

Connor Walters' Project Setup

Created by
Connor Walters
Oct 25, 2017

Getting Started

Instructions for getting started are already laid out well on the following pages. Follow the instructions in order:

1. [Create Cloud Account \(Do this first\)](#)

Create Cloud Account (Do this first)

Created by
JakeR

Last updated Feb 09, 2017

Recently JIRA released free development servers. These development servers are required for running our plugin. See <https://developer.atlassian.com/blog/2016/04/cloud-ecosystem-dev-env/> and <https://developer.atlassian.com/static/connect/docs/latest/guides/development-setup.html> for more information.

To register for an account, go to <http://go.atlassian.com/cloud-dev>

2. [Up and Running \(Do this second\)](#)

Up and Running (Do this second)

+2

Created by
JakeR

Last updated Apr 26, 2018 by
Brandon Jackson

Before doing this tutorial, see the *Create Cloud Account* how-to article.

Make sure your **ssh keys** are setup for Bitbucket. <https://confluence.atlassian.com/bitbucket/set-up-ssh-for-git-728138079.html>

Make sure **git** is downloaded. <https://git-scm.com/downloads>

Find the **Bitbucket repository** at <https://bitbucket.org/hasker/abouttime>

Open a terminal or command prompt. Next, go into a directory where you would like the project to be located.

```
cd ~/Documents          # This is a linux/mac command. Windows file paths are
different
```

Clone the repository. If you setup your keys correctly, the repository should download.

```
git clone git@bitbucket.org:hasker/abouttime.git
cd abouttime
git checkout dev
```

Next, **install the dependencies**

```
npm install
npm run-script install-global
```

Now we will need to create our environment file based on the sample. Simply copy the **.sample-env** file, and rename it to **.env** – You can also do:

```
cp .sample-env .env
```

Now open up the file with your favorite text editor and fill in your information. Don't worry, this is in our .gitignore file, so this information will remain private.

- JIRA_URL: This is the url for the development server that you registered for.
- JIRA_USERNAME: Your JIRA username (for your new cloud account)
- JIRA_PASSWORD: Your JIRA password (for your new cloud account)
- DB_HOST: This will probably be "localhost" unless you are working on the production server
- DB_NAME: This will be "time_logger"
- DB_USER: "root" is the default for when you install mysql
- DB_PASS: "" is the default for when you install mysql
- BITBUCKET_USERNAME: Retrieve the information from Dr. Hasker, as it's stored in a private confluence page
- BITBUCKET_PASSWORD: Same as above.

Note, the Bitbucket credentials are only necessary for testing the "Submit an issue" feature

Now **install MySQL**. This should be a simple google search and a couple commands. After installing, run the following commands

1. `mysql -u root` #There could be a different -u if you created another user

```
2. CREATE DATABASE time_logger;
3. exit
4. db-migrate up
```

Now we can start the nodejs-express server with:

```
npm start
```

If you are using an IDE, you can just run index.js instead.

In order to make our server publicly available (and accessible by JIRA cloud), we need to use ngrok.

```
ngrok http 8081
```

Next we need to create the personalized atlassian-connect.json file. Simply copy the atlassian-connect-sample.json file and rename it to atlassian-connect.json. This document is in the public folder. You can also copy using following command:

```
cp atlassian-connect-sample.json atlassian-connect.json
```

Now, change the **baseUrl** attribute in your atlassian-connect.json file. Set it as the **https** url provided by ngrok.

Go to your development JIRA environment (that you created in the "Create Cloud Account" How-To Article), and install the plugin.

Next, go to **Cog Icon → Add-ons → Manage Addons → Settings → Click "enable development mode"**

Then, install the plugin at **Cog Icon → Add-ons → Manage Addons → Upload add-on** and provide your ngrok url with atlassian-connect.json such as **https://e2347666.ngrok.io/atlassian-connect.json**

After it uploads, you should be able to **refresh your page** and see the add-on in your top navigation bar

Localhost

The app can also be used through localhost to test projects and sprints from the MSOE Jira (with ngrok you must create your own projects and sprints to test on your cloud account).

In order to do this you need to create a row in the database table called 'organization' This row must contain the base url: msoese.atlassian.net and the correct Jira and confluence secret keys ('jira_shared_secret' and 'confluence_shared_secret').

For this row, leave server_version, plugins_version, and deleted_at must be kept as NULL (default).

The updated_at and inserted_at should be changed to a date in the past such as: 2018-04-22 00:00:00 for April 22, 2018.

Now go to the project folder and find abouttime/middleware/orgloader. In this you may (or may not) need to un-comment the 3 lines shown below:

```
orgloader.run = function(req, res, next) {  
  req.jwt = {},  
  // if (process.env.ENV === "dev") {  
  //   host = process.env.JIRA_URL;  
  // } else  
  if (req.query.xdm_e) {  
    var host = req.query.xdm_e; //provided by jira  
  } else if (req.cookies.host) {  
    var host = req.cookies.host;  
  } else {  
    return next();  
  }  
  res.cookie('host', host, {expires: moment().add(1, 'days')});  
  //console.log("host", host);  
}
```

After you complete the instructions on both of those pages, you should have a working project.