

Go + Google AppEngine

Sathish VJ

Goal

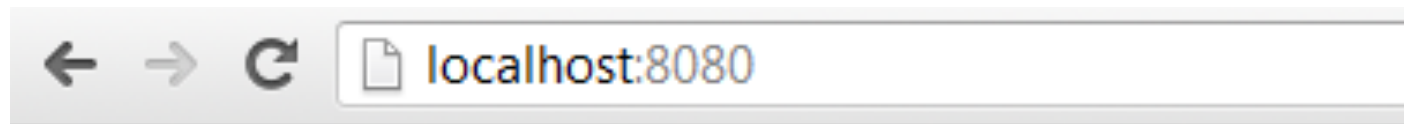
- * Introduce to you the main parts in writing and configuring a Go webapp for Google AppEngine
- * Familiarize you with app configuration
- * Templates for html
- * Using the context and logging
- * Deploying and monitoring your app
- * Storing data in the Google Datastore
- * Retrieving data from the Google Datastore

Setting up AppEngine for Go

- Download the zip file for your platform from:
<https://developers.google.com/appengine/downloads>
- Extract to known location
 - say, .../coding/golang/appengine
- Add that location to your PATH
 - you will be running tools like [dev_appserver.py](#) and [appcfg.py](#) from there)
- AppEngine tools are in Python 2.7.*
 - Install from <http://www.python.org/getit/> or using *apt-get*

Try one demo program

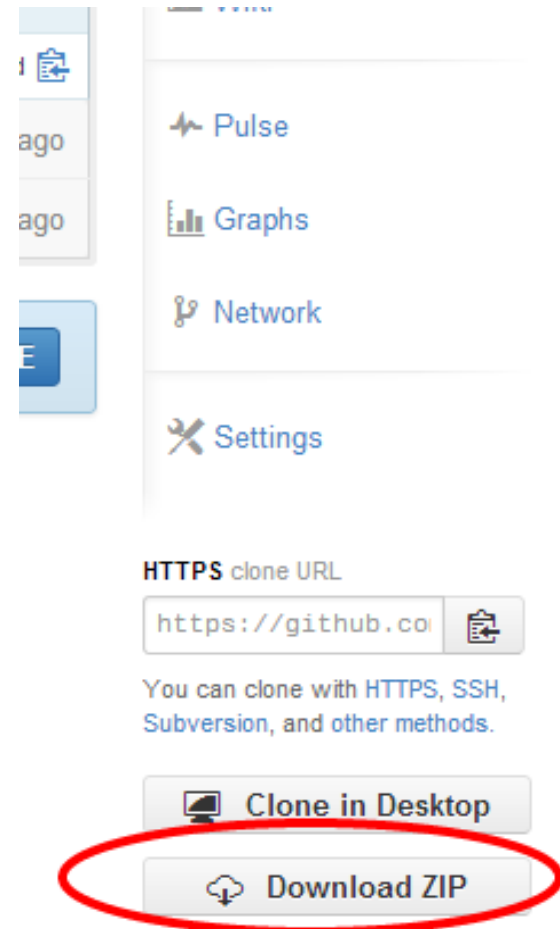
- Go to[/coding/golang/appengine/demos](#)
- Run the app using the command below
 - [dev_appserver.py helloworld](#)
- In your browser, go to <http://localhost:8080>
- If you see a page as shown below, all's good



Hello, World! 세상아 안녕!

Download workshop source code

- Go to <https://github.com/sathishvj/golang-workshops>
- Clone the project ... (OR)
- Download the Zip file



1 - yaml file for app configuration

- `app.yaml` is read by the AppEngine runtime to configure your app
- It has to be in the root folder of your app

```
application: myhelloworld-app-id
version: 0-1
runtime: go
api_version: go1

handlers:
- url: /*
  script: _go_app
```

1 - yaml file for app configuration

The unique id for your app as identified by the AppEngine.

```
application: myhelloworld-app-id
```

```
version: 0-1
```

```
runtime: go
```

Use the Go runtime (and not Java, PHP, Python).

```
api_version: go1
```

Use the 1.* version of Go.

```
handlers:
```

All the url routing information goes here.

```
- url: /*
```

All incoming requests are handled by ...

```
script: _go_app
```

... the go application, starting at init() function.

1-ex Hands-on experience

- Go to workshop code: [golang-workshops/appengine-beginner](#)
- In the file [1-ex/app.yaml](#) ...
- Add in the right parts where there is a “TODO” comment
- Run the app with: [dev_appserver.py 1-ex](#)
- Check <http://localhost:8080> on browser

2 - yaml file for static files

- The handlers are processed in lexical order
 - the redirection should go from the more specific ones to the more generic ones

```
handlers:  
- url: /images  
  static_dir: images  
  
- url: /*  
  script: _go_app
```

2 - yaml file for static files

```
handlers:
```

Redirect all requests to the url “/images” to the directory “images”.

```
- url: /images
```

```
  static_dir: images
```

Everything else, let the go application handle it.

```
- url: /*
```

```
  script: _go_app
```

3 - AppEngine Context and logging debug information

- Each application has a context in which it executes. Get it with `appengine.NewContext`
- ??

```
import (  
    "appengine"
```

```
func onRoot(w http.ResponseWriter, r *http.Request) {  
    c := appengine.NewContext(r)  
  
    c.Debugf("Within the onRoot function ... ")
```

Deploying and monitoring on Google's AppEngine Servers

- <https://appengine.google.com/>
- “Create Application”

Google app engine

sat

My Applications

Application	Title	Store
greetings-gdg-blr	Greetings GDG Bangalore	High
Create Application		

Deploying and monitoring on Google's AppEngine Servers

Create an Application

You have 6 applications remaining.

Application Identifier:

.appspot.com

Yes, "mytestapp-123" 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:

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. 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 (e.g. an HR tracking tool) that is only accessible to accounts on your Google Apps domain. This option cannot be changed

☐ **(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.

Deploying and monitoring on Google's AppEngine Servers

- Deploy to server with ...
- `appcfg.py <app folder name>`
- Provide your username, password ...
- AppEngine will
 - read the app.yaml file,
 - upload all the files,
 - create an instance based on version
 - make it available on appspot.com
- Eg. <http://0-2.myhelloworld.appspot.com>

Deploying and monitoring on Google's AppEngine Servers



Application: [High Replication] ▼

Main

[Dashboard](#)

[Instances](#)

[Logs](#)

[Versions](#)

[Cron Jobs](#)

[Task Queues](#)

[Quota Details](#)

Data

[Datastore Indexes](#)

[Datastore Viewer](#)

[Datastore Statistics](#)

[Blob Viewer](#)

[Prospective Search](#)

[Text Search](#)

[Datastore Admin](#)

[Memcache Viewer](#)

Administration

[Application Settings](#)

[Permissions](#)

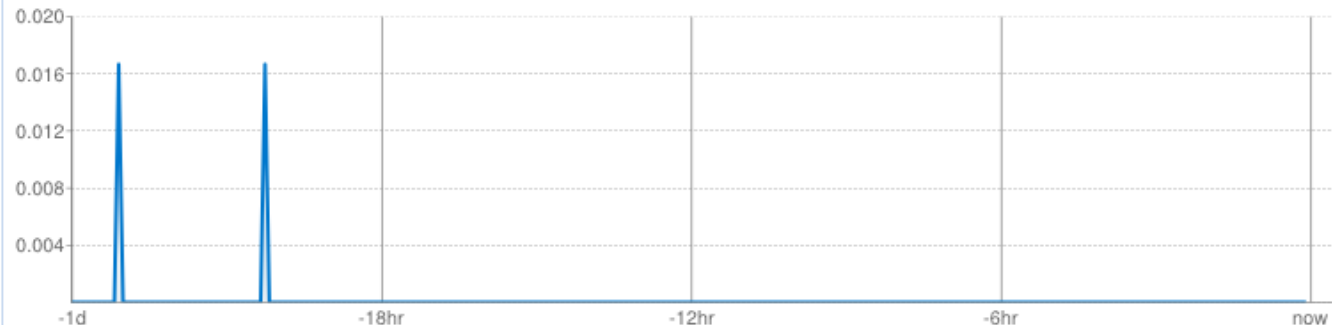
[Blacklist](#)

[Admin Logs](#)

Version: ▼

Charts ?

▼



Instances ?

App Engine Release

Total number of instances

Average QPS*

Ave

Billing Status: Free - [Settings](#)

Resource

Usage

Frontend Instance Hours

0%

0.00 of 28.00 In

Backend Instance Hours

0%

0.00 of 9.00 Inst

Datastore Stored Data

0%

0.00 of 1.00 GB

Logs Stored Data

0%

0.00 of 1.00 GB

Task Queue Stored Task Bytes

0%

0.00 of 0.49 GB

Blobstore Stored Data

0%

0.00 of 5.00 GB

Deploying and monitoring on Google's AppEngine Servers

Main

[Dashboard](#)[Instances](#)[Logs](#)[Versions](#)[Cron Jobs](#)[Task Queues](#)[Quota Details](#)

Data

[Datastore Indexes](#)[Datastore Viewer](#)[Datastore Statistics](#)[Blob Viewer](#)[Prospective Search](#)[Text Search](#)[Datastore Admin](#)[Memcache Viewer](#)

Administration

[Application Settings](#)[Permissions](#)[Blacklist](#)[Admin Logs](#)

Billing

[Billing Status](#)

Version: 1

Total Logs Storage: 1 KByte spanning 23 hours (0% of the Retention limit) Total Logs Storage for Ve

Show: ☒ All requests ☐ Logs with minimum severity: Error[Options](#)Filter: ☒ Regex ☐ LabelsRegex filters are case-sensitive [regular expressions](#), e.g. `^[tT]raceback` or `hello.*rid`Before: ☒ Now☐ 2013-08-05 09:08:47

Rows: 20

Search

Tip: Click a log line to show or hide its details.

< Prev 20 1-10 Next 20 > (Top: 20:16:07 ago)

	2013-08-04 12:52:40.883	/favicon.ico	200	12ms	0kb	Mozilla/5.0 (Windows NT 6.1; WOW64) Apple
	2013-08-04 12:52:40.115	/	200	11ms	0kb	Mozilla/5.0 (Windows NT 6.1; WOW64) AppleWebKit/537
	2013-08-04 12:52:39.519	/sign	302	169ms	0kb	Mozilla/5.0 (Windows NT 6.1; WOW64) AppleWebKit
	2013-08-04 12:52:36.480	/favicon.ico	200	199ms	0kb	Mozilla/5.0 (Windows NT 6.1; WOW64) App
	2013-08-04 12:52:35.599	/	200	319ms	0kb	Mozilla/5.0 (Windows NT 6.1; WOW64) AppleWebKit/53
	2013-08-04 12:52:35.599	This request caused a new process to be started for your application, an				
	2013-08-04 10:02:21.574	/favicon.ico	200	13ms	0kb	Mozilla/5.0 (Windows NT 6.1; WOW64) Apple
	2013-08-04 10:02:21.009	/	200	183ms	0kb	Mozilla/5.0 (Windows NT 6.1; WOW64) AppleWebKit/53
	2013-08-04 10:02:20.391	/sign	302	194ms	0kb	Mozilla/5.0 (Windows NT 6.1; WOW64) AppleWebKit

Deploying and monitoring on Google's AppEngine Servers



Main

[Dashboard](#)

[Instances](#)

[Logs](#)

[Versions](#)

Version	Default	Deployed	Delete
1  instances 1.70 KBytes go instance_class: F1 api_version: go1	Yes		Cannot delete default version.
Make Default			

Main

[Dashboard](#)

[Instances](#)

[Logs](#)

[Versions](#)

[Cron Jobs](#)

[Task Queues](#)

[Quota Details](#)

Data

[Datastore Indexes](#)

[Datastore Viewer](#)

Query

Create

By kind: [Greeting](#) kinds as of 0:00:48 ago

[Options](#)

Greeting Entities

◀ Prev 20 1-2 Next 20 ▶

<input type="checkbox"/> ID/Name	Content	Date
<input type="checkbox"/> id=2001	Hello	2013-08-04 19:52:39.3524
<input type="checkbox"/> id=3001	There!	2013-08-05 16:12:22.6144

Delete

Flush Memcache

Local server monitoring

- http://localhost:8080/_ah/admin/

Google App Engine

dev~helloworld Development Console

Datastore Viewer

[Datastore Indexes](#)

[Datastore Stats](#)

[Interactive Console](#)

[Memcache Viewer](#)

[Task Queues](#)

[Cron Jobs](#)

[XMPP](#)

[Inbound Mail](#)

[Backends](#)

[Text Search](#)

Datastore Viewer

Entity Kind:

List Entities

Create New Entity

[Select different namespace](#)

4 - Templates

- Static HTML/text content that can be merged with dynamic data

```
{{define "root"}}  
<html>  
<body>  
  <h3>Enter info:</h3>  
  <form action="">  
    <input name="info" type="text">  
    <input type="submit">  
  </form>  
</body>  
</html>  
{{end}}
```

```
import (  
  "html/template"
```

```
t, err := template.ParseFiles("tmpl/root.tmpl")  
err = t.ExecuteTemplate(w, "root", nil)
```

5a - Writing to the Google Datastore

- Datastore is 'schema-less'
- Define your 'entity' as a **struct**
- A 'key' identifies a unique entity
- 'Put' and 'Get' values of that type

```
import (  
    "appengine"  
    "appengine/datastore"
```

```
type Info struct {  
    Data string  
    Time time.Time  
}
```

```
key := datastore.NewIncompleteKey(c, "Info", nil)  
newInfo := Info{s, time.Now(), }  
datastore.Put(c, key, &newInfo)
```

5b - Reading from the Google Datastore

```
q := datastore.NewQuery("Info").Order("-Time")
var infos []Info
q.GetAll(c, &infos)
t.ExecuteTemplate(w, "root", infos)
```

Here, the values in 'infos' == the dot in the template.
Therefore, '.Data' will contain the value of the member field Info.Data

```
<h2>Existing Infos</h2>
{{range .}}
{{.Data}} at {{.Time.Format "2006-01-02 3:04" }}<br>
{{end}}
```

Thank You

Sathish VJ