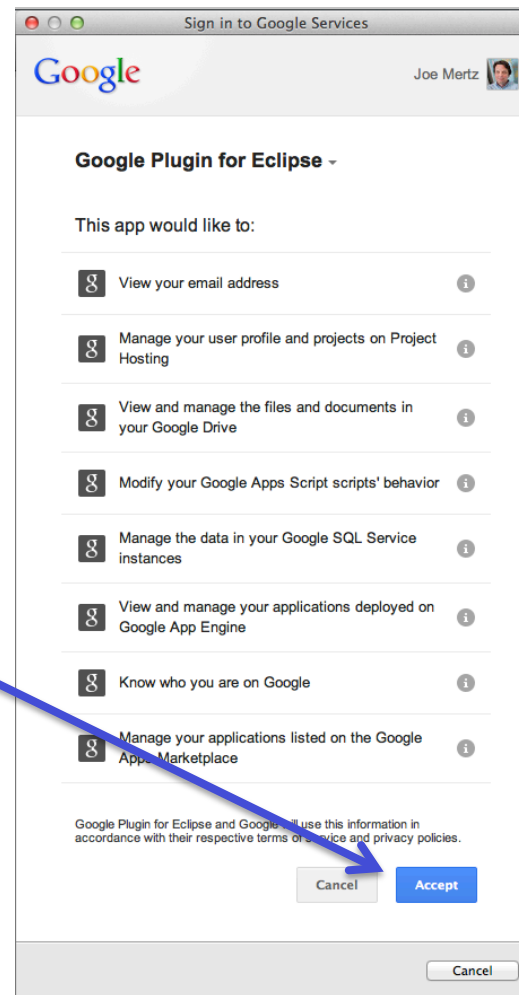
Sample Code

☒ Generate project sample code

Cancel Finish

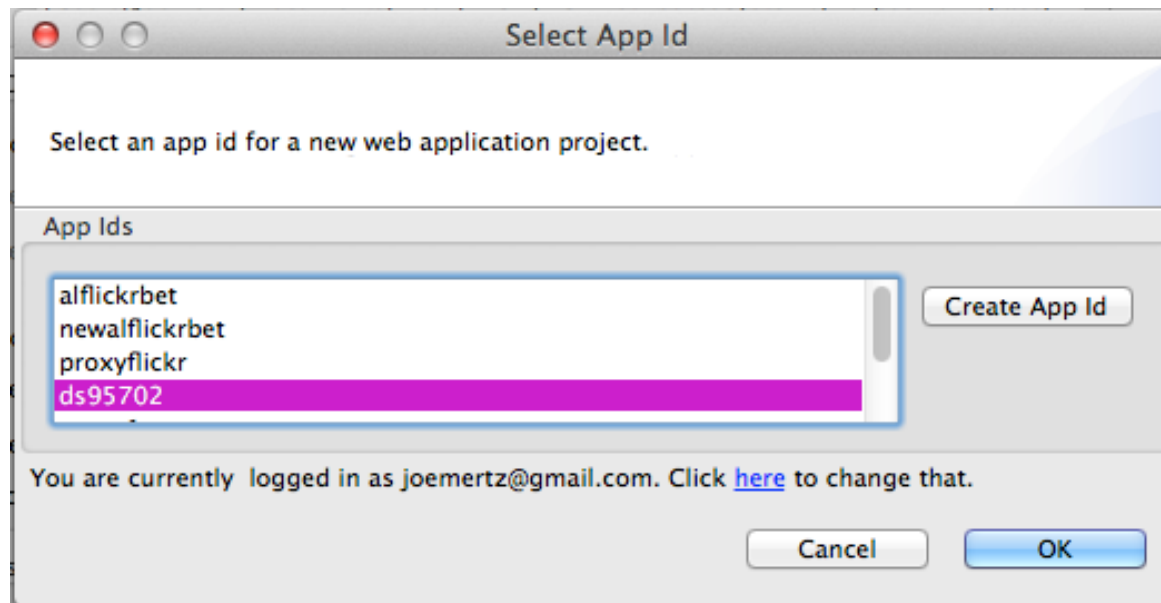
# Create a Web Application Project

Authentication  
will look something  
like this.  
"Accept"



# Create a Web Application Project

Once you are authenticated, you will see your list of applications you created on Google App Engine



Select the app id you want to link to this application and click OK

# Create a Web Application Project

The screenshot shows the 'New Web Application Project' dialog box in the Eclipse IDE. The dialog is titled 'New Web Application Project' and has a Google logo in the top right corner. It contains several sections for configuring the project:

- Create a Web Application Project**: Subtitle 'Create a Web Application project in the workspace or in an external location'.
- Project name:** 'HelloWorld'.
- Package: (e.g. com.example.myproject)**: 'mertz.helloworld'.
- Location**:
  - ☒ Create new project in workspace
  - ☐ Create new project in:
  - Directory:** '/Users/JoeMertz/Documents/Desktop/MISM/DS-DistributedSystems/EclipseWorkspace/HelloWorld' (partially visible).
- Google SDKs**:
  - ☐ Use Google Web Toolkit
    - ☒ Use default SDK (GWT - unknown version)
    - ☐ Use specific SDK: 'GWT - unknown version'.
  - ☒ Use Google App Engine
    - ☒ Use default SDK (App Engine (9) - 1.8.6)
    - ☐ Use specific SDK: 'App Engine - unknown version'.
- Identifiers for Google App Engine**:
  - ☐ Leave App Id field blank
  - ☒ Use App Id: 'ds95702'.
- Google Apps Marketplace**:
  - ☐ Add support for listing on Google Apps Marketplace
- Sample Code**:
  - ☒ Generate project sample code

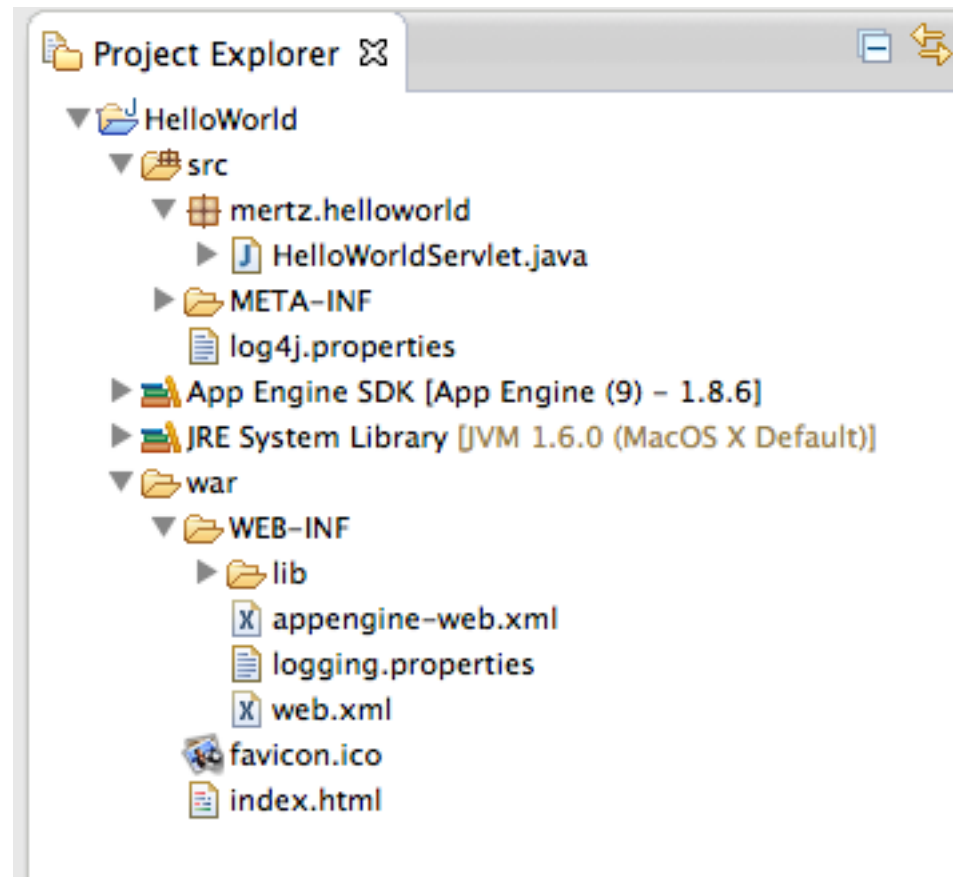
At the bottom, there are 'Cancel' and 'Finish' buttons. A blue arrow points from the 'Finish' button to the text 'Click Finish' on the right. Another blue arrow points from the 'Use App Id' field to the text 'The app id should now be included.' on the right.

The app id  
should now be  
included.

Click Finish



# Familiar Directory Structure



# Servlet

```
HelloWorldServlet.java
1 package mertz.helloworld;
2
3 import java.io.IOException;
4
5
6 @SuppressWarnings("serial")
7 public class HelloWorldServlet extends HttpServlet {
8     public void doGet(HttpServletRequest req, HttpServletResponse resp)
9         throws IOException {
10         resp.setContentType("text/plain");
11         resp.getWriter().println("Hello, world");
12     }
13 }
14
```

- Same standard servlet code
- Eclipse does not provide ProcessRequest
  - Different than the default in NetBeans
- Only an example of doGet
  - You could implement doPost, doPut, doDelete, etc.

# Deployment Descriptor (web.xml)

- GAE installs Jetty
- Jetty is a FOSS web container
- Works with Eclipse and Google App Engine Plugin
- Does not use @WebServlet annotation
  - Must map URL to servlet in web.xml

```
web.xml
1 <?xml version="1.0" encoding="utf-8" standalone="no"?>
2 <web-app xmlns="http://java.sun.com/xml/ns/javaee"
3         xmlns:web="http://java.sun.com/xml/ns/javaee/web-app_2_5.xsd"
4         xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
5         version="2.5"
6         xsi:schemaLocation="http://java.sun.com/xml/ns/javaee
7                             http://java.sun.com/xml/ns/javaee/web-app_2_5.xsd">
8     <servlet>
9         <servlet-name>HelloWorld</servlet-name>
10        <servlet-class>mertz.helloworld.HelloWorldServlet</servlet-class>
11    </servlet>
12    <servlet-mapping>
13        <servlet-name>HelloWorld</servlet-name>
14        <url-pattern>/helloworld</url-pattern>
15    </servlet-mapping>
16    <welcome-file-list>
17        <welcome-file>index.html</welcome-file>
18    </welcome-file-list>
19    <servlet>
20        <servlet-name>SystemServiceServlet</servlet-name>
21        <servlet-class>com.google.api.server.spi.SystemServiceServlet</servlet-class>
22    <init-param>
23        <param-name>services</param-name>
24        <param-value/>
25    </init-param>
26    </servlet>
```

Replicate servlet & servlet-mapping when adding servlets.

servlet-mapping maps a URL to a servlet-name

servlet maps the servlet name to the servlet-class.

# Run on local server

Click  or Run As->Web Application

Stop

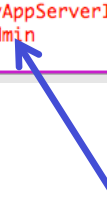


```
Console
HelloWorld [Web Application] /System/Library/Frameworks/JavaVM.framework/Versions/1.6.0/Home/bin/java (Mar 18, 2012 6:35:49 PM)
Mar 18, 2012 10:35:52 PM com.google.apphosting.utils.jetty.JettyLogger info
INFO: Logging to JettyLogger(null) via com.google.apphosting.utils.jetty.JettyLogger
Mar 18, 2012 10:35:52 PM com.google.apphosting.utils.config.AppEngineWebXmlReader readAppEngineWebXml
INFO: Successfully processed /Users/JoeMertz/Documents/Desk/MISM/DS-DistributedSystems/EclipseWorkspace/HelloWorld/war/WEB-INF/appengine-web.xml
Mar 18, 2012 10:35:52 PM com.google.apphosting.utils.config.AbstractConfigXmlReader readConfigXml
INFO: Successfully processed /Users/JoeMertz/Documents/Desk/MISM/DS-DistributedSystems/EclipseWorkspace/HelloWorld/war/WEB-INF/web.xml
Mar 18, 2012 6:35:55 PM com.google.appengine.tools.development.DevAppServerImpl start
INFO: The server is running at http://localhost:8888/
Mar 18, 2012 6:35:55 PM com.google.appengine.tools.development.DevAppServerImpl start
INFO: The admin console is running at http://localhost:8888/_ah/admin
```

local URL

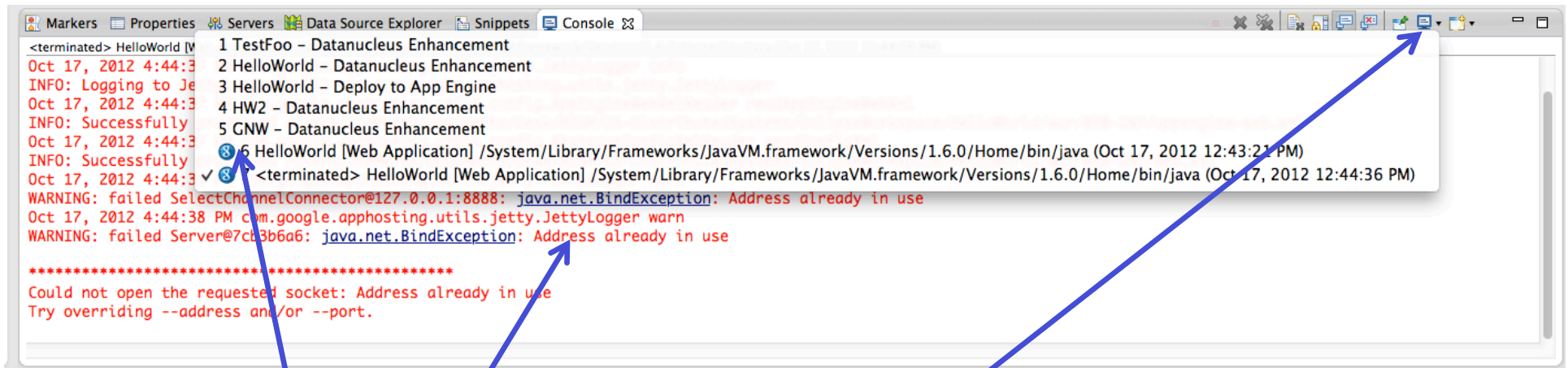


local admin  
URL



# Address already in use

Unlike NetBeans, Eclipse "Run" deploys, it does not redeploy.  
So it does not stop your previous deployment.



If you get "Address already in use"

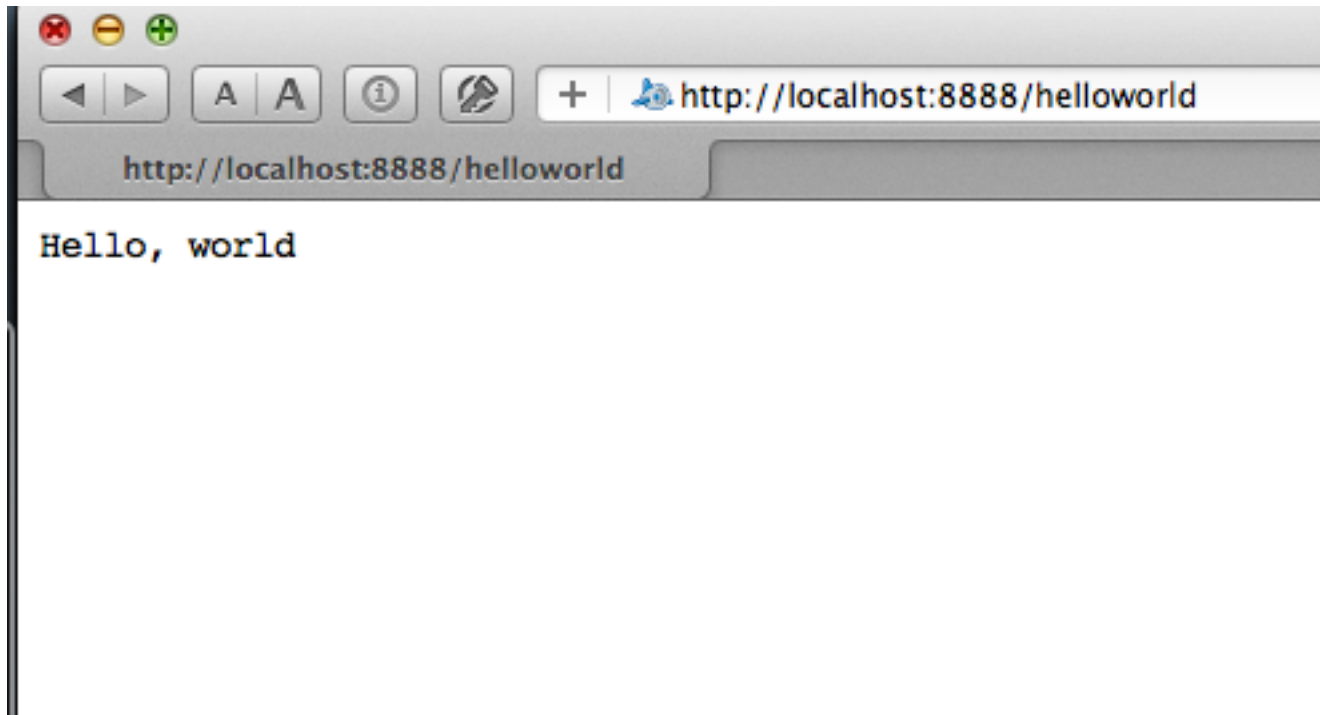
Click on the right of the "Display Selected Console" icon

Select the console of the already-running web app.

Then hit

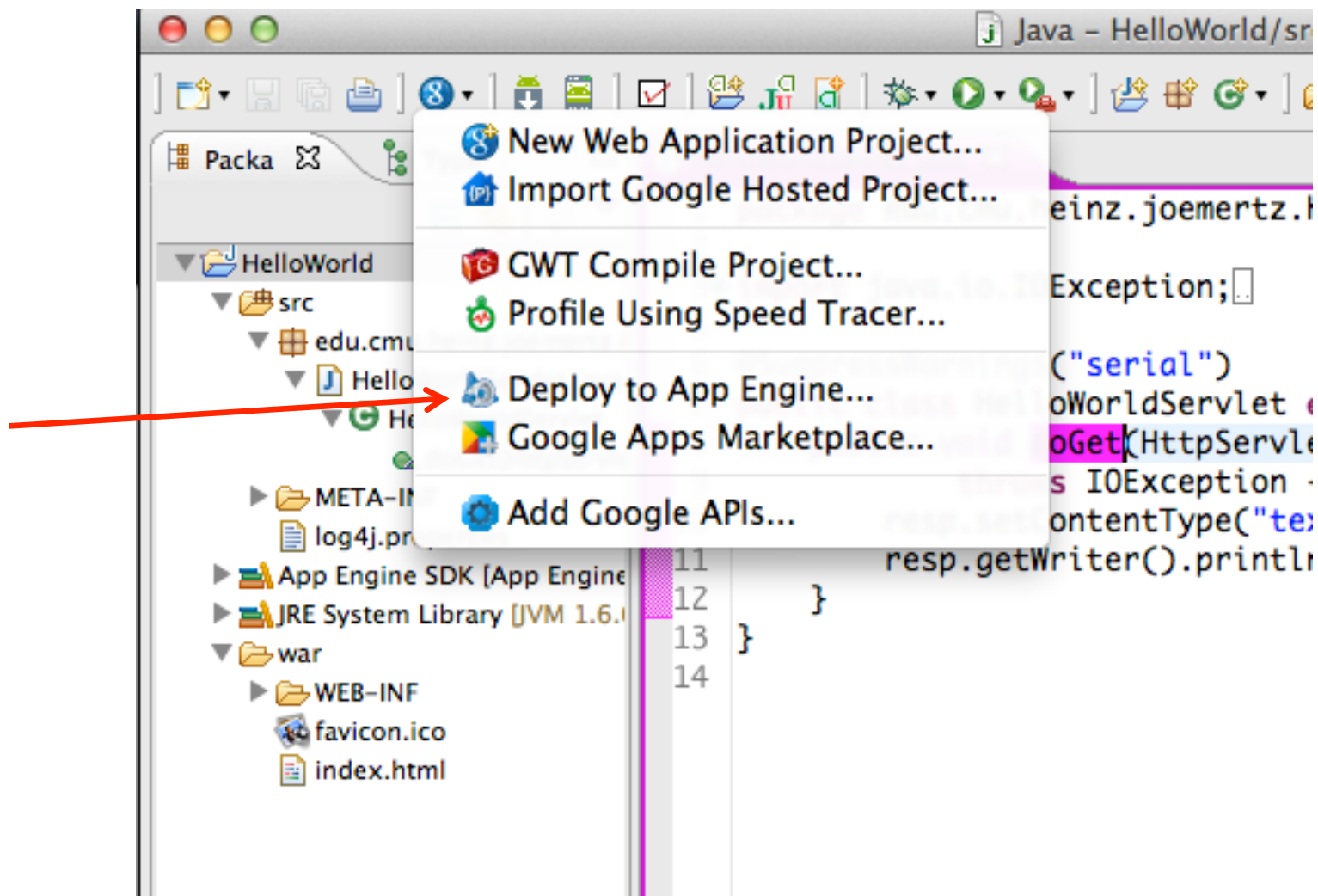


# Servlet on local server

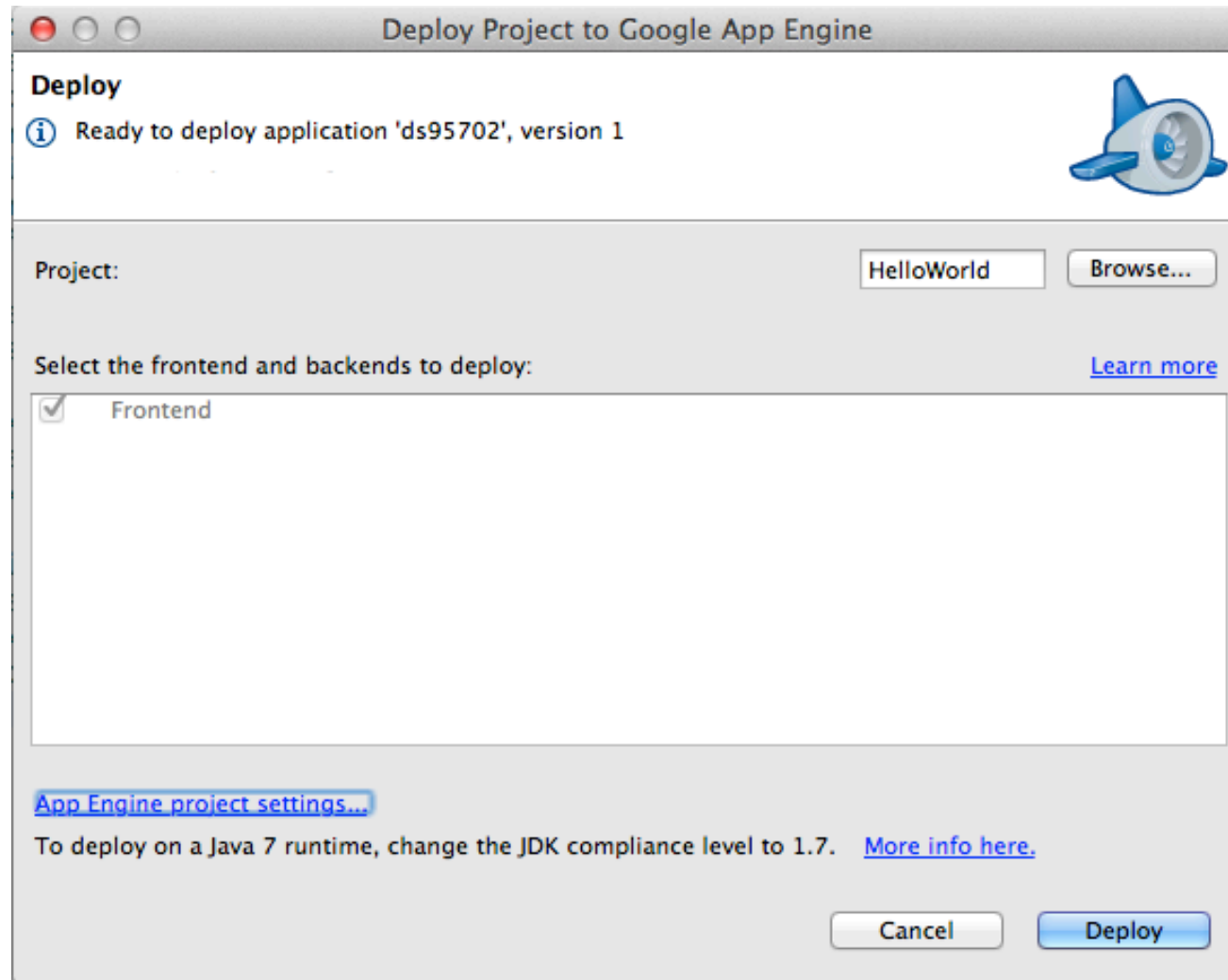




# Deploy application to GAE



# Deploy



# Typical deployment messages

```
Console
HelloWorld - Deploy to App Engine

----- Deploying frontend -----

Preparing to deploy:
  Created staging directory at: '/var/folders/mz/s5_k2czd0qj4cfs993bwm6_00000gq/T/appcfg2251144913474145146.tmp'
  Scanning for jsp files.
  Scanning files on local disk.
  Initiating update.
  Cloning 2 static files.
  Cloning 19 application files.

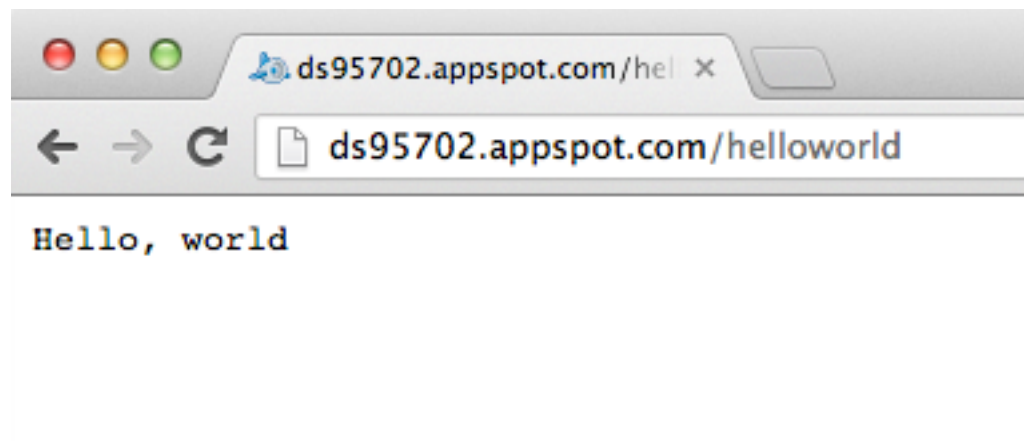
Deploying:
  Uploading 2 files.
  Uploaded 1 files.
  Uploaded 2 files.
  Initializing precompilation...
  Sending batch containing 2 file(s) totaling 2KB.
  Deploying new version.

Verifying availability:
  Will check again in 1 seconds.
  Will check again in 2 seconds.
  Will check again in 4 seconds.
  Closing update: new version is ready to start serving.

Updating datastore:
  Uploading index definitions.

Deployment completed successfully
```

# Servlet on Google App Engine



# Exception on GAE

- If you get an exception when running your application on GAE:
  - Go to <https://appengine.google.com/>
  - Select your application (on left)
  - Click on "Logs" (on left)
  - Expand the top log error, to see the full stack trace.

# GAE Lab

1. Get HelloWorld working locally and on GAE
  - Follow the screen shot slides
2. Deploy one of your servlets: Project 1, Task 2, 3, or 4
  - Not Project1Task1, sun.misc.BASE64Encoder is not available
  - Task 4 is the most challenging and most fun
  - Create files in Eclipse
  - Copy the code into those files
    - Remove @WebServlet's and add mappings to web.xml
  - Test on localhost
  - Deploy to GAE
3. To get credit, fill in the form at:
  - <http://tinyurl.com/95702gae>
  - For this lab only, you do **not** need to show a TA
  - As always, lab is due before class in one week.



# TimeZone problems

- If you get the runtime exception:
  - "Unable to restore the previous TimeZone".
  - Add the following in run configuration:
    - "-Dappengine.user.timezone=UTC"
  - I.e.
    - Go to "run configuration",
    - Under the "Arguments" tab add to VM arguments:
      - "-Dappengine.user.timezone=UTC"

# Unable to update app: Failed to compile jsp files.

- If you run into a problem with Eclipse recognizing .jsp files, you might need to let Eclipse know about the location of the JDK. Check out this solution:

<http://stackoverflow.com/questions/3112497/jsp-file-not-working-for-google-app-engine-guestbook-tutorial>