



Lab 4: API BaaS - Building social, mobile apps

Overview

In this lab, we will use the Apigee API BaaS and the JavaScript SDK to build a simple HTML5 application. This lab will explore the social aspects of API BaaS such as users, connections, and activities.

Step 0 - Download the Application

If you have a git client installed on your workstation, you can clone the repository located at <https://github.com/chrislatimer/apigee-baas-example.git>. Otherwise, download the application zip file from the location provided by your instructor.

Step 1 - Add the Apigee API BaaS JavaScript SDK

In the demo directory, you will find an index.html file. Open the file in your favorite text editor and locate the following comment:

```
<!-- STEP 1: Insert Apigee JavaScript SDK -->
```

At this location in the file, add a reference to the JavaScript SDK by including the following HTML snippet:

```
<script type="text/javascript" src="js/apigee.js"></script>
```

Gist:

<https://gist.githubusercontent.com/chrislatimer/1483509eec5e18f2c615/raw/da2df09d8e7a97e316604c400419b07fdbaa2745/API-BaaS-Step-1>

Step 2 - Connect to the API BaaS

In the demo directory, you will find a file called baas-demo.js located in the /js directory. Open that file in your favorite text editor and locate the following comment:



```
<!-- STEP 2: Add API BaaS Connection -->  
  <!-- Add step 2 code here -->  
<!-- End STEP 2 -->
```

Add the following snippet of JavaScript.

```
apigeeClient = new Apigee.Client({  
  orgName:'',  
  appName:'',  
  URI: 'https://api.usergrid.com'  
});
```

Gist:

<https://gist.githubusercontent.com/chrislatimer/d86086f1cc021322ed8c/raw/b7c3199c0acae5872376b6189a6fce249b923f18/API-BaaS-Step-2>

Your instructor will provide the appropriate values to use for orgName, appName, and URI.

Step 3 - Initialize API BaaS connection

In the index.html file, locate the following comment:

```
<!-- STEP 3: Initialize API BaaS Connection -->  
  <!-- Add STEP 3 code here -->  
<!-- End STEP 3 -->
```

then replace the comment with the following code:



```
$(document).ready(function(){
    $('following').hover(function(){
        $(this).text("Unfollow");
    },function(){
        $(this).text("Following");
    });
    loadUsers();
});
```

Gist:

<https://gist.githubusercontent.com/chrislatimer/00ec2c69750aaf1277a8/raw/b7fe17a121e31692a0f621087b058e0886d2813f/API-BaaS-Step-3>

Step 4 - Add loadUsers() JavaScript method

In the js/baas-demo.js file, locate the following comment:

```
<!-- STEP 4: Add loadUsers() function -->
<!-- Add step 4 code here -->
<!-- End STEP 4 -->
```

then replace the comment with the following code:

```
my_users = new Apigee.Collection({ "client":apigeeClient, "type":"users" });
function loadUsers (){
    my_users.fetch(function () {
        while (my_users.hasNextEntity() ) {
            var current_user = my_users.getNextEntity();
            if (current_user.get('username') != 'arthur'){
                $('#contentdiv').append("<div class='item'> <a href='#'> <img class='avatar'
src='img/smallred.png'/>"
                + "<strong class='name'>" +current_user.get('username') + "</strong>"
                + " <span class='username '>@"+current_user.get('username') + "</span>"
                + "</a><span class='button follow' id='"+ current_user.get('username') +'>Follow
</span> </div>");
            }
        }
    },
    function () {
        alert("Failed to read Users");
    }
}
```



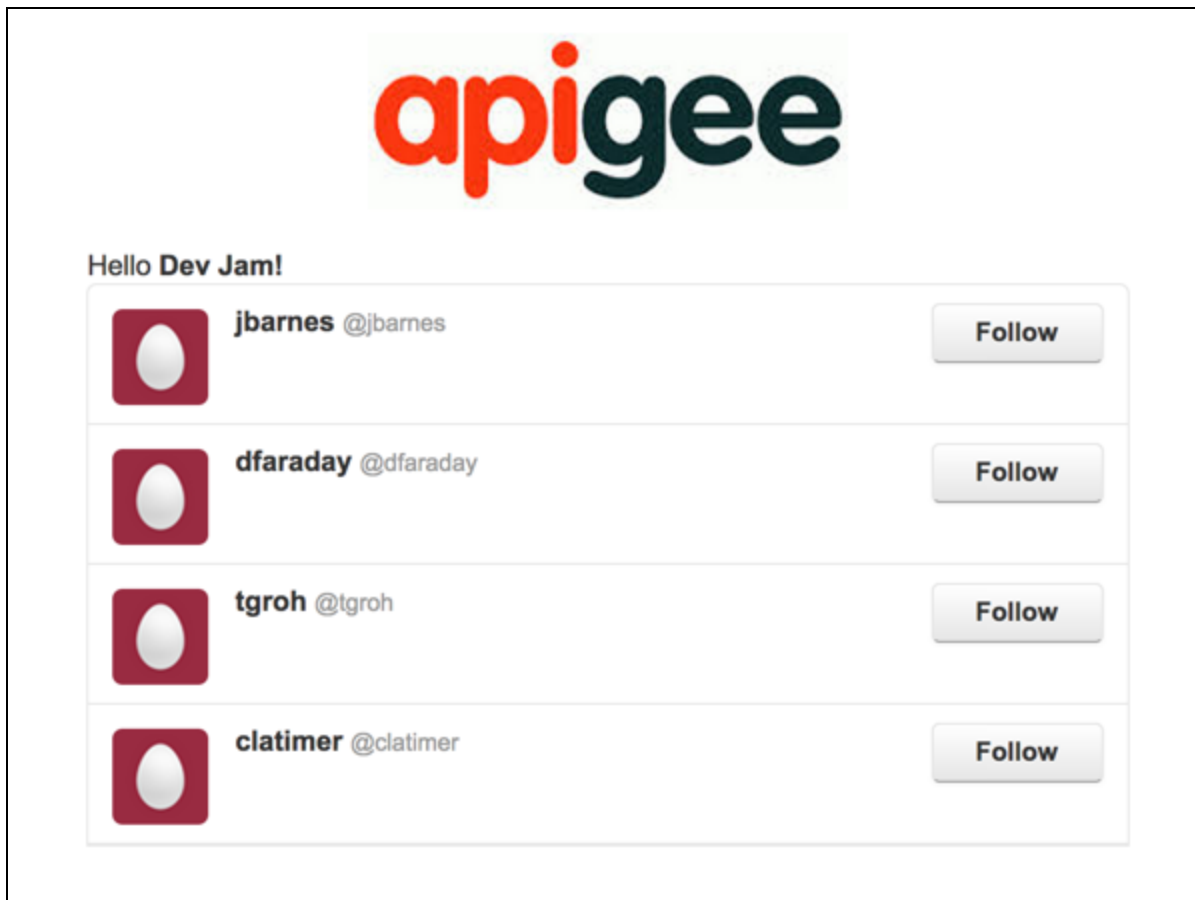
```
};  
}
```

Gist:

<https://gist.githubusercontent.com/chrislatimer/df217c3cb7bb2b816a56/raw/385101e4effc9f883cda65d5f17c762bc62262cb/API-BaaS-Step-4>

Step 5 - Open index.html

In your favorite browser, open index.html and verify that you see a list of users displayed. Your index should look similar to the following image, although the list of users may be different:



Step 6 - Add EventListener on Follow Buttons



In the js/baas-demo.js file, locate the following comment:

```
<!-- STEP 6: Add EventListeners -->
  <!-- Add step 6 code here -->
<!-- End STEP 6 -->
```

and replace it with the following JavaScript snippet:

```
$(".follow, .following").live('click', function() {
  $(this).toggleClass('following follow').unbind("hover");
  if($(this).is('.follow')){
    var element = $(this);
    var userID = element.attr("id");
    createRelationship ('arthur',userID,'unfollows');
    createActivityStream("arthur",userID,"unfollows");
    $(this).text("Follow");
  }
  else{
    var element = $(this);
    var userID = element.attr("id");
    createRelationship ('arthur',userID,'follows');
    createActivityStream("arthur",userID,"follows");
    $(this).bind({mouseleave:function(){ $(this).text("Following");}, mouseenter:function(){
      $(this).text("Unfollow"); }
    });
  }
});
```

Gist:

<https://gist.githubusercontent.com/chrislatimer/e94bd1bb1e76bc0f1185/raw/96db97df578b157f3f6b43435a7ced7a7c9b6f84/API-BaaS-Step-6>

Step 7 - Add createRelationship JavaScript function

In the js/baas-demo.js file, locate the following comment:



```
<!-- STEP 7: Add createRelationship() function -->  
  <!-- Add createRelationship code here -->  
<!-- End STEP 7 -->
```

and replace it with the following JavaScript snippet:



```
function createRelationship (follower, leader, relationship){
  var connecting_entity_options = {
    client: apigeeClient, data: {
      type:'users', username: follower
    }
  };
  var connecting_entity = new Apigee.Entity(connecting_entity_options);
  var connected_entity_options = {
    client: apigeeClient,
    data: {
      type:'users', username:leader
    }
  };
  if (relationship == 'follows'){
    connecting_entity.connect(relationship, connected_entity,function (error, result) {
      if (error) {
        alert ("Error occurred while creating the link between Arthur & "+leader);
      }
      else {
        console.log ("Successfully created the link between Arthur & "+leader);
      }
    });
  }
  else {
    connecting_entity.disconnect('follows', connected_entity, function (error, result) {
      if (error) {
        alert ("Error occurred while removing the link between Arthur & "+leader);
      }
      else {
        console.log ("Successfully created the link between Arthur & "+leader);
      }
    });
  }
}
```

Gist:

<https://gist.githubusercontent.com/chrislatimer/d0dc2176626eab59e913/raw/dca1c9396d3769c701ded535f7d5f90eb137939f/API-BaaS-Step-7>

Step 8 - Add createActivityStream JavaScript function

In the js/baas-demo.js file, locate the following comment:

```
<!-- STEP 8: Add createActivitySteam() function -->
```



```
<!-- Add createActivityStream code here -->
<!-- End STEP 8 -->
```

and replace it with the following JavaScript snippet:

```
function createActivityStream(follower,leader,activity){
  var options = {
    method:'POST',
    endpoint:'users/'+follower+'/activities',
    body: {
      "actor":{
        "displayName":"Arthur",
        "username":follower,
        "content": follower + " " + activity + " " + leader
      },
      "verb":"post"
    }
  };
  apigeeClient.request(options, function (err, data) {
    if (err) {
      console.log('Write failed!');
      alert("An error occurred while creating the activity stream in Apigee");
    } else {
      console.log('Write success!');
    }
  });
}
```

Gist:

<https://gist.githubusercontent.com/chrislatimer/caa386bc90b5f3f5ca60/raw/bade5ad651903250f3d3e79bd8f0d5eea80801b0/API-BaaS-Step-8>

Step 9 - Refresh page in browser

In your browser refresh the page. You should now be able to follow/unfollow users by clicking on the follow/unfollow button.

Please talk to your instructor about how to check if the relationships are actually being created between the different users.

References



-
- Create an account with Apigee Edge
https://accounts.apigee.com/accounts/sign_up
 - API BaaS SDKs
<http://apigee.com/docs/app-services/content/app-services-sdks-0>
 - Creating users in app services
<http://apigee.com/docs/app-services/content/user>
 - Posting an activity into the data store
<http://apigee.com/docs/api-baas/content/creating-activity>
 - Connecting entities
<http://apigee.com/docs/app-services/content/connecting-users-other-data>