

PGIBot installing manual

1 - Uploading the application

- Once you have your own Google App Engine application registered (see [1] to know how to do that) you will have to change some files in order to make it work for your new site. It basically consists on replacing the string of characters 'pgibot' with your new app name ('myrobot' in this manual) and some other little changes. Search for the old line in your documents:

- app.yaml:

```
- old - application: pgibot
      new - application: myrobot
```

- pgibot.py:

```
- old - GADGET_BASE = 'http://pgibot.appspot.com'
      new - GADGET_BASE = 'http://myrobot.appspot.com'
- change the variable called 'url2' to the url where your REST import by POST
  request is placed (it should probably end by
  'admin/settings/rdfme/rest/import')
- one line under 'url2' in the variable 'values2' change the 'name' dictionary key's
  value, 'admin', by your desired user in PGI and the password. Also change the
  'pass' dictionary key's value to your user's password, encrypted by MD5 algorithm.
```

- static/gadget.xml:

```
- old - <script type="text/javascript"
      src="http://pgibot.appspot.com/static/jquery-1.4.4.js"></script>
      new - <script type="text/javascript"
      src="http://myrobot.appspot.com/static/jquery-1.4.4.js"></script>
```

- static/installer.xml:

```
- the string 'pgibot' is placed in several parts of this file. Just replace them with
  'myrobot'
```


- Having done the previous steps, you need to know exactly where is your google_appengine folder (the one we provided or the one you downloaded from Google) and the path to your application (inside it you should see the *static* and *waveapi* folders and the files named *app.yaml*, *pgibot.py* and *storage.py*, i.e. the pgibot folder which we provided). Once you know this, open your terminal and write the following line:

```
/my_appengine_folder/appcfg.py update /my_pgibot_folder/
```

it should work in any operating system, with its own syntax. If it doesn't work, try to write 'python' or 'python2.7' before in the same line or the python's binary executable file path if you don't have it in your \$PATH environment variable. **Note that you need python2.7** to make this work. Python 2.6 could work, but 2.7 is the latest working version for this application. Anyway, if everything's OK, it should ask for your Google account user and password in order to upload the files. **It must be the same account you used in App Engine.**

2 - Installing and running the application

Once you have your application online, it's time to install it on Google Wave. There are a few steps to make this work:

- Go to Google Wave and log in.
- At the left panel, open 'EXTENSIONS > All', look for the Extension Installer, open it and push 'Install'.
- Now, at the same left panel, go to the first option, 'Inbox', and create a new wave. If you are currently editing the wave, you should see the Extension Installer icon in the toolbar. Push it, paste the installer.xml url ('<http://myrobot.appspot.com/static/installer.xml>') and press 'Install'. An installing dialog should appear inside the wave. Press 'Install' again to finally install the application. If you ever want to remove the application or the Extension Installer you must go to the left panel's option 'Settings' > 'Extension settings' and look for the one you want to uninstall. Of course, you need to press its 'Uninstall' button.
- If you want to use the application, just create a new wave, and when editing it, look for the Drupal  icon at the toolbar and push it. You will see that a new participant has been added. If you want to create a new application's form, remove that participant and push the icon again. If you haven't remove the old form it's OK, the robot will manage both the old one and the new one.

[1] <http://code.google.com/appengine/docs/python/gettingstarted/uploading.html>