MIRACLE
SOFTWARE SYSTEMS

Creating CRUD Application with Cloudant, IBM Bluemix and Node JS

AP Cloud Lab | Miracle Innovation Labs

Team Miracle

Miracle Software Systems, Inc.

November 9th, 2017

©2017 Miracle Software Systems, Inc.

Creating CRUD Application with Cloudant, IBM Bluemix and Node JS

Goal

In this lab we will guide you how to create a CRUD application with Cloudant using Node.js for backend, Bootstrap and HTML for frontend. You will then deploy the application to IBM Bluemix.

Pre-Requisites

The following installations will need to be completed for this lab to be run successfully,

- Account with IBM Bluemix
- Node.js and NPM Installed
- Cloudant account for storing data
- Text Editor such as Sublime Text (or) Notepad ++

Technology Involved

- Server Side - NodeJS
- Client Side Technologies (HTML, CSS, Bootstrap)
- Cloud Technologies – IBM Bluemix
- Database - IBM Cloudant

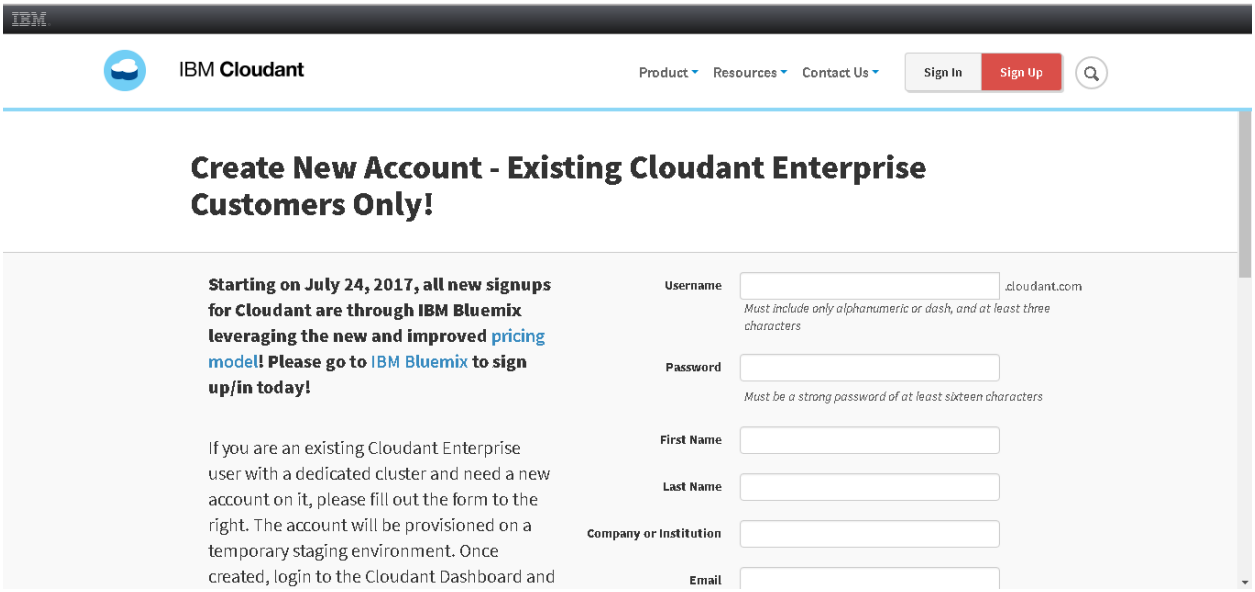
Lab Steps

Let's get started with the lab!

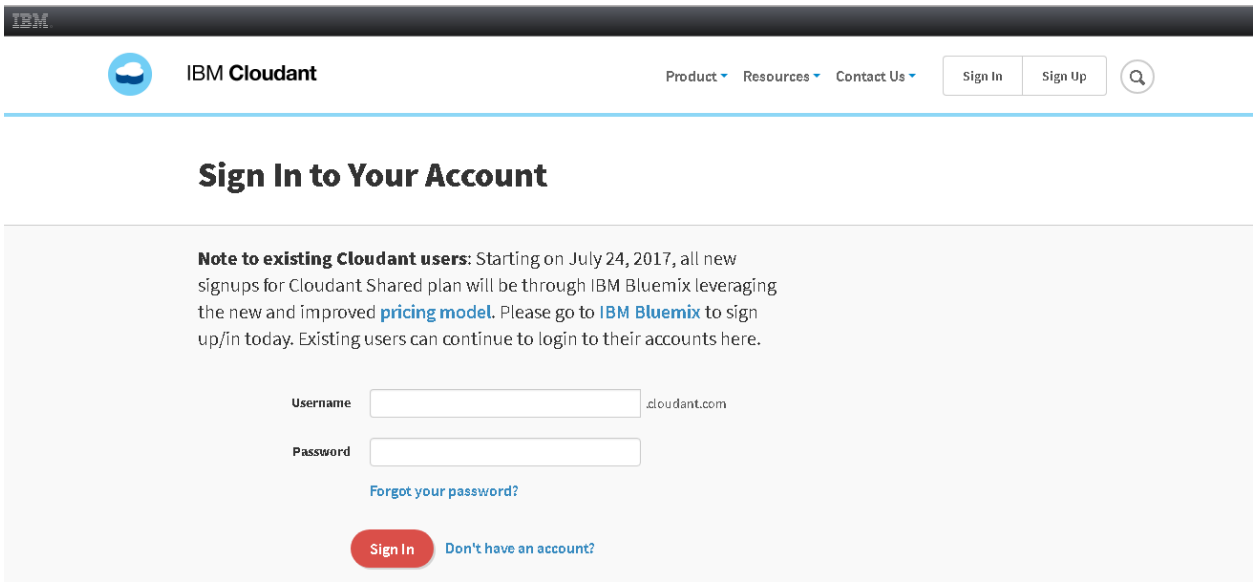
#1 | Creating an IBM Cloudant account

For creating a new Cloudant account visit the below link and click on Sign Up button,
https://cloudant.com/sign-up/

©2017 Miracle Software Systems, Inc.

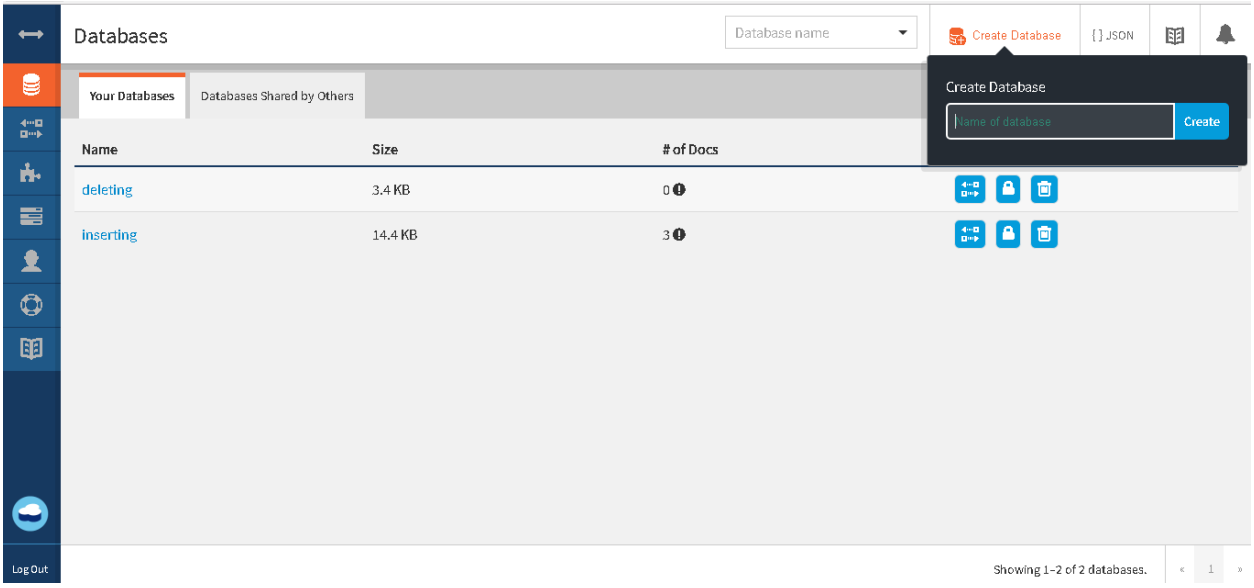


Provide the details and click on Create Account button for new registration to Cloudant. Click on Sign in for logging into the account.



Provide Cloudant credentials and login to your account. After signing in you should to able to see the below dashboard.

©2017 Miracle Software Systems, Inc.



By clicking on create database (you can find Create Database option on top right corner), we can create a new database, by clicking that option, a popup will appear on the screen which asks to enter database name.

Also we can do the same through Node JS code from server side. Below are the steps to perform Cloudant CRUD operations using NodeJS.

#2 | Initializing Cloudant in Node JS

Below are the steps to establish the connection between Cloudant and Node JS

1. Install the Cloudant NPM module in Node JS
2. Import the module in the code
3. Provide your Cloudant credentials (Username and Password) in Node JS

Installing Cloudant Module

Open Command Prompt and give the following command,

npm install cloudant --save

Import Cloudant Module

To import this module in NodeJS, use the below statement in the script file

©2017 Miracle Software Systems, Inc.

```
var Cloudant = require('cloudant');
```

Database Connection

Below is the code snippet for connecting to Cloudant database.

```
var user = <'your Cloudant user name'>;
var pwd= <'your Cloudant password'>;
var cloudant = Cloudant ({account : user, password: pwd});
```

#3 | Creating Cloudant API's for CRUD operations

To create and insert data into Cloudant database using Node.js follow the below steps,

Inserting Data

Copy and paste the code of Create.html in GitHub Repo **crud-app** folder for front end code and for API creation - below is code snippet,

```
app.get('/create', function(req, res) {
var obj = { <JSON Data Object>};
cloudant.db.create('inserting', function() {
var alice = cloudant.db.use('inserting');
alice.insert(obj, function(err) {
   if (err) {
    return console.log('[alice.insert] ', err.message);
   }
   res.send("Inserted Successfully");
  });
 });
});
```

Retrieving Data

Copy and paste the code of Retrieve.html in GitHub Repo **crud-app** folder for front end code and for API creation - below is code snippet

```
app.get('/ret', function(req, res) {
var id = req.query.ss;
var alice = cloudant.db.use('inserting');
```

©2017 Miracle Software Systems, Inc.

```
        var bb = {
                "selector": {
                  _id: id
                }
        };
        alice.find(bb, function(err, result) {
         if (err)
           throw err;
           var obb = result.bookmark;
          if (obb === "nil") {
                res.send("no data found");
         } else {
                 res.send(result)
        }
 }) });
```

Updating Data

Copy and paste the code of Update.html in GitHub Repo **crud-app** folder for front end code and for API creation - below is code snippet

```
        app.get('/update', function(req, res) {
         var id = req.query.ss;
         var alice = cloudant.db.use('inserting');
          var bb = {
            "selector": {
              _id: id
            }
          };
          alice.find(bb, function(err, result) {
            if (err)
              throw err;
            var obb = result.bookmark;
            if (obb === "nil") {
              res.send("no data found");
            } else {
                  res.send(result);
```

©2017 Miracle Software Systems, Inc.

```
        }
      })
});
```

Deleting Data

Copy and paste the code of Delete.html in GitHub Repo **crud-app** folder for front end code and for API creation - below is code snippet

```
app.get('/delete', function(req, res) {
var email1 = req.query.ss;
var alice = cloudant.db.use('inserting');
 var bb = {
  "selector": {
   _id: email1
  }
 };
 alice.find(bb, function(err, result) {
  if (err)
   throw err;
  var obb = result.bookmark;
  if (obb === "nil") {
   res.send("no data found");
  } else {
   alice.destroy(b, a, function(err) {
    if (err) {
     throw err;
    }
    res.send("Deleted Successfully");
   })
  }
 })
});
```

©2017 Miracle Software Systems, Inc.

#4 | Rendering HTML Pages

For rendering HTML pages in Node JS, we need to add the following snippet in our Server code.

```
app.use(express.static(__dirname + '/', {
        index: 'Create.html'
}));
```

#5 | Consuming CRUD API's

In the above portion we created APIs for performing CRUD operations, to make use of those APIs we need to consume API's in front-end using JavaScript.

In this application we created 4 APIs named Create, Update, Retrieve and Delete.

Here is the code snippet for consuming API in JavaScript

```
var xhttp = new XMLHttpRequest();
xhttp.open("POST", "Your Rest URL Here", true);
xhttp.setRequestHeader("Content-type", "application/json");
xhttp.send();
var response = JSON.parse(xhttp.responseText);
```

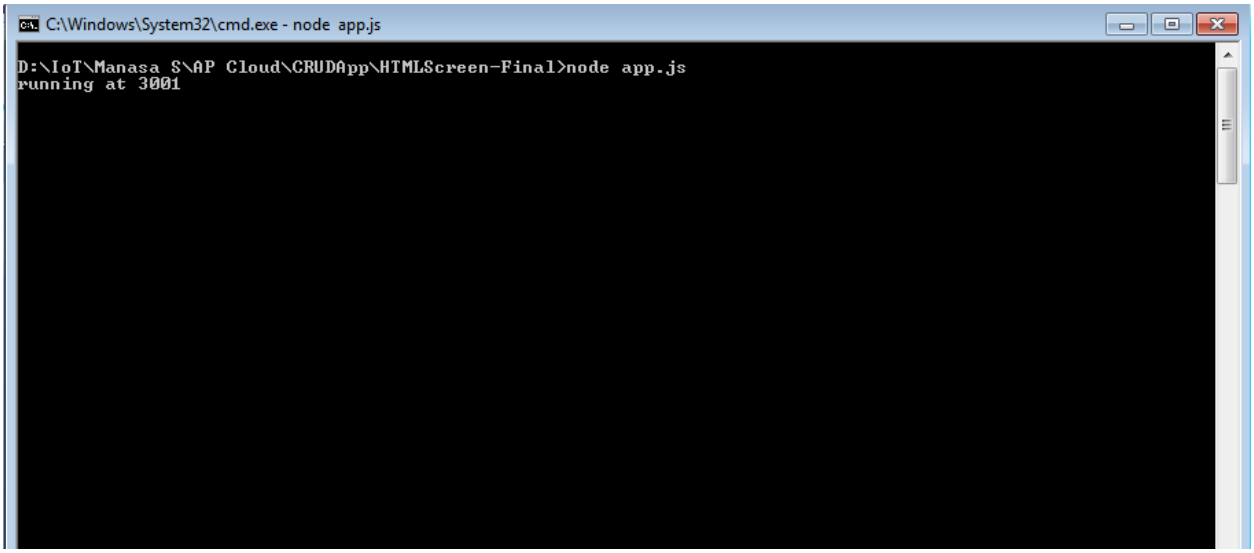
You can find the complete code in GitHub repository

#6 | Run the application

Navigate to the workspace folder where the code exists, and open command prompt

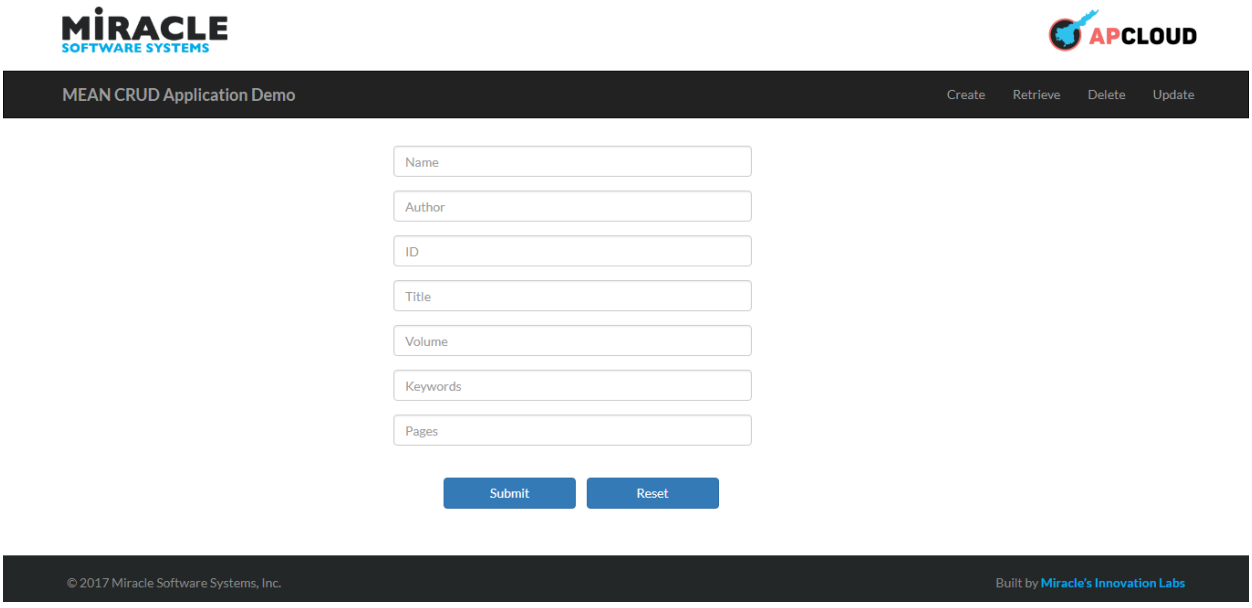
Run **node app.js**

©2017 Miracle Software Systems, Inc.

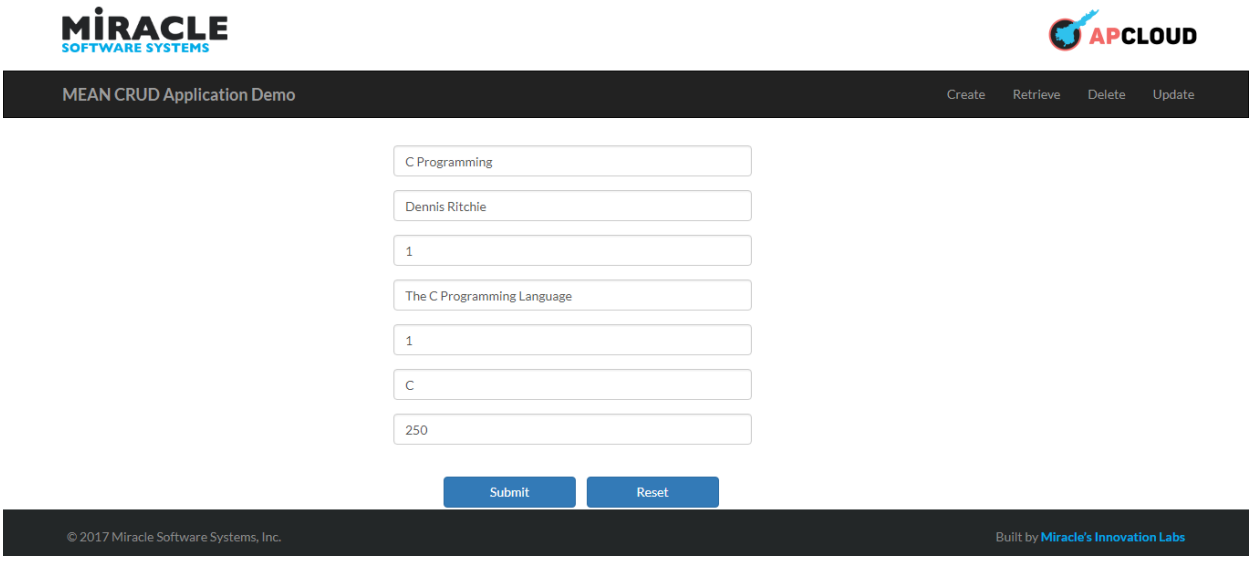


Application is running at http://localhost:3001/

Open in browser, then you can see the below page.

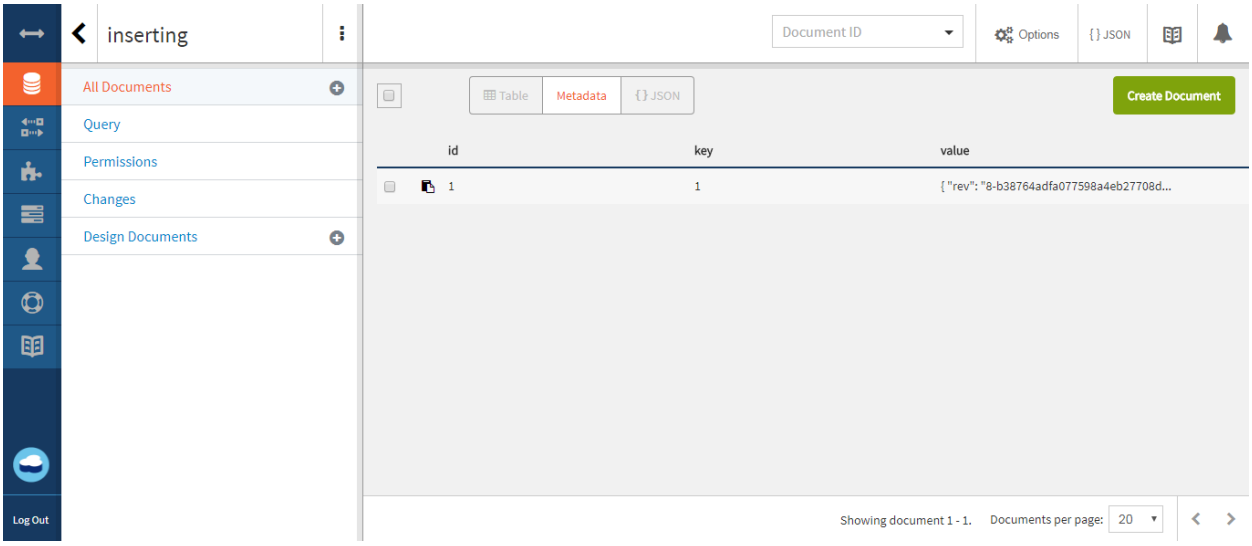


Insert the data as below. After inserting the data will be stored in Cloudant database

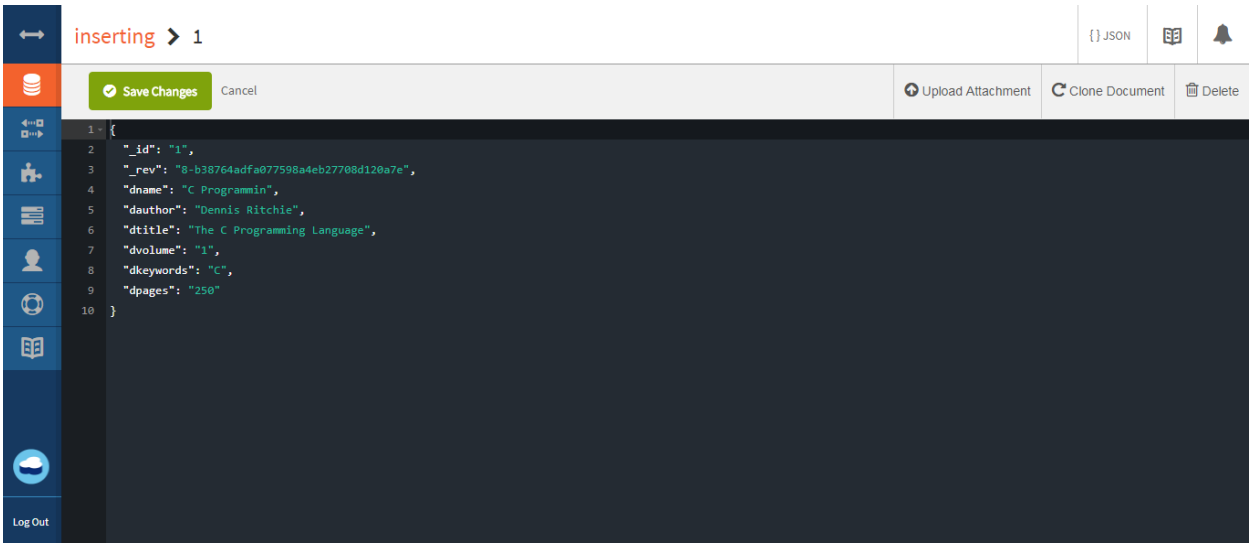


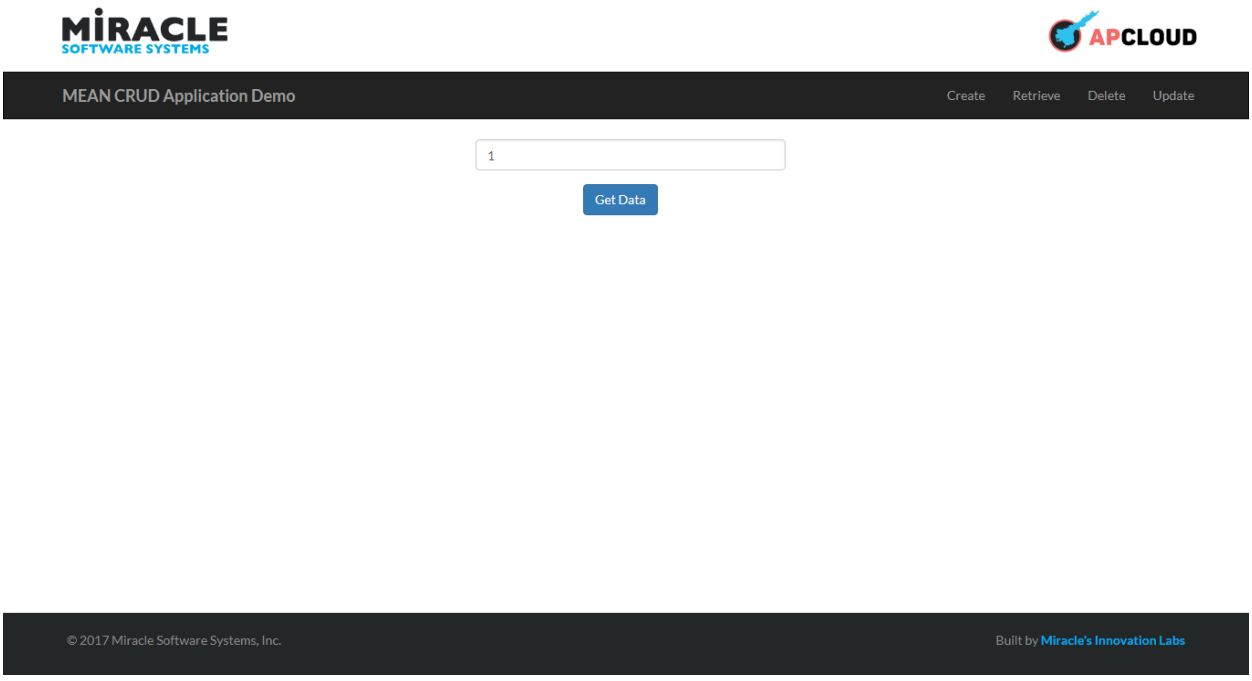
For retrieving the data, click on Retrieve button.

Open IBM Cloudant Dashboard and check whether the data is inserted or not.

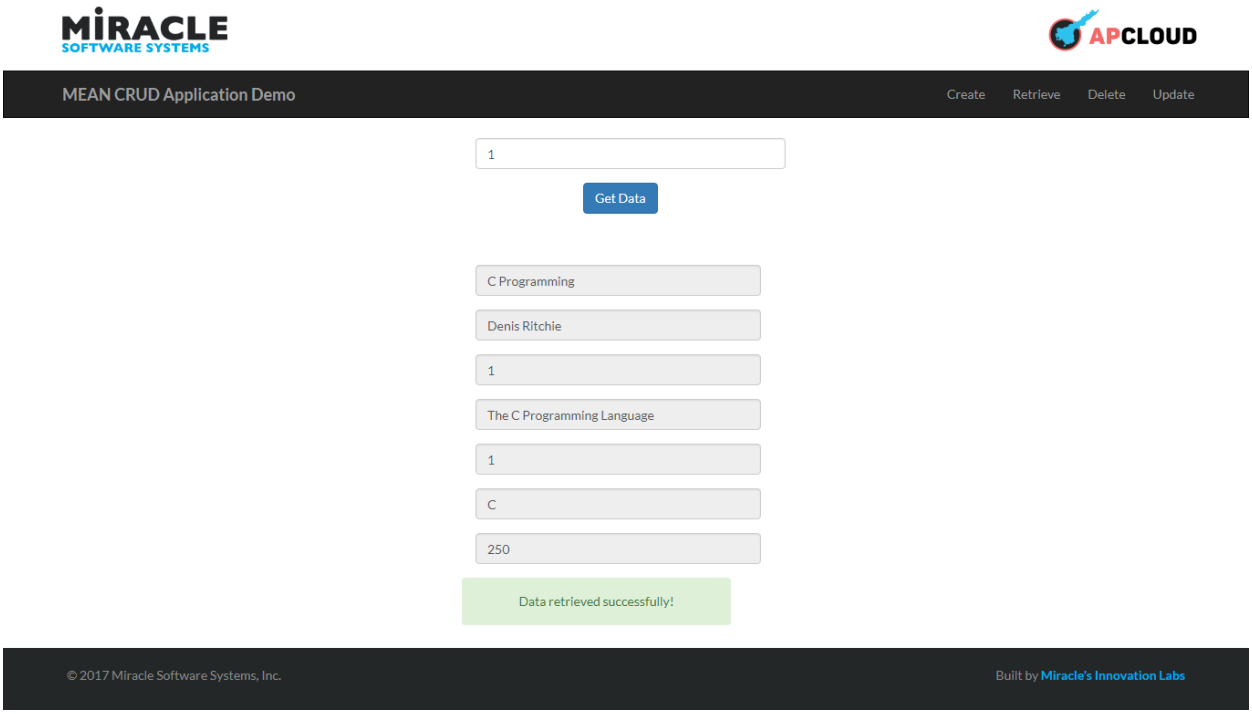


Open the ID and view the data that is available

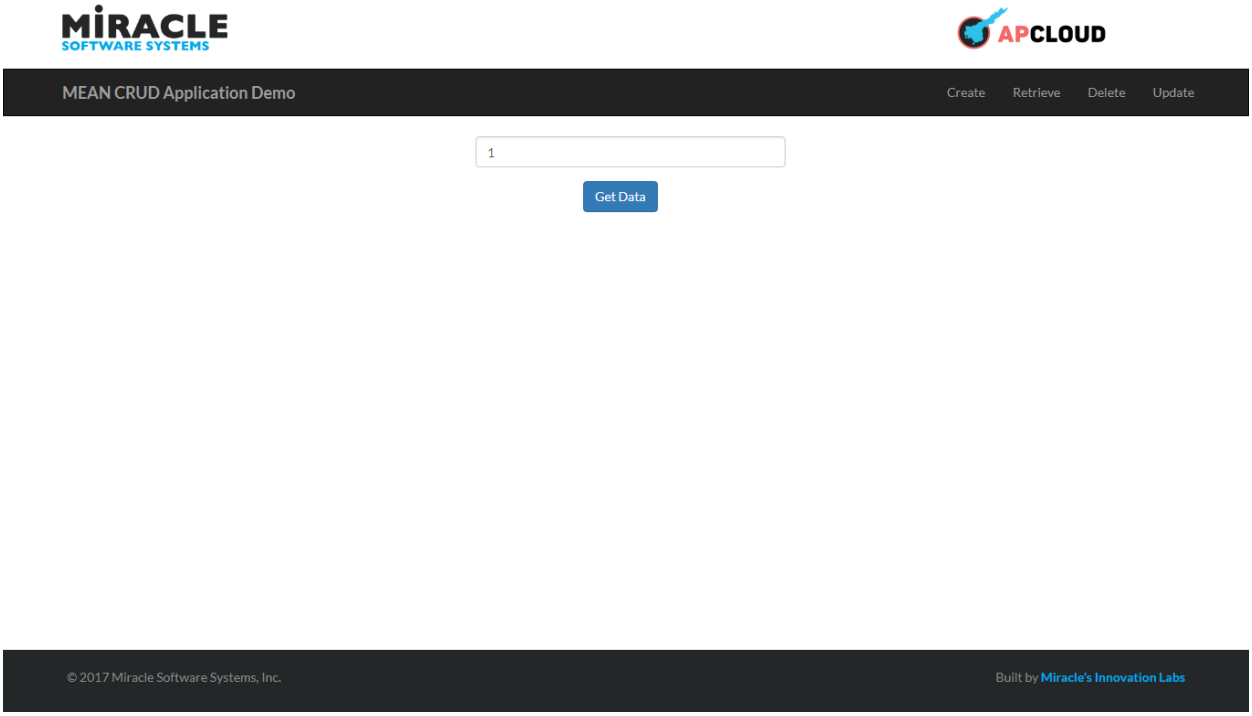




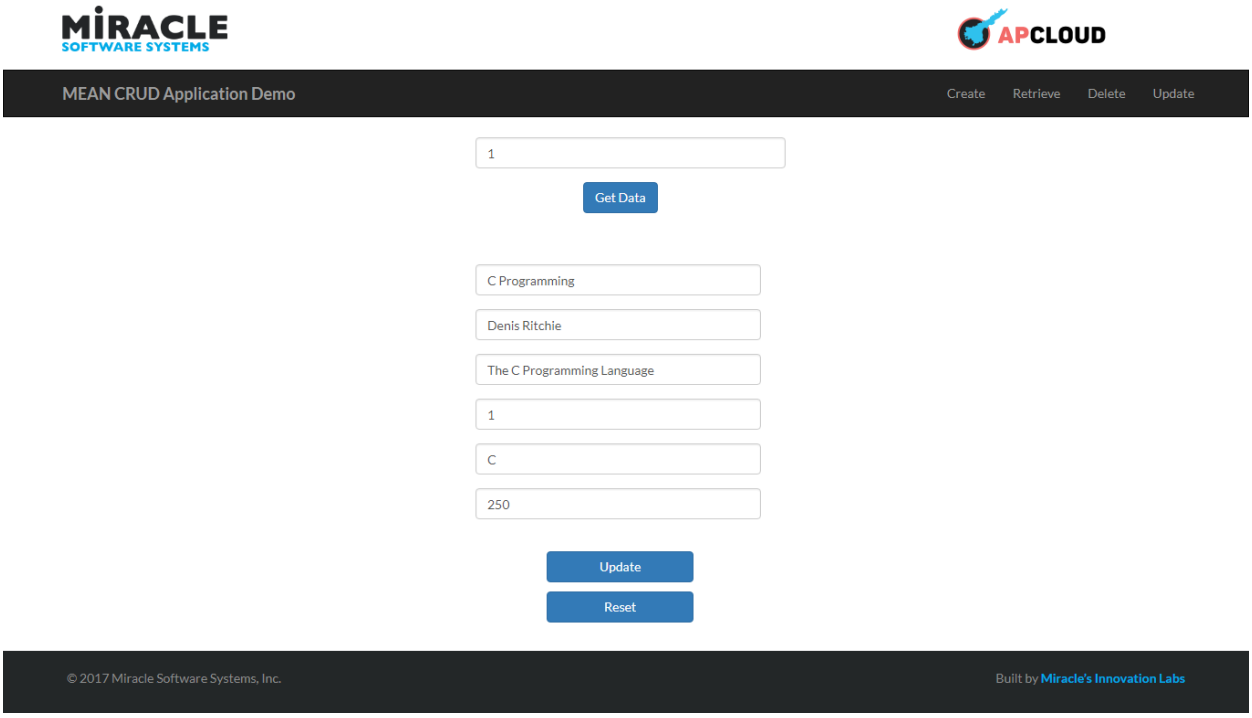
Provide the ID for which you want to retrieve the data. Here we are retrieving ID '1' , the data will be displayed as below,



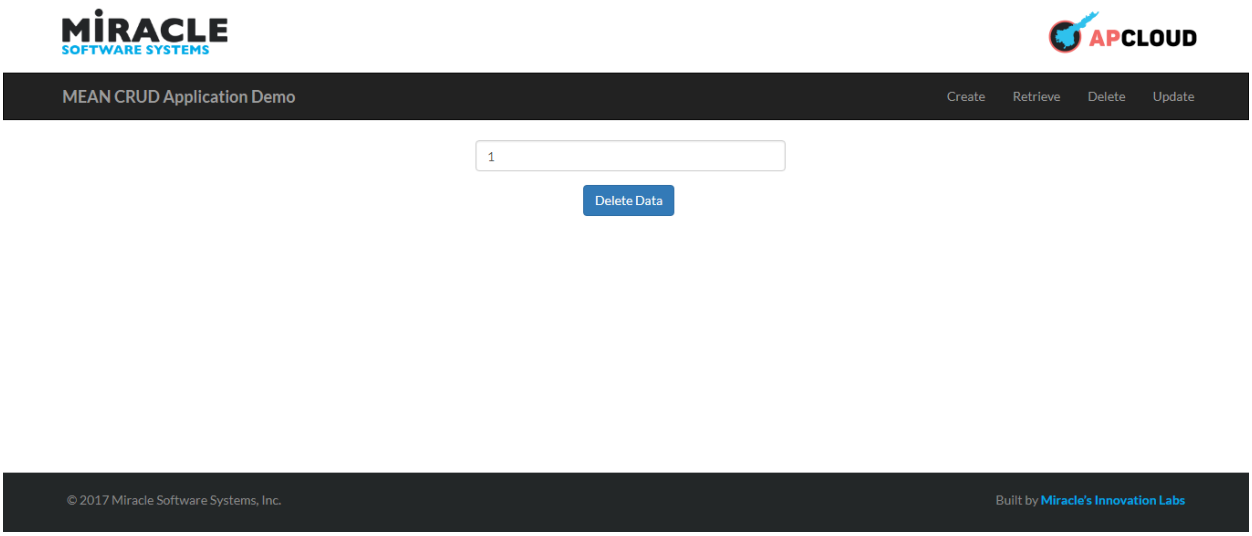
For updating the data, click on Update button.



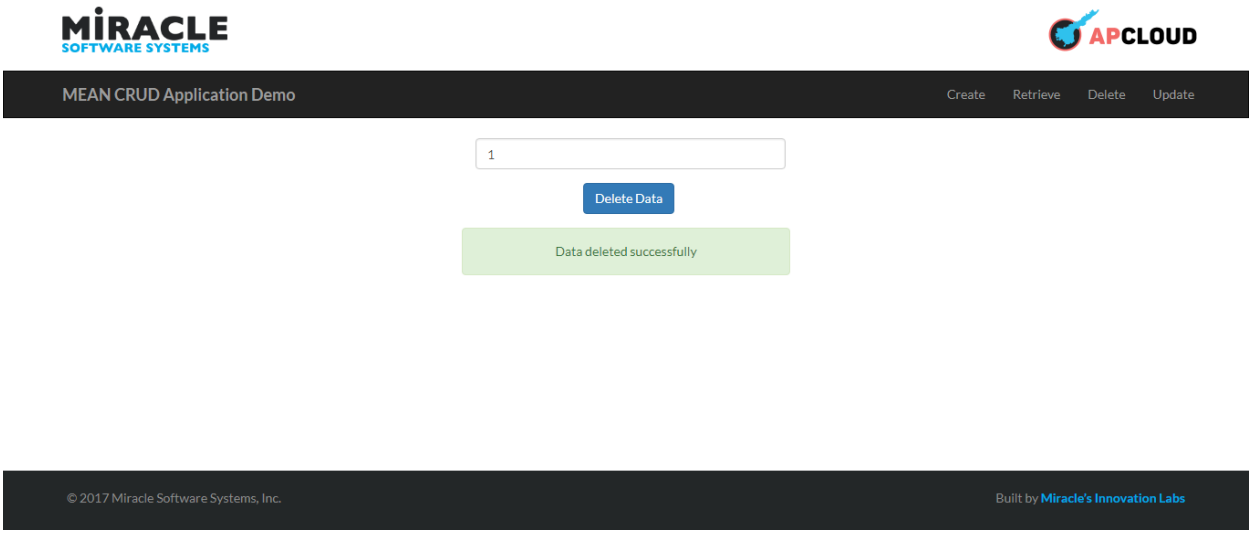
Provide the ID of the data which you want to change the values. After updating the data those updated values will be stored in database successfully.



For deleting the document from database, click on Delete tab.



Provide the ID value which you want to delete.



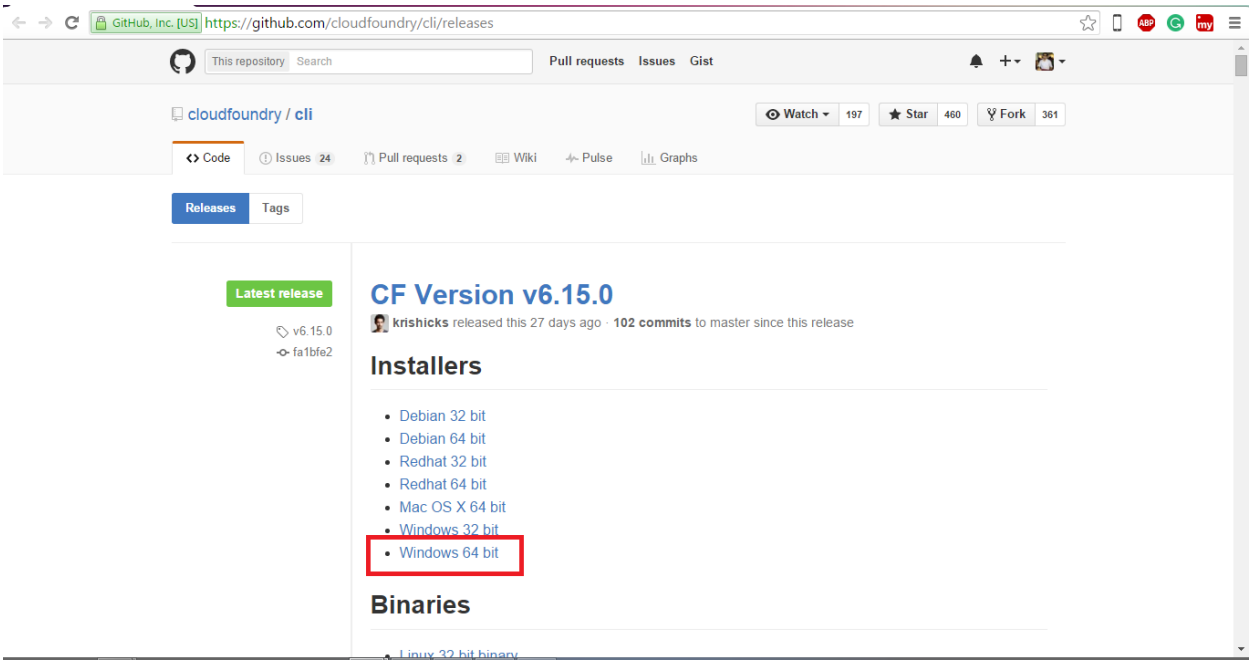
This data will be permanently deleted in the database.

#7 | Installing the CF CLI

At this point you will be asked how you want to start coding your application. For this lab, we will be using the **Cloud Foundry (CF) CLI** option.

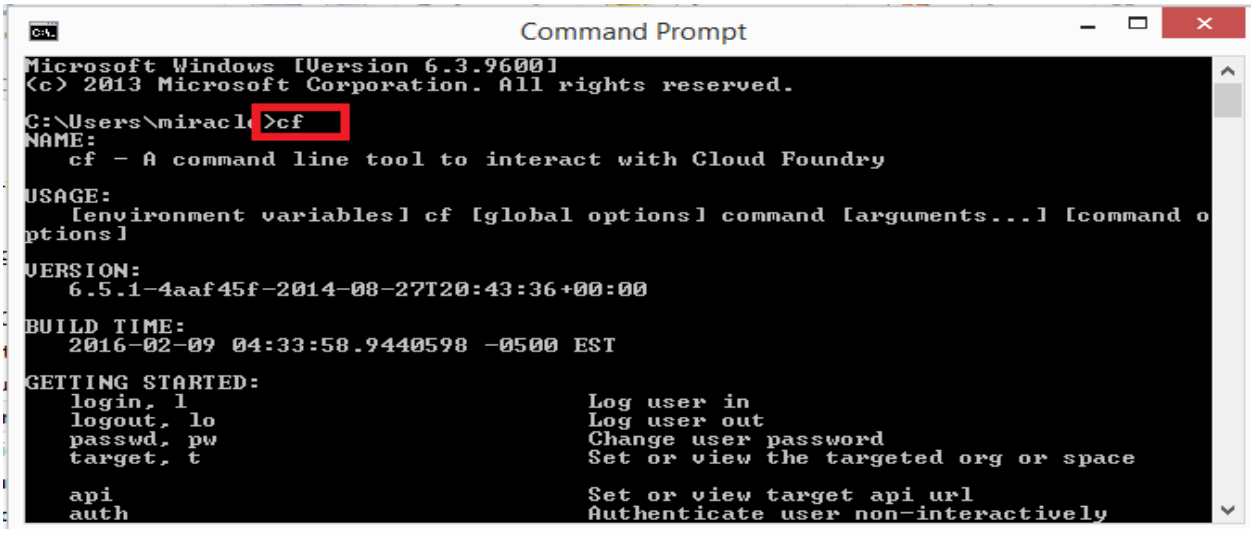
Open this link for installing and downloading,

https://github.com/cloudfoundry/cli



Then you will get a zip file. After extracting the zip file, you can find a **.exe** file inside. Install it.

To check whether **CF** is installed properly or not, open command prompt and execute CF command. Then it will show you a set of **CF** commands, which indicates that CF is successfully installed on your machine.

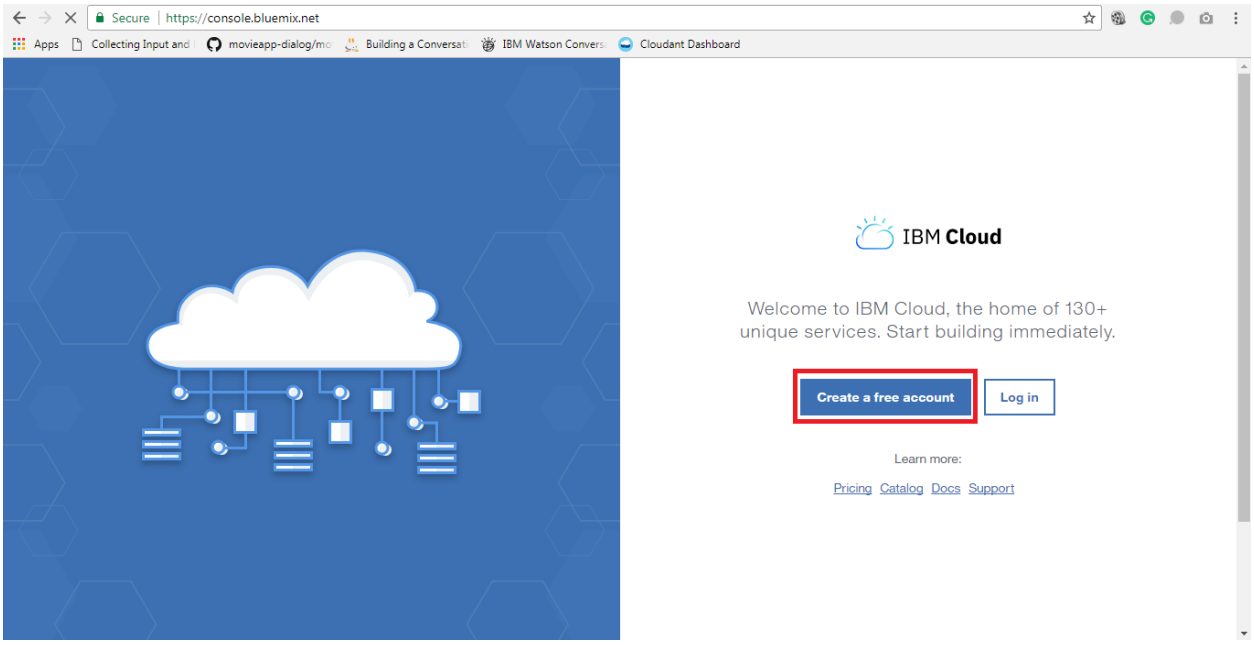


©2017 Miracle Software Systems, Inc.

#8 | Creating an IBM Bluemix account

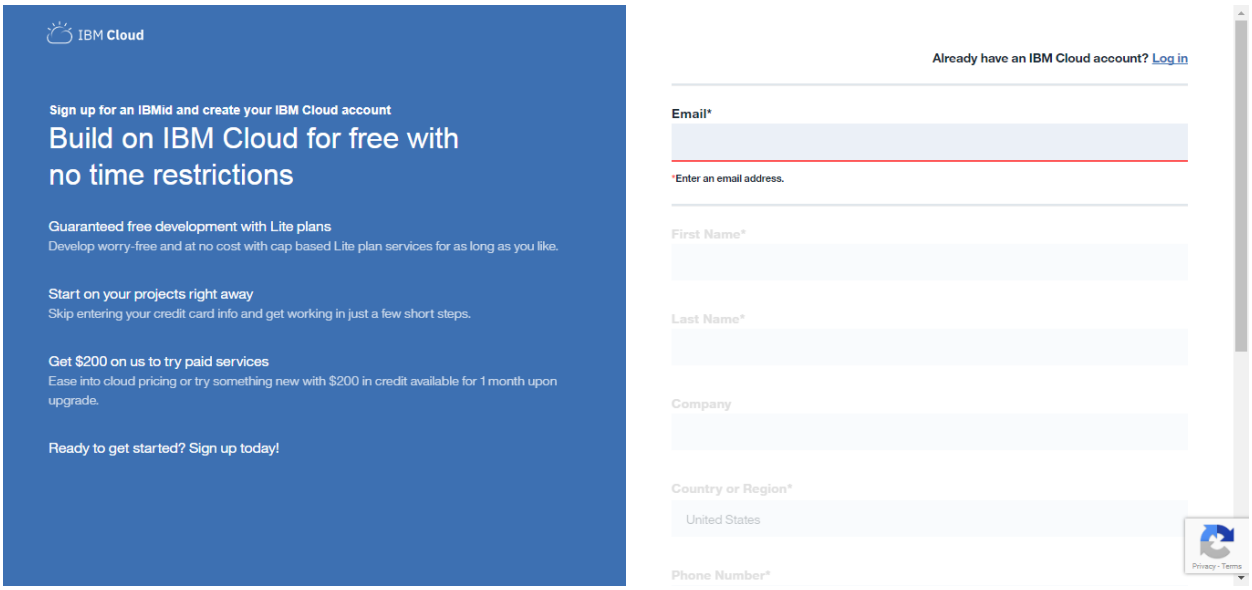
The next step will be to make sure that we have access to the IBM Bluemix Console with either the free trial option (or) the paid subscription option.

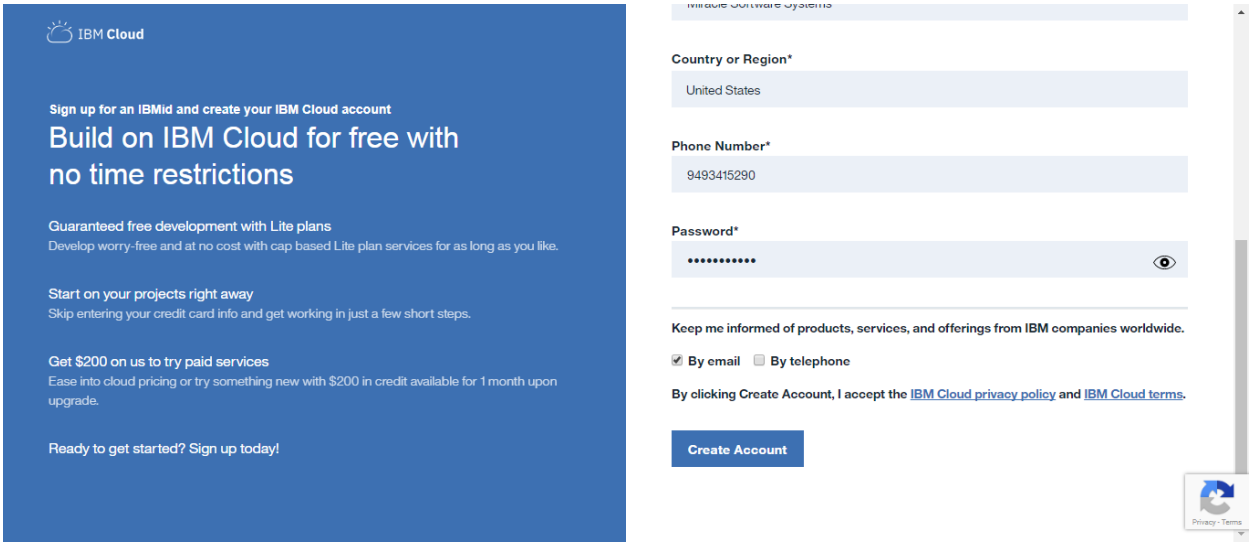
Login to Bluemix at **http://bluemix.net** (or) Register today at **https://console.ng.bluemix.net/registration/**



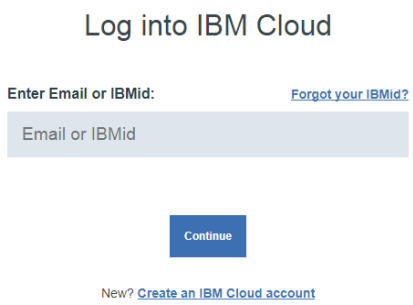
Click on Create a free account, and the fields as required.

©2017 Miracle Software Systems, Inc.

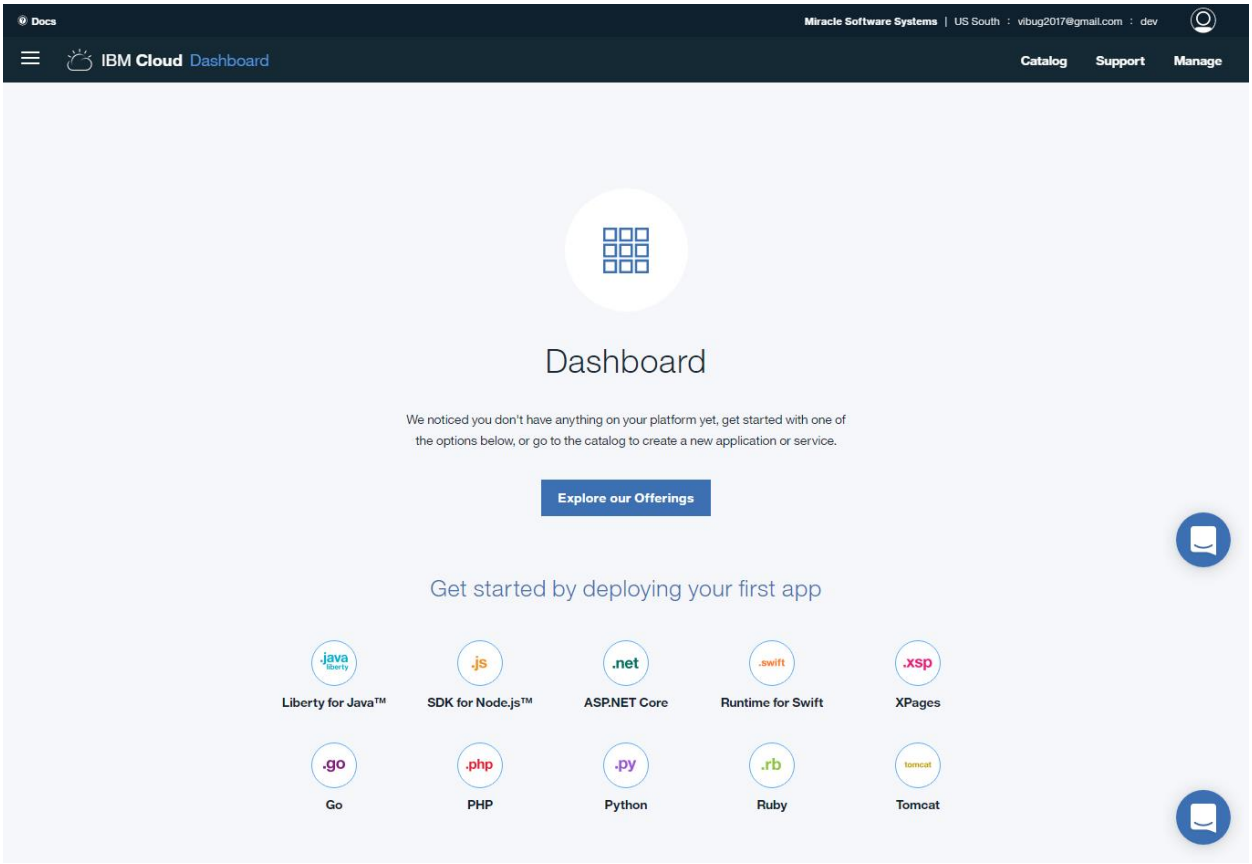




After Clicking on "**Create Account**", confirmation mail will be sent to the registered mail id. Click on Confirm account and then Login to your Bluemix account.



After you login, you can see the dashboard where you can take a look at your applications and services.



©2017 Miracle Software Systems, Inc.

#9 | Deploying the application to IBM Bluemix

The next step will be to take your application and deploy it back to Bluemix so that you can share it with your friends.

Add manifest file, for pushing the application to IBM Bluemix.

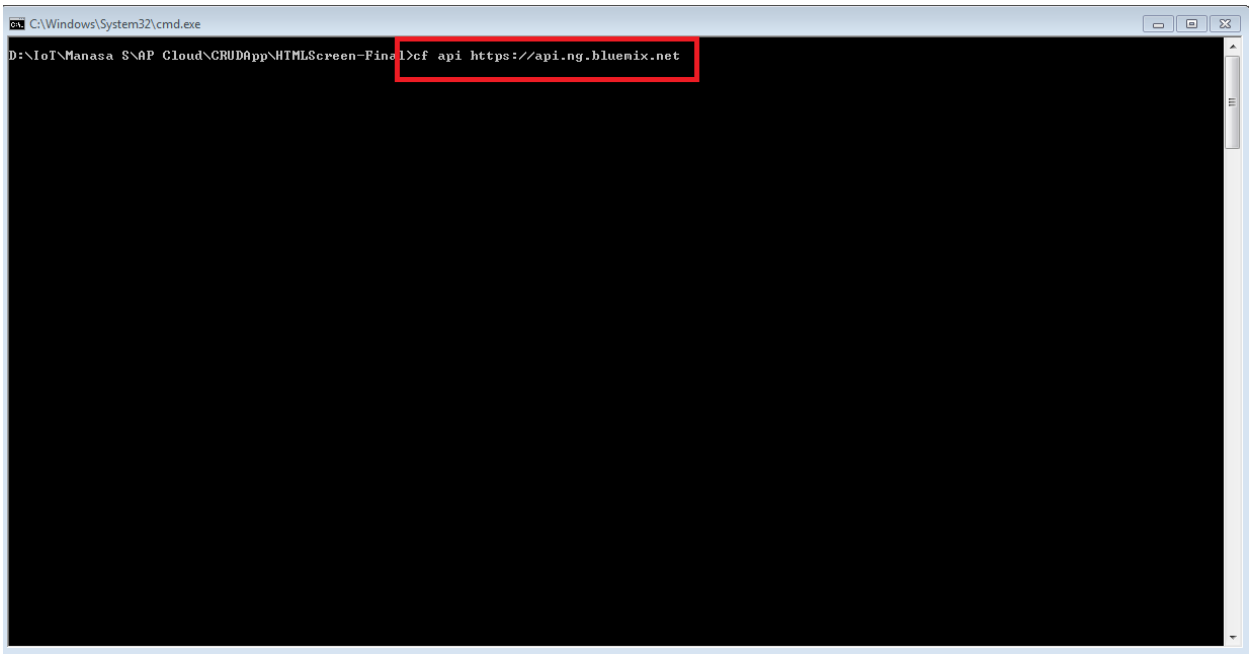
```
applications:
- path: .
  memory: 256M
  instances: 1
  domain: mybluemix.net
  name: crud-node-app
  host: crud-node-app
  disk_quota: 1024M
```

Name the file as manifest.yml and save in the same folder.

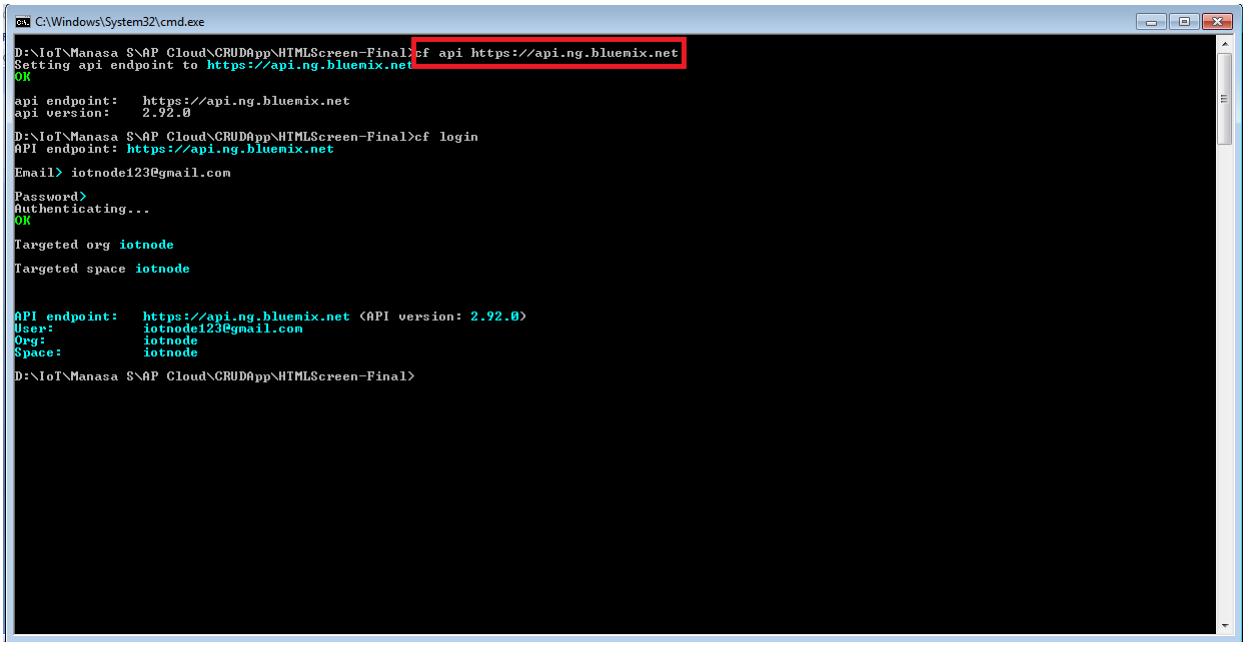
Open the **Command Prompt** and go to the location where you have your workspace. Then, connect to Bluemix using one of the following commands (Depends on which region you selected in your profile).

For Sydney: cf api https://api.au-syd.bluemix.net
For US South: cf api https://api.ng.bluemix.net
For United Kingdom: cf api https://api.eu-gb.bluemix.net

©2017 Miracle Software Systems, Inc.

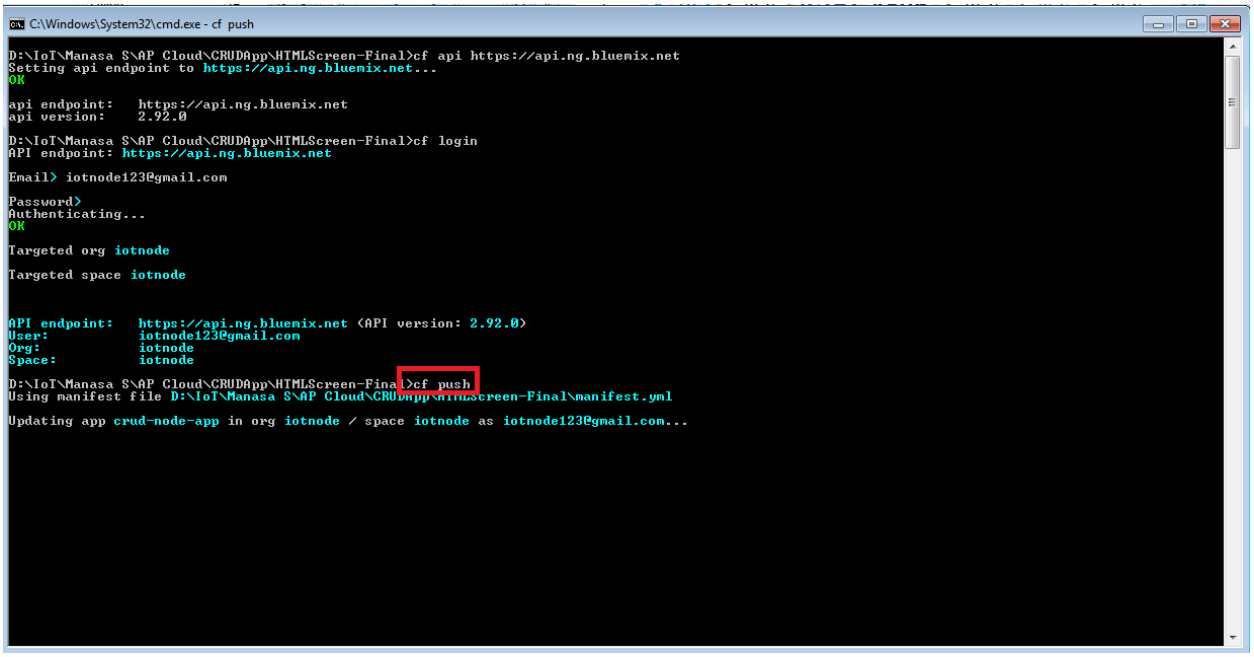


Login to Bluemix using the **“cf login”** command, and when prompted enter your user ID and password to login.



©2017 Miracle Software Systems, Inc.

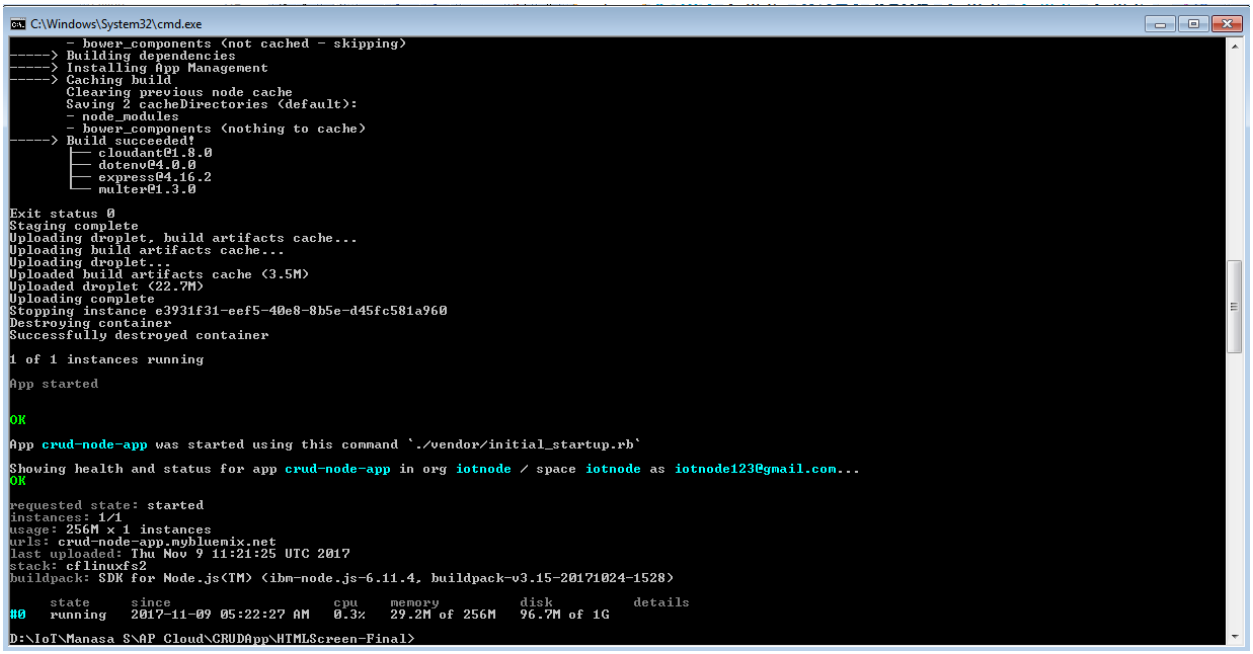
Make sure that you are within your application's directory and use the **"cf push"** command to push your application to your Bluemix Organization.



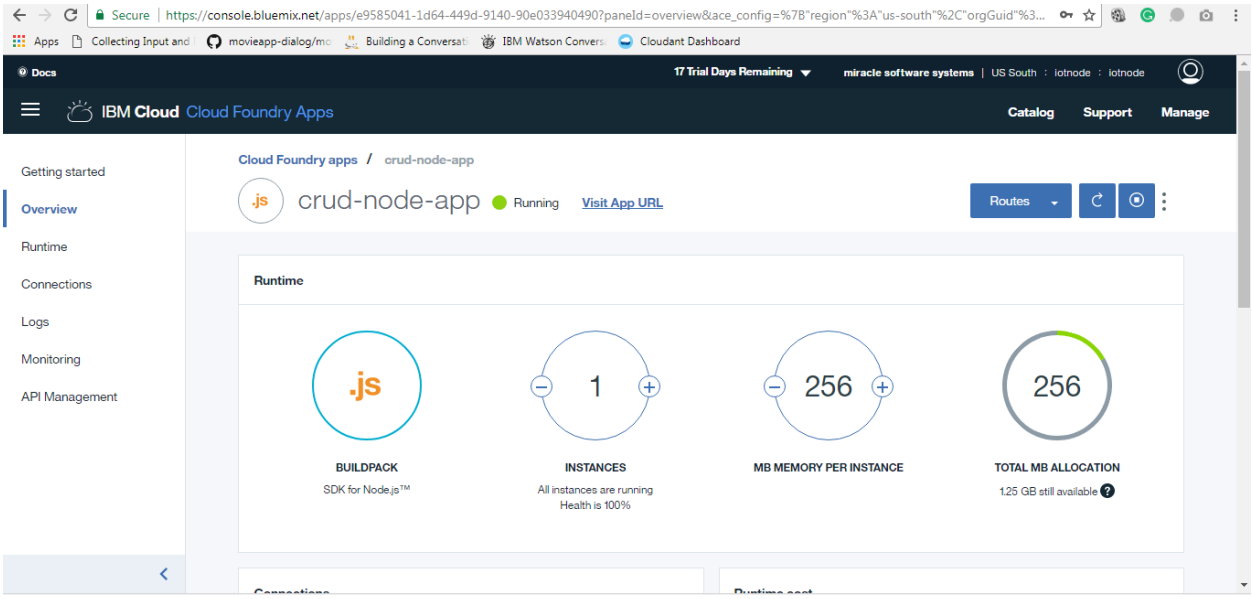
Note: This process might take around 3 to 5 minutes for completion

Once your application is pushed, your Command Prompt should look as shown below,

©2017 Miracle Software Systems, Inc.

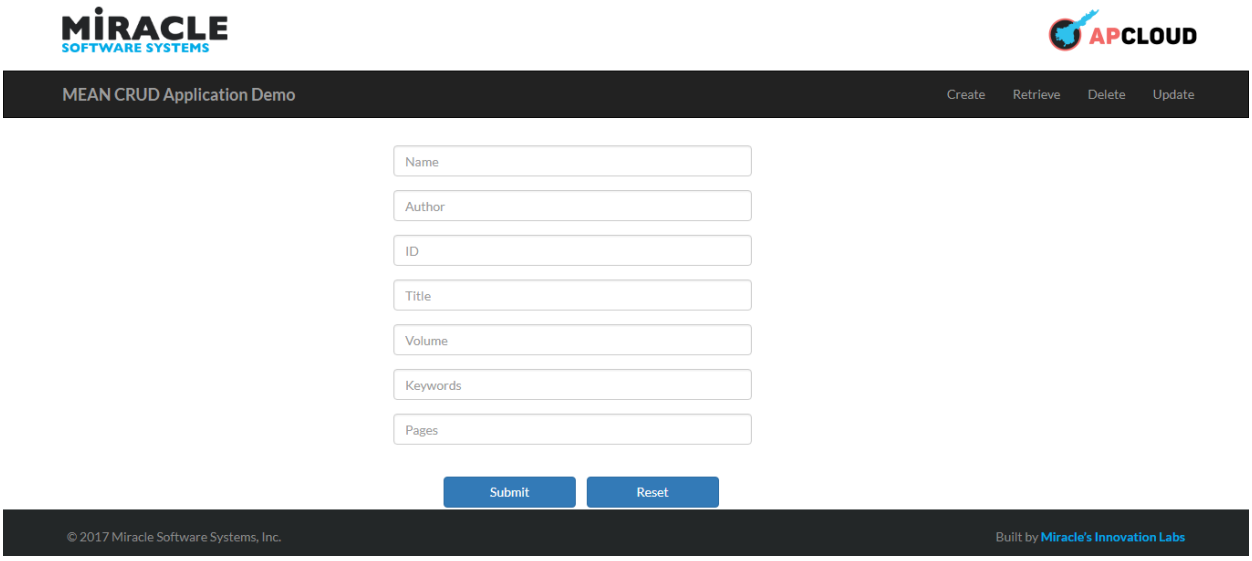


Now, you can go back to your Bluemix account in the browser and access your applications URL through **Dashboard->Application Overview->Application URL**.



'Your very own application, that you created and deployed in IBM Bluemix, should now be available as shown below!

©2017 Miracle Software Systems, Inc.



Now, perform the same operations as above.