


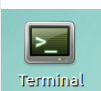
You are in: [DevOps Technical Workshop Wiki](#) > [V2 Lab VII - The Cloud Awakens](#) > V2 Lab 7 - Steps 2A & 2B

## V2 Lab 7 - Steps 2A & 2B

[Like](#) | Updated 16 June 2017 by [Katamneni, Krishna Sravya](#) | Tags: *None*

NOTE - The Mongodb Service on Bluemix is no longer free and the database used by the TODOs application is Mongo. This means a production deploy on Bluemix is not possible unless you pay for it, but the pipelines setup below give a good insight into what is possible in IBM DevOps Services.

 Anything in this box needs to be edited in Sublime Text

 This box contains lists of commands that should be executed in order on the Terminal

### (A) Create a repository on Jazz hub

**The objective of this lab is to create a new remote to host your code outside of the VM. This is allow us collaborate on the code & share it with other teams. This also leads us into the pipelines on Bluemix**

- On <https://hub.jazz.net/> log in and *Create Project*. Call the new project `todolist` and select *Create a repo on Bluemix*.
- Un-tick the *Add features for Scrum development* option and tick the *Make this a Bluemix Project* box. On the drop downs that are shown, select the region you want to deploy the app to eg UK. The Organisation and Space will have to be selected too. For example, I am using the `MyHome` org and `MyHomeSpace` space

Select a Bluemix space to bill your services to:

Region

Organization

Space

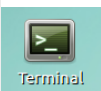
These selections can be changed later in the options for your Project Settings.

- On the page that loads, follow the instructions for setting up an existing repository. For my project they were as follows

To push the contents of an existing repository, type these commands from the repository directory:

```
git remote add new_repo https://hub.jazz.net/git/springdo/todolist
git push -u new_repo master
```

Back on the terminal in the devops vm, go to the `/share/todolist` directory and add the new remote where `<USER>` is the one for the Jazz account



```
git remote add jazz https://hub.jazz.net/git/<USER>/todolist
git push -u jazz --all
```

```
[→ todolist git:(develop) ✕ git remote add jazz https://hub.jazz.net/git/springdo/todolist
[→ todolist git:(develop) ✕ git push -u jazz --all
[Username for 'https://hub.jazz.net': springdo
[Password for 'https://springdo@hub.jazz.net':
Counting objects: 443, done.
Compressing objects: 100% (207/207), done.
Writing objects: 100% (443/443), 89.88 KiB | 0 bytes/s, done.
Total 443 (delta 215), reused 438 (delta 212)
remote: Resolving deltas: 100% (215/215)
remote:
remote:
remote: Processing changes: refs: 2, done
To https://hub.jazz.net/git/springdo/todolist
 * [new branch]      develop -> develop
 * [new branch]      master -> master
Branch develop set up to track remote branch develop from jazz.
Branch master set up to track remote branch master from jazz.
→ todolist git:(develop) ✕
```

- Refresh the page on Jazz hub and you should see the code in the cloud.

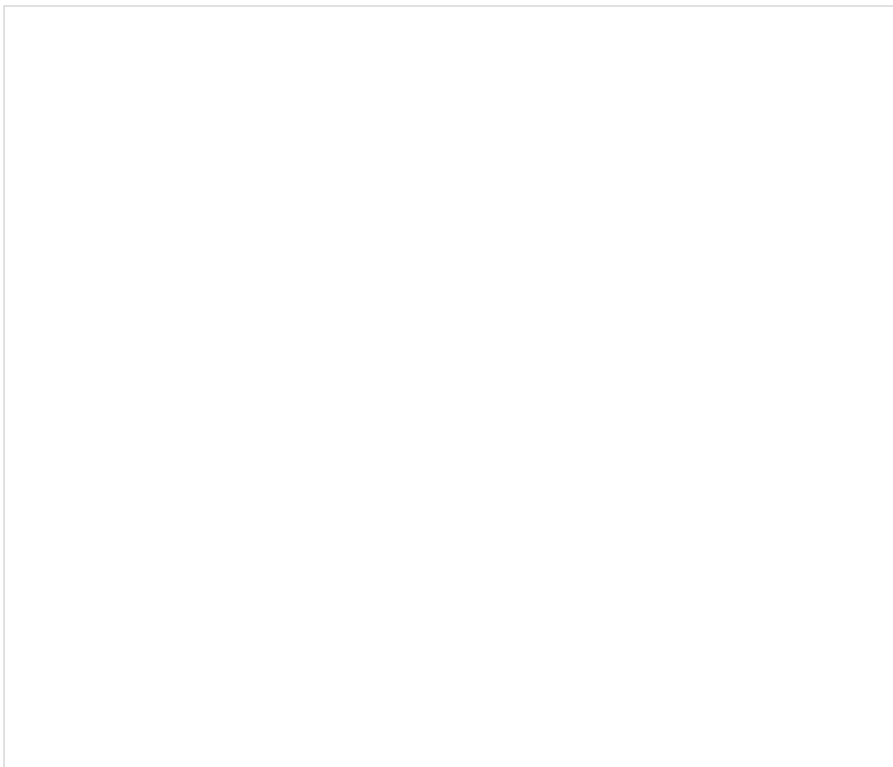
## (B) Create a simple Build and Deploy pipeline on Bluemix

***The objective of this lab is to build a simple pipeline on Bluemix and demonstrate the power of the cloudified version in comparison to the Jenkins pipelines in previous labs***

- On the Jazz hub UI, select *Build and Deploy* in the upper right hand corner of the UI for the project



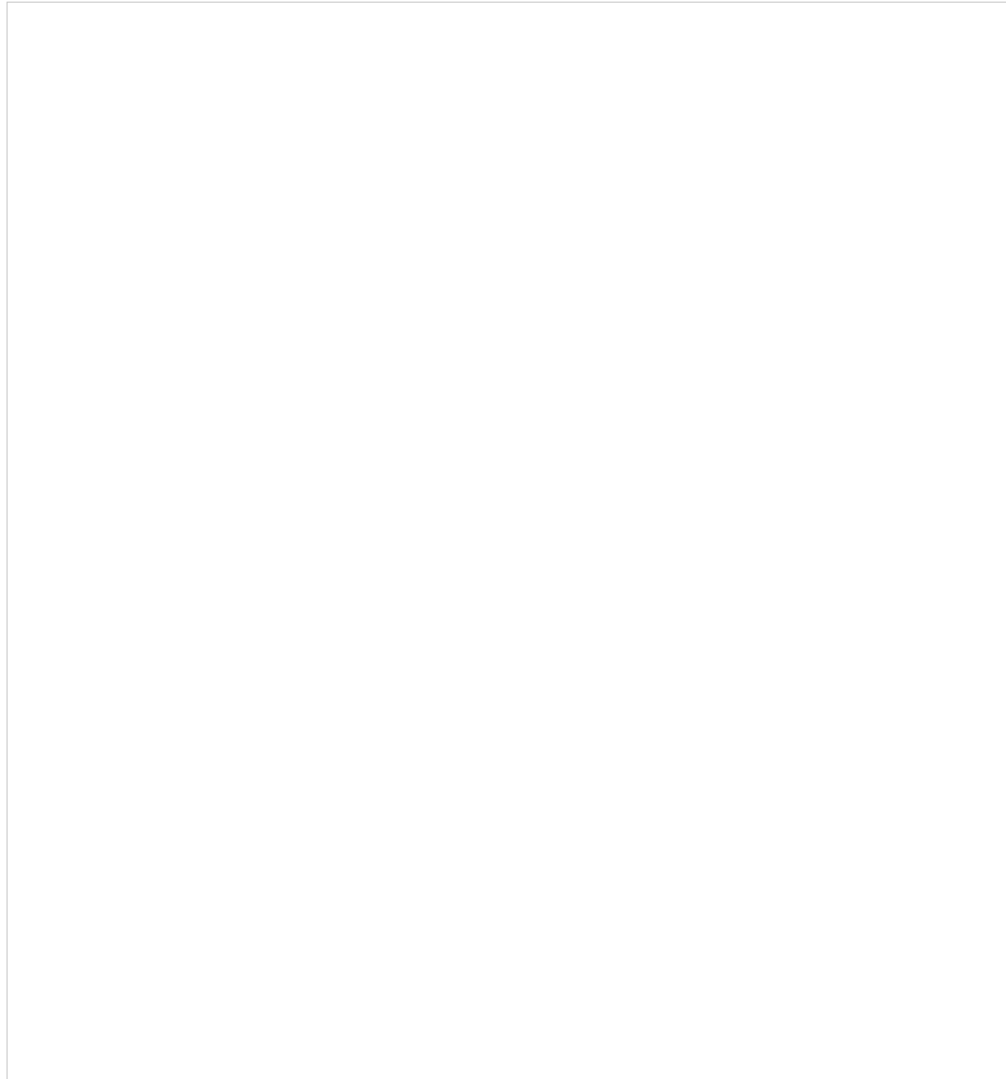
- Add stage to the pipeline and give it a sensible name such as `todolist-build`. Ensure the input for this stage is set to *SCM Repository*. The DevOps services should auto-fill out the rest. Set the *Stage Trigger* to be Run jobs whenever a change is pushed to Git to ensure builds are triggered on check-in.



- On *Jobs* tab for the stage being configured add a *Build* step. Give it a name such as `grunt-build` and set the *Builder Type* to be

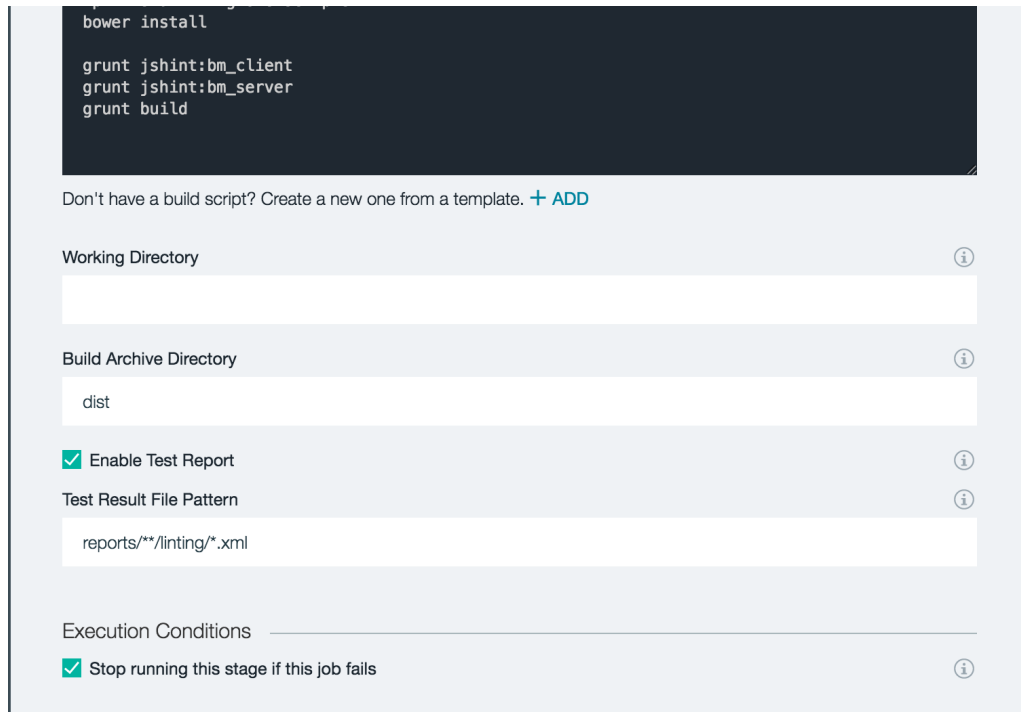
*Grunt.* In the shell command that is given, uncomment the path variable to use Node JS 4.4. and add `grunt build` to the task as shown below. In the **Build Archive Directory** set to `dist`.

```
export PATH=/opt/IBM/node-v4.2/bin:$PATH
npm install -g bower
npm install --ignore-scripts
bower install
```



– Add linting and the build step to our stage by adding these above the lines below. Enable Test Reporting and specify `reports/**/*.linting/*.xml` as the path

```
grunt jshint:bm_client
grunt jshint:bm_server
grunt build
```



The screenshot shows a build configuration interface. At the top, a dark terminal window displays the following commands:

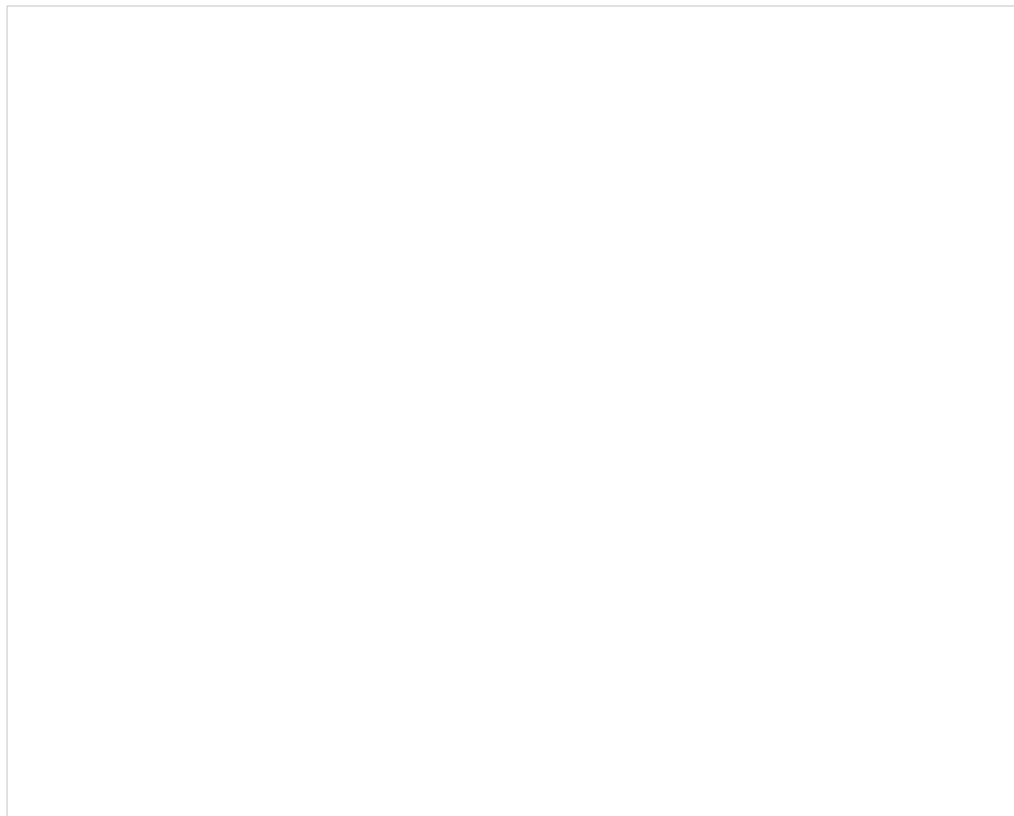
```
bower install  
  
grunt jshint:bm_client  
grunt jshint:bm_server  
grunt build
```

Below the terminal, a link says: "Don't have a build script? Create a new one from a template. [+ ADD](#)".

The configuration fields are as follows:

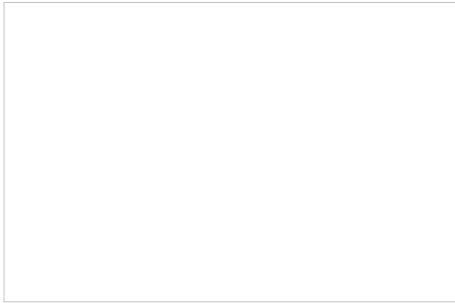
- Working Directory:** An empty text input field.
- Build Archive Directory:** A text input field containing the value "dist".
- Enable Test Report:** A checkbox that is checked.
- Test Result File Pattern:** A text input field containing the value "reports/\*\*/\*.linting/\*.xml".
- Execution Conditions:** A section with a checkbox "Stop running this stage if this job fails" which is checked.

- Add stage and give it a sensible name such as `todolist-deploy-ci`. On the input for this stage, set the *Input Type* to `Build Artifacts`, the *Stage* to `todolist-build` and the *Job* to `grunt-build`. This will probably be auto completed for you by DevOps Services

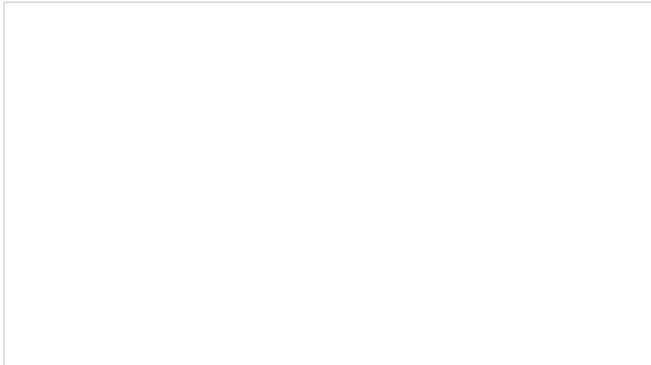


This is a large, empty rectangular box, likely representing the configuration area for a new stage as described in the instructions above.

- Hit the **JOBS** tab and add a *Deploy* phase. DevOps services will auto complete this again, but in the *Application Name* field set to something different eg `<your>-<name>-todolist-ci` as shown below. Hit Save



- Finally, a `manifest.yml` for Bluemix will be added to our git repo and pushed through our pipeline. In the root of the project directory ( `/share/todolist` ) back in the VM, change the `name` variable to the name of your application from the step before: eg `<your>-<name>-todolist-ci`



- Git commit the code and watch the build trigger on DevOps services.

Pipeline: All Stages

### todoapp-build

STAGE PASSED

LAST INPUT [Git URL](#)

Last commit by Donal Spring  
[Updating manifest for Bluemix](#) 8 min ago

**JOBS** [View logs and history](#)

✔ **grunt-build** Succeeded 3 min ago

LAST EXECUTION RESULT

grunt-build 9

### todolist-deploy-ci

STAGE PASSED

LAST INPUT Stage: todoapp-build / Job:...

grunt-build 9

**JOBS** [View logs and history](#)

✔ **Deploy** Succeeded 2 min ago

LAST EXECUTION RESULT

donal-spring-todolist-ci  
[donal-spring-todolist-ci.eu-gb.myblue...](#)  
[View runtime log](#)

grunt-build 9


+ ADD STAGE


- The deployed application will be available on `<your>-<name>-todolist-ci.eu-gb.mybluemix.net`  
eg <http://donal-spring-todolist-ci.eu-gb.mybluemix.net/>



- The Test scores and other information about the build can be accessed by hitting the *View Logs and History* item on the Stage homepage



LAST INPUT

 [Git URL](#)

 Last commit by Donal Spring 2 min ago  
[Fix for linting on bluemix](#)

JOBS

[View logs and history](#)

 **grunt-build** Running 13% 

- The Test scores are seen on the Test tab beside the logs

LOGS

**TESTS**

ARTIFACTS

Duration: 0 seconds

3 Pass

[jshint](#)


Name	Status
client/app/app.js	Passed
client/app/todos/todos.controller.js	Passed
client/app/todos/todos.controller.spec.js	Passed


- If you cannot see the logs but see the *Changes* tab as below, then select the name of the *JOBS* above this ie `grunt-build`

<NUMBER> Failed to show the logs, tests and artifacts generated from the build

10

Failed 14 minutes ago

STARTED Today at 9:38 PM DURATION 3 minutes, 57 seconds TRIGGERED BY  springdo

JOB  grunt-build 10 Failed

CHANGES

No Changes

### (C) Extension tasks

- Create an extension to the build phase for the todoapp-build that pushes code to Bluemix using the CLI. The CLI is installed in the slave containers for convenience.
- Integrate the pipeline into Slack for notifications on build failures.
- Extend the Bluemix pipelines with tests / jshint scores
- Look at <https://mlab.com/> for extending out to SI and PROD envs

## Comments

*There are no comments.*