

Introduction

This goal of this workshop is to allow developers to gain some general familiarity with the 3Scale API Management System. During the session, participants will touch on the many functional areas that comprise 3Scale including the creation of an API Service definition, Application Plans, Metrics, Developer Portal and more.

*This is not intended to be a training course.

This workshop assumes the participants are using the 3Scale On Line Free Trial. Use the link below and follow the instructions provided.

Free Trial Online: <http://www.3scale.net/signup>

If the workshop is using an on premise installation, your instructor will provide you with additional information regarding links, user id's and passwords.

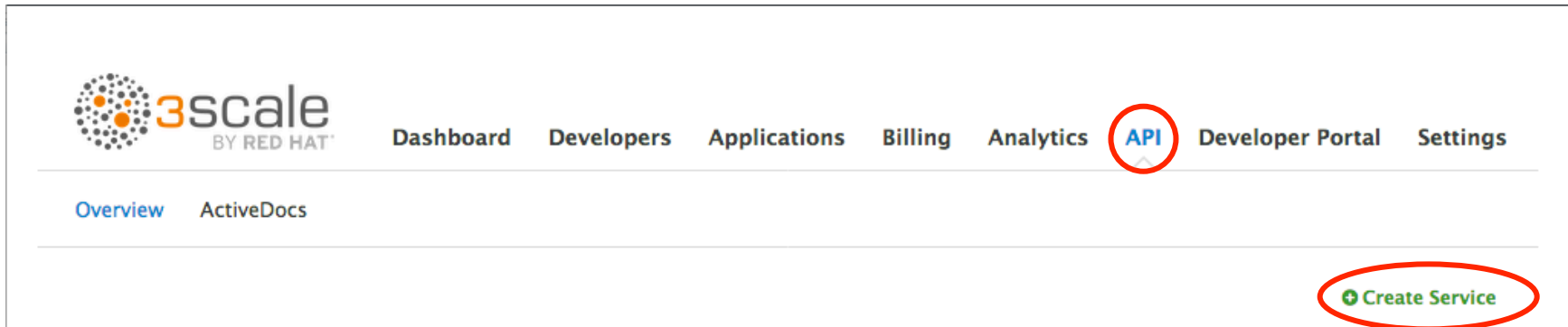
During the workshop you will:

1. Define and configure a new API Service and Application
2. Create an API user
3. Create an Application Plan with policies
4. Create Mapping Rules
5. Import a SWAGGER doc
6. Modify the Developer Portal

Have Fun!

Create a new API Service

From the main screen click the 'API' link and then the 'Create Service' link.





[Dashboard](#) [Developers](#) [Applications](#) [Billing](#) [Analytics](#) [API](#) [Developer Portal](#) [Settings](#)

[Overview](#) [ActiveDocs](#)

Create New Service

Name*

USC Service API

System name*

usc_service

Only ASCII letters, numbers, dashes and underscores

Description

Fill out the form provided with the following values:

Name* ***USC Service API***

System name* ***usc_service***

Click the 'Create' button at the bottom of the screen.

Create Application Plan(s)

▼ JSC Service API

Definition, Integration and Settings

On the Integration page, add your *API base URL* to the *Private Base URL* field in the staging box and hit **Update & Test**. Once the staging box is green, hit **Deploy** in the production box and you have completed a basic integration.

[Configure APIcast](#)

Authenticated by **API key**

ID for API calls is 4 and system name is **usc_service**

Users **can** manage application keys

Users **can** manage applications

Users **can request plan change**

Users **cannot select a plan** when creating an application

Published Application Plans ?

[+ Create Application Plan](#)

There are no published application plans. Create one!

You have 0 application plans (0 published) with a total of 0 live applications.

Analytics

Hits
0 hits

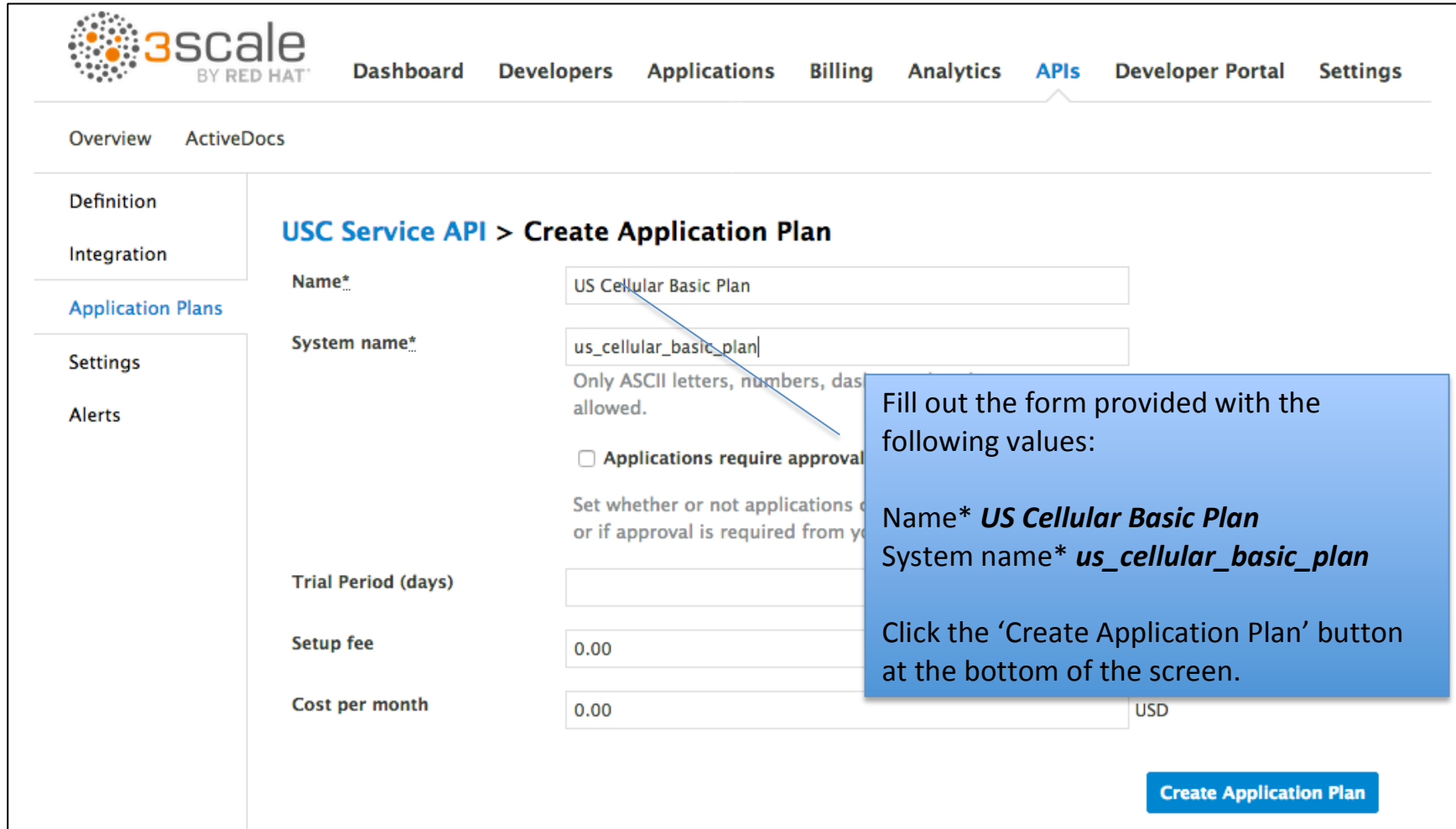
Latest alerts

There are no alerts.

Latest Apps

There are no latest applications.

Create the Basic Plan



3scale
BY RED HAT™

Dashboard Developers Applications Billing Analytics **APIs** Developer Portal Settings

Overview ActiveDocs

Definition
Integration
Application Plans
Settings
Alerts

USC Service API > Create Application Plan

Name*

System name*

Only ASCII letters, numbers, dashes, and underscores are allowed.

☐ Applications require approval

Set whether or not applications require approval or if approval is required from your account.

Trial Period (days)

Setup fee

Cost per month USD

Create Application Plan

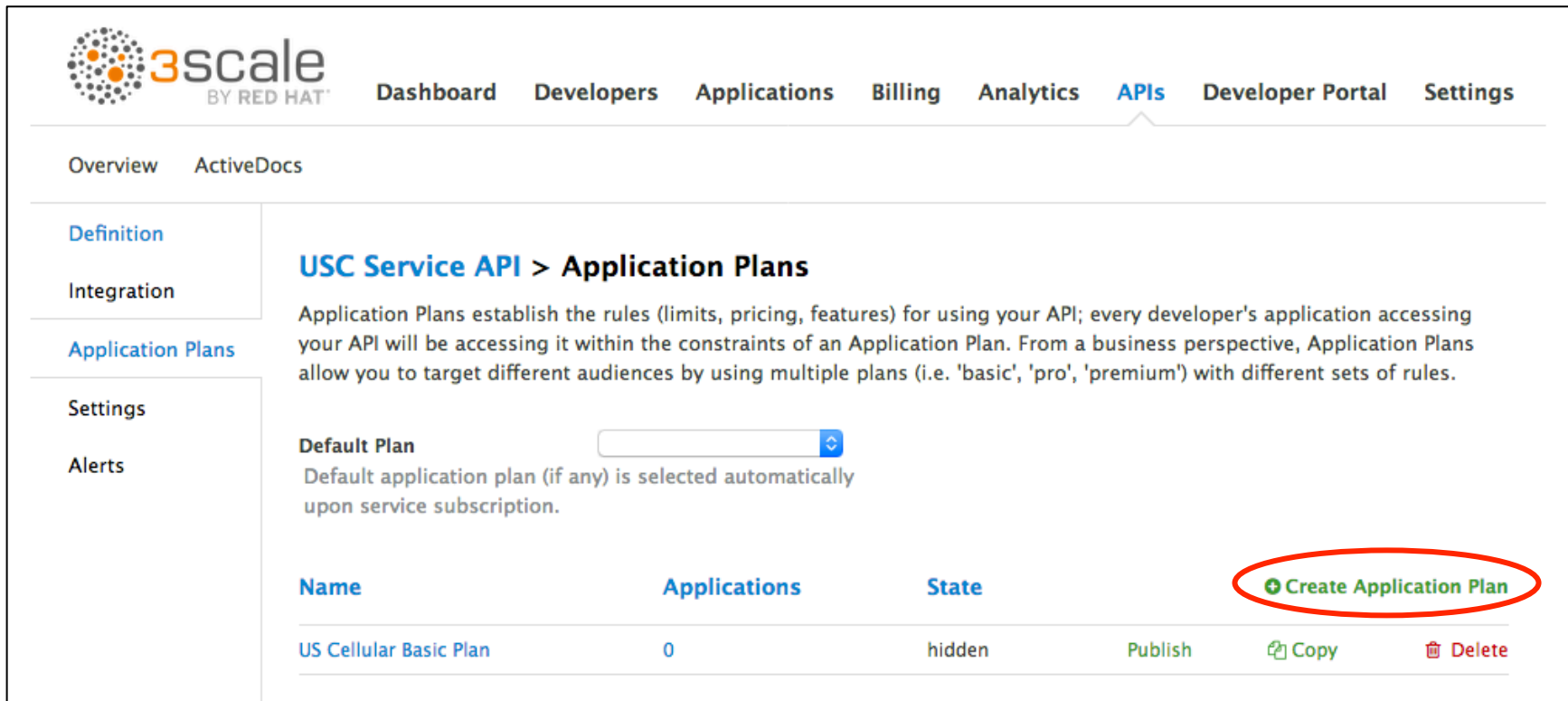
Fill out the form provided with the following values:

Name* **US Cellular Basic Plan**

System name* **us_cellular_basic_plan**

Click the 'Create Application Plan' button at the bottom of the screen.

Create the Premium Plan



The screenshot shows the 3scale API management console. The top navigation bar includes links for Dashboard, Developers, Applications, Billing, Analytics, APIs (selected), Developer Portal, and Settings. The left sidebar has links for Overview, ActiveDocs, Definition, Integration, Application Plans (selected), Settings, and Alerts. The main content area is titled 'USC Service API > Application Plans'. It contains a paragraph explaining Application Plans and a 'Default Plan' dropdown menu. Below this is a table with columns: Name, Applications, State, and actions. The table lists one plan: 'US Cellular Basic Plan' with 0 applications and a hidden state. A red circle highlights the '+ Create Application Plan' button in the top right corner of the table.

USC Service API > Application Plans

Application Plans establish the rules (limits, pricing, features) for using your API; every developer's application accessing your API will be accessing it within the constraints of an Application Plan. From a business perspective, Application Plans allow you to target different audiences by using multiple plans (i.e. 'basic', 'pro', 'premium') with different sets of rules.

Default Plan

Default application plan (if any) is selected automatically upon service subscription.

Name	Applications	State	
US Cellular Basic Plan	0	hidden	Publish Copy Delete

[+ Create Application Plan](#)

Click the link shown above to add another application plan. Use the values as shown below:

Name* **US Cellular Premium Plan**

System Name* **us_cellular_premium_plan**

Choose Default and Publish

3scale
BY RED HAT

Dashboard Developers Applications Billing Analytics **APIs** Developer Portal Settings

Overview ActiveDocs

Definition
Integration
Application Plans
Settings
Alerts

USC Service API > Application Plans

Application Plans establish the rules (limits, pricing, features) for using your API; every developer's application accessing your API will be accessing it within the constraints of an Application Plan. From a business perspective, Application Plans allow you to target different audiences by using multiple plans (i.e. 'basic', 'pro', 'premium') with different sets of rules.

Default Plan
Default application plan (if any) is selected automatically upon service subscription.

US Cellular Basic Plan

Name	Applications	State			
US Cellular Basic Plan	0	hidden	Publish	Copy	Delete
US Cellular Premium Plan	0	hidden	Publish	Copy	Delete

+ Create Application Plan

1. Select '**US Cellular Basic Plan**' as your Default Plan
2. Click the '**Publish**' link for each plan
3. Click the '**Definition**' link to view Methods and Metrics.

Methods and Metrics

Methods

Add the methods of this API to get data on their individual usage. Method calls trigger the built-in Hits-metric. Usage limits and pricing rules for individual methods are defined from within each [Application Plan](#). A method needs to be mapped to one or more URL patterns in the [Mapping Rules](#) section of the integration page so specific calls to your API up the count of specific methods.

Method	System Name	Unit	Description	Mapped	+ New method
You have no methods					

USC Service API > New Method

Friendly name*

get-employees

e.g. Create new user

System name*

get-employees

e.g. users/create or create_user. Spaces are

Description

Get a list of all employees

Fill out the form provided with the following values:

Name* **Get All Employees**

System name* **getEmployees**

Click the 'Create Method' button at the bottom of the screen.

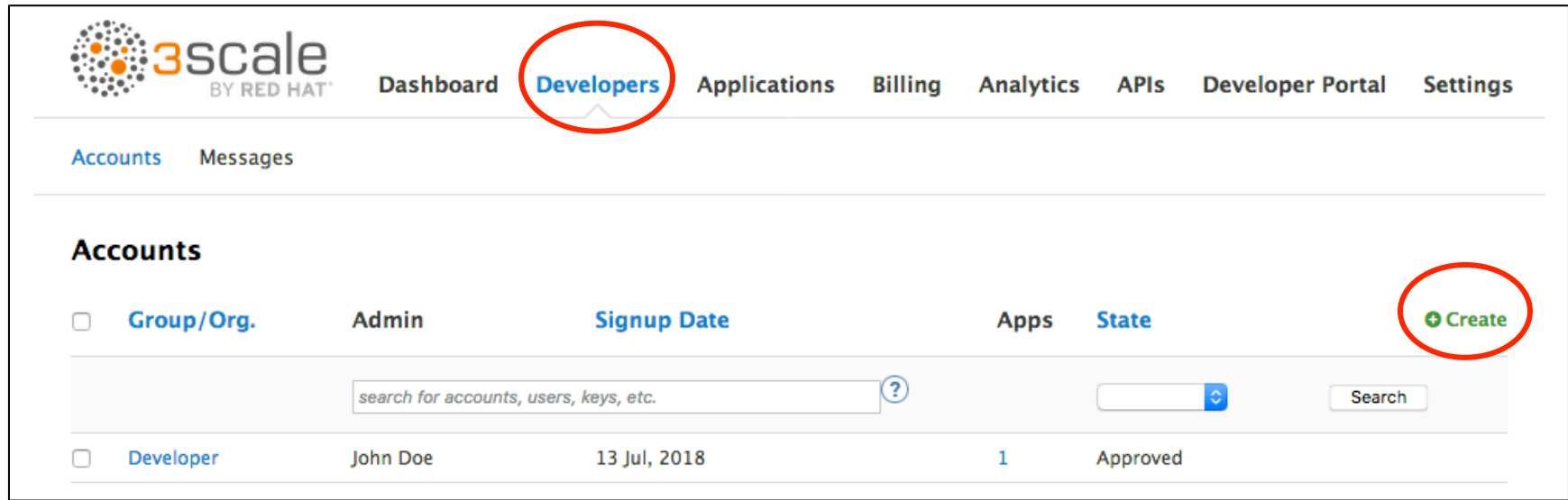
Create Method

Repeat the process above and add a new method with the values:

Friendly name* **Get Employee by ID**

System name* **getEmployeeByID**

Create a Developer Account



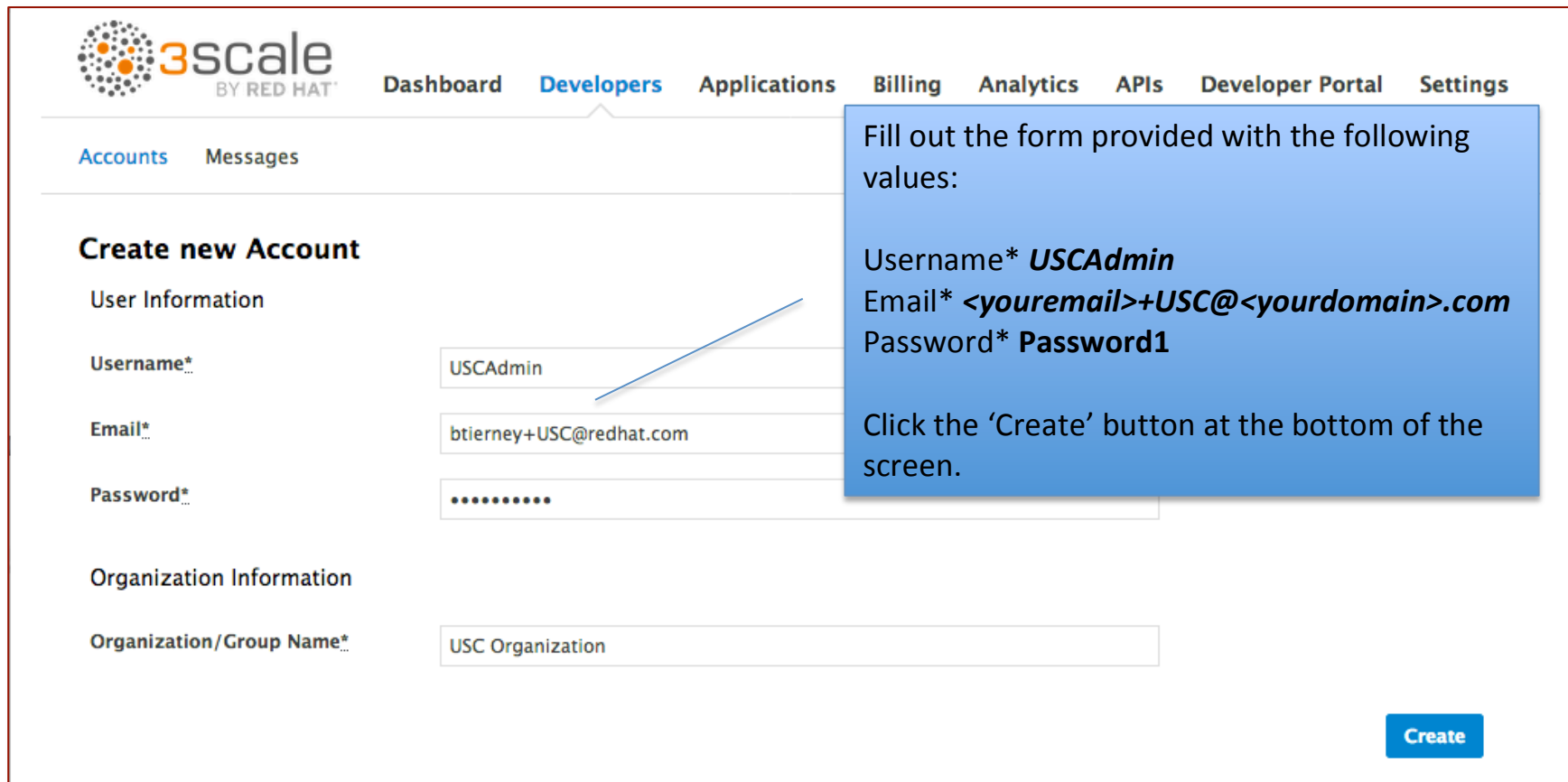
The screenshot shows the 3scale Developer Portal interface. The top navigation bar includes links for Dashboard, Developers (circled in red), Applications, Billing, Analytics, APIs, Developer Portal, and Settings. Below the navigation bar, there are tabs for Accounts and Messages. The main content area is titled 'Accounts' and displays a table of accounts. The table has columns for Group/Org., Admin, Signup Date, Apps, and State. A 'Create' button (circled in red) is located in the top right corner of the table. Below the table, there is a search bar with the placeholder text 'search for accounts, users, keys, etc.' and a 'Search' button. A single account is listed in the table: a Developer named John Doe, signed up on 13 Jul, 2018, with 1 app and an 'Approved' state.

<input type="checkbox"/>	Group/Org.	Admin	Signup Date	Apps	State
<input type="checkbox"/>	Developer	John Doe	13 Jul, 2018	1	Approved

From the top menu, click on the 'Developers' link. Notice that there is a default account already setup for you. We're going to add a new account and associate it with our new Service API.

Click on the 'Create' link toward the lower right side of the page.

Fill out Developer Form



3scale
BY RED HAT

Dashboard **Developers** Applications Billing Analytics APIs Developer Portal Settings

Accounts Messages

Create new Account

User Information

Username* USCAdmin

Email* btierney+USC@redhat.com

Password*

Organization Information

Organization/Group Name* USC Organization

Create

Fill out the form provided with the following values:

Username* **USCAdmin**
Email* **<youremail>+USC@<yourdomain>.com**
Password* **Password1**

Click the 'Create' button at the bottom of the screen.

Developer Detail Screen

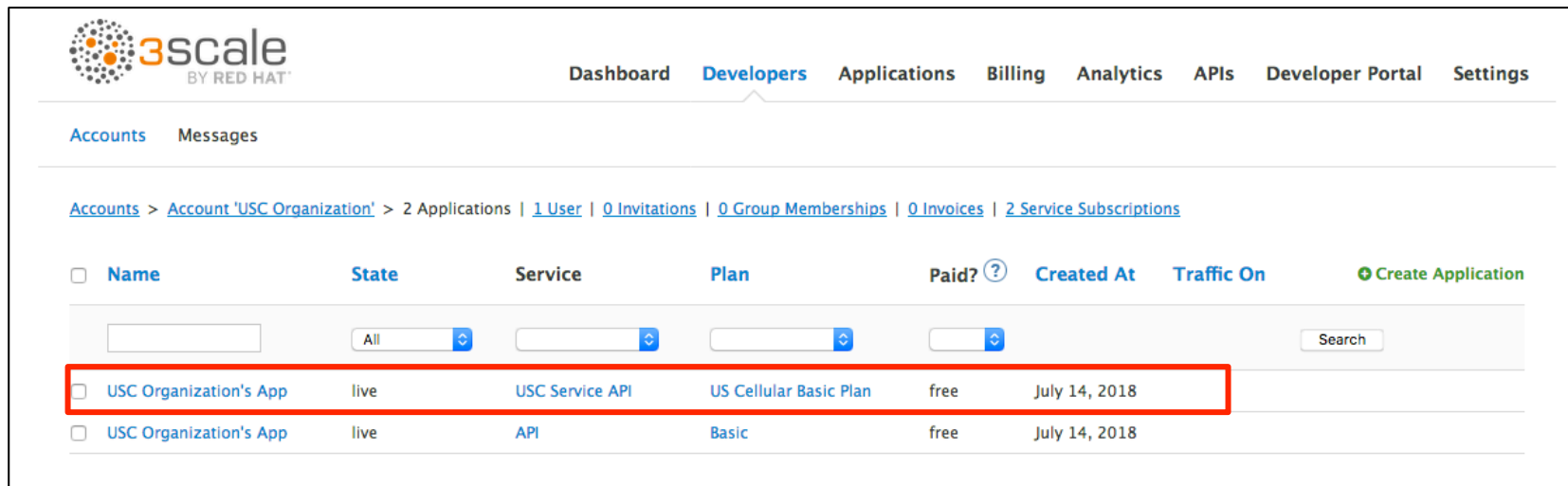
The screenshot shows the 3scale BY RED HAT interface. The top navigation bar includes links for Dashboard, Developers, Applications, Billing, Analytics, APIs, Developer Portal, and Settings. Below this, there are sub-links for Accounts and Messages. The main breadcrumb path is 'Accounts > Account 'USC Organization' > 2 Applications | 1 User | 0 Invitations | 0 Group Memberships | 0 Invoices | 2 Service Subscriptions'. The '2 Applications' link is circled in red. Below the breadcrumb, the page title is 'USC Organization: Account Summary'. There are two action links: 'Send message' and 'Edit'. The main content area is divided into two columns. The left column contains a table with the following details:

Organization/Group Name	USC Organization
Status	Approved
Administrator	USCAdmin (btierney+USC@redhat.com)
Signed up on	July 14, 2018 21:21

The right column contains the 'Account Plan: Default' section with a 'Customize' link. Below the table, there is a 'Billing Status' section indicating that 'Monthly billing is enabled' with a 'Disable' button.

Click on the link shown about. The value (1,2 ..) may vary.

View User Key



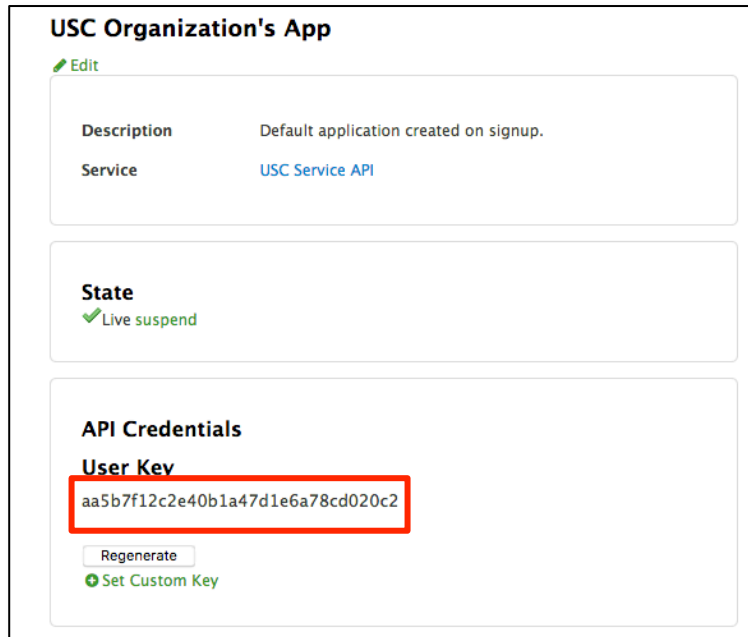
3scale
BY RED HAT

Dashboard Developers Applications Billing Analytics APIs Developer Portal Settings

Accounts Messages

Accounts > Account 'USC Organization' > 2 Applications | 1 User | 0 Invitations | 0 Group Memberships | 0 Invoices | 2 Service Subscriptions

<input type="checkbox"/>	Name	State	Service	Plan	Paid? [?]	Created At	Traffic On	Create Application
<input type="checkbox"/>	USC Organization's App	live	USC Service API	US Cellular Basic Plan	free	July 14, 2018		
<input type="checkbox"/>	USC Organization's App	live	API	Basic	free	July 14, 2018		



USC Organization's App

[Edit](#)

Description Default application created on signup.

Service [USC Service API](#)

State
✔ Live suspend

API Credentials

User Key
aa5b7f12c2e40b1a47d1e6a78cd020c2

[Regenerate](#)

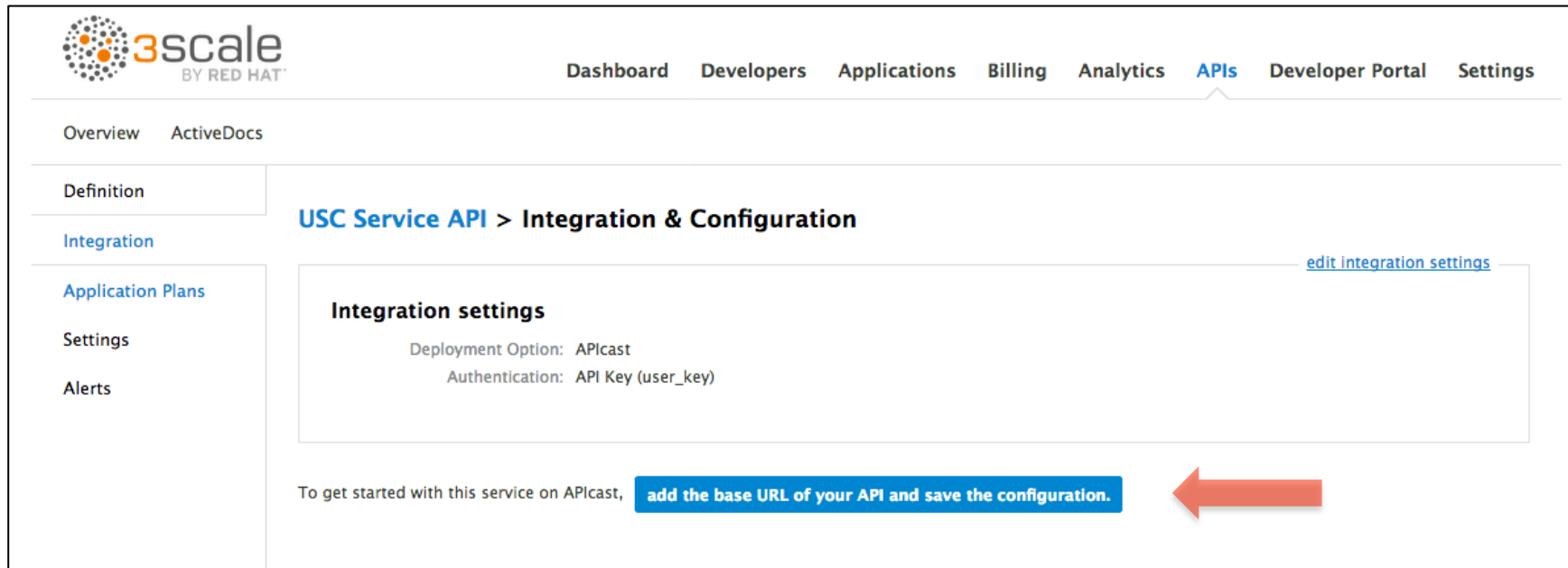
[Set Custom Key](#)

Click the **'USC Organization's App'** link. Be sure to choose the one with the US Cellular Basic Plan as highlighted in the image above.

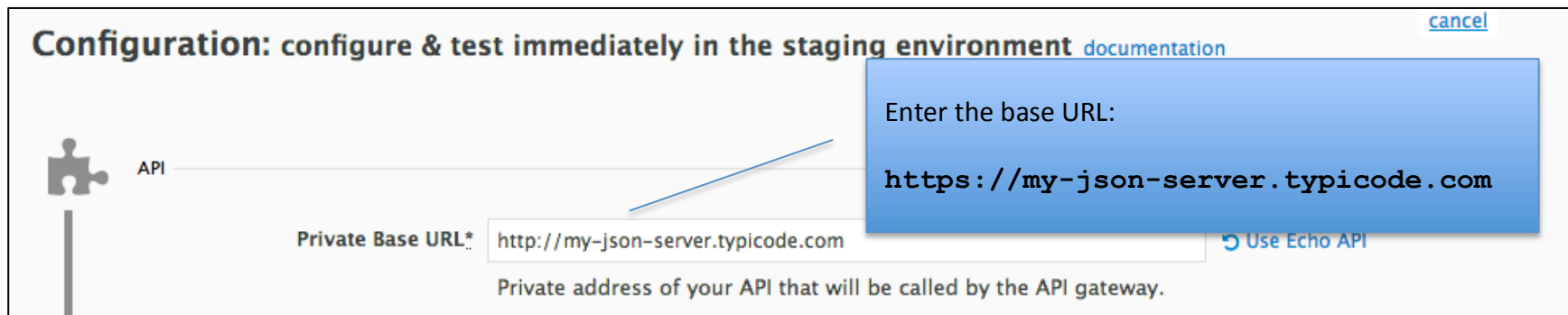
You will be presented with some details information including the API Credentials **'User Key'**. This is shown on the image to the left.

At this point click the **'APIs'** link at the top of the page, and then click the **'Integration'** link under the **'USC Service API'**.

Integration and Configuration



The screenshot shows the 3scale BY RED HAT interface. The top navigation bar includes Dashboard, Developers, Applications, Billing, Analytics, APIs (highlighted), Developer Portal, and Settings. On the left, a sidebar lists Overview, ActiveDocs, Definition, Integration (highlighted), Application Plans, Settings, and Alerts. The main content area is titled "USC Service API > Integration & Configuration" and includes a link to "edit integration settings". Under "Integration settings", it shows "Deployment Option: APICast" and "Authentication: API Key (user_key)". A blue button labeled "add the base URL of your API and save the configuration." is highlighted with a red arrow. Below this, text reads: "To get started with this service on APICast, add the base URL of your API and save the configuration."



The screenshot shows the "Configuration: configure & test immediately in the staging environment" page. It features a "cancel" link in the top right. On the left, there is a puzzle piece icon labeled "API". The main form has a "Private Base URL*" field containing "http://my-json-server.typicode.com". A blue callout box points to this field with the text "Enter the base URL: https://my-json-server.typicode.com". Below the field is a description: "Private address of your API that will be called by the API gateway." A "Use Echo API" button is visible on the right.

Integration and Configuration (cont'd)


Verb	Pattern	+	Metric or Method (Define)
GET	/btierney/threescale/employees	1	hits
Add Mapping Rule			

Verb	Pattern	+	Metric or Method (Define)
GET	/btierney/threescale/employees	1	<div>hits ✓ getEmployees getEmployeeByID</div>
Add Mapping Rule			

1. Click the pencil icon on the right to enable editing
2. Enter the URL pattern employees: **/btierney/threescale/employees**
3. Select the drop down and apply the '**getEmployees**' mapping rule
4. Click the 'Add Mapping Rule' link adding the rule for '**getEmployeeByID**'
Using this pattern: **/btierney/threescale/employees/{id}**

Verb	Pattern	+	Metric or Method (Define)
GET	/btierney/threescale/employees	1	getEmploy
GET	/btierney/threescale/employees/{id}	1	getEmploy
Add Mapping Rule			

Integration and Configuration (cont'd)

 CLIENT ?

API test GET request

Optional GET request to a API gateway endpoint. We will use this call to validate your API gateway setup using credentials of the first live application. You can try it yourself by copying the following command into your shell:

```
curl "https://usc-service-clin-apicast-staging.apps.boston.openshiftworkshop.com:443/btierney/threescale/employees?user_key=aa5b7f12c2e40b1a47d1e6a78cd020c2"
```

Hit the test button to check the connections between client, gateway & API.

[Update & test in Staging Environment](#)

```
btierney — -bash — 86x23
{"name": "Offenbacker, Git"}
{
  "id": 23,
  "name": "Pruskowski, Rudolph"
},
{
  "id": 24,
  "name": "Raider, Arne"
},
{
  "id": 25,
  "name": "Sandor, Belt"
},
{
  "id": 26,
  "name": "Sonnee, W. Mann"
},
{
  "id": 27,
  "name": "Schange, Spare"
}
]MacBook-Pro-10:~ btierney$
```

Click the '**Update & test in Staging Environment**' link above

You should notice that the bold line on the left turns green indicating that the configuration is valid and working.

Open up a console window. Copy & paste the provided CURL statement into the command line and hit enter. You should see the employee list.

Analytics


USC Service

Usage

Daily Averages

Top Applications

Hourly Averages



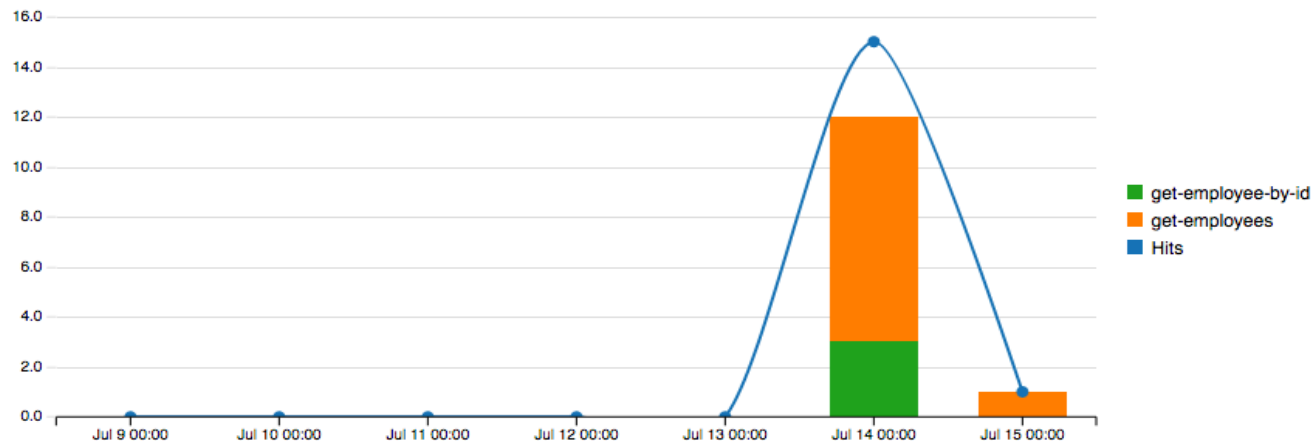
Click the '**Analytics**' link at the top of the screen. Then locate the **USC Service** and select the '**Usage**' link underneath.

The resulting page will display charting and data on the service usage broken down by the methods/metrics we defined earlier.

USC Service > Usage

Show last [24 hours](#) **[7 days](#)** [30 days](#) [12 months](#) | from 07/09/2018 until 07/15/2018 per day

16.0 Hits ▼

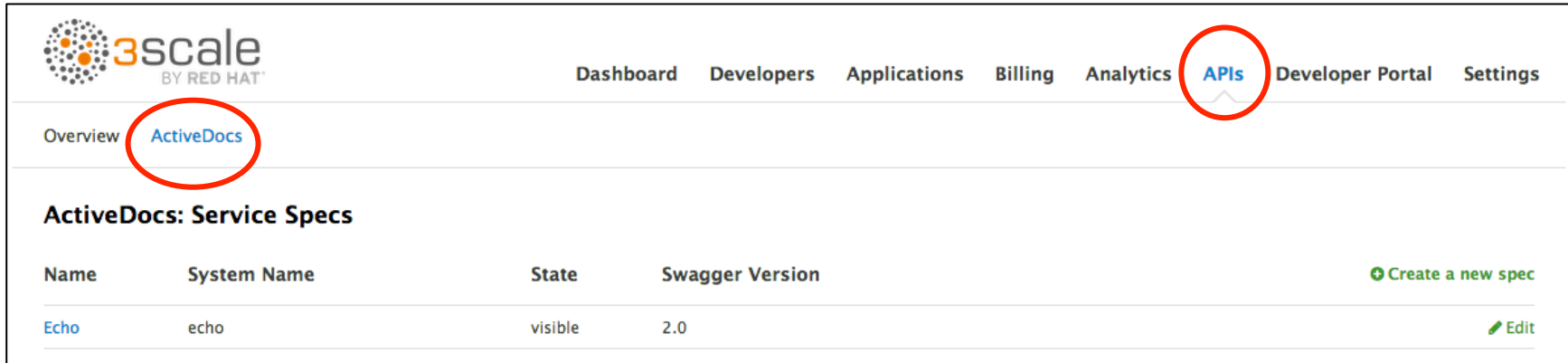


[Download CSV](#)

Method	From	To	Traffic
Hits	09 Jul 2018 00:00:00 CDT	15 Jul 2018 23:59:59 CDT	16
get-employees	09 Jul 2018 00:00:00 CDT	15 Jul 2018 23:59:59 CDT	10
get-employee-by-id	09 Jul 2018 00:00:00 CDT	15 Jul 2018 23:59:59 CDT	3

Using time zone (GMT-06:00) Central Time (US & Canada)

Import Swagger Docs

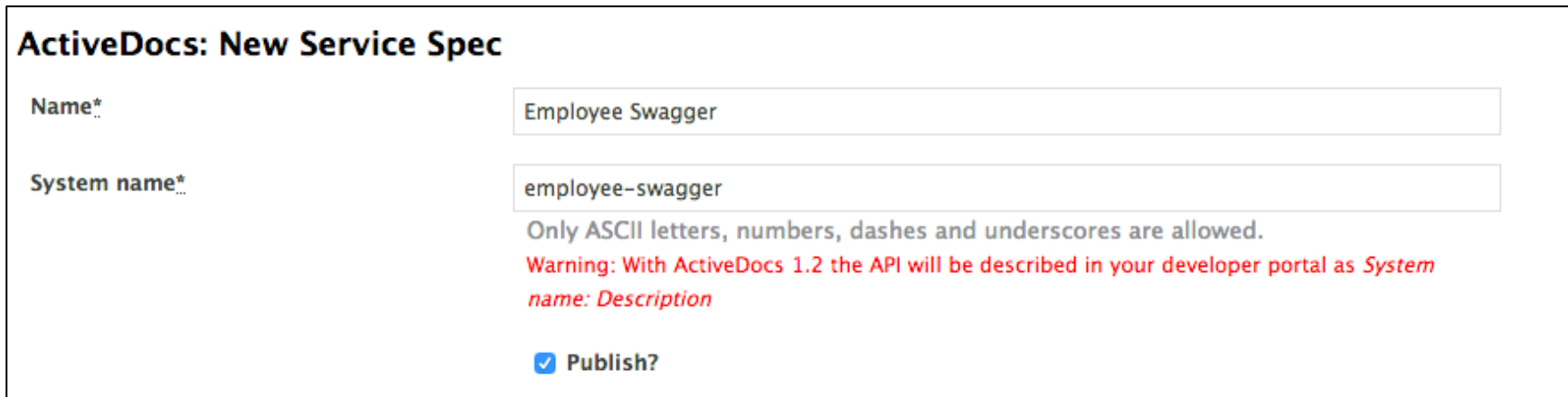


The screenshot shows the 3Scale BY RED HAT interface. The top navigation bar includes links for Dashboard, Developers, Applications, Billing, Analytics, APIs (circled in red), Developer Portal, and Settings. Below the navigation bar, the 'ActiveDocs' link is circled in red. The main content area is titled 'ActiveDocs: Service Specs' and contains a table with the following data:

Name	System Name	State	Swagger Version	
Echo	echo	visible	2.0	Edit

At the top right of the table, there is a link 'Create a new spec' with a plus icon.

1. Click on the 'APIs' link on the top of the page
2. Click on the 'ActiveDocs' link under the 3Scale icon
3. Click on the 'Create a new spec' link at the right of the page



The screenshot shows the 'ActiveDocs: New Service Spec' form. It contains the following fields and options:

- Name***: A text input field containing 'Employee Swagger'.
- System name***: A text input field containing 'employee-swagger'.
- Validation rules**: Below the system name field, it states 'Only ASCII letters, numbers, dashes and underscores are allowed.' and a red warning: 'Warning: With ActiveDocs 1.2 the API will be described in your developer portal as System name: Description'.
- Publish?**: A checkbox that is checked.

1. Name* **Employee Swagger**
2. System name* **employee-swagger**
3. Check **Publish**

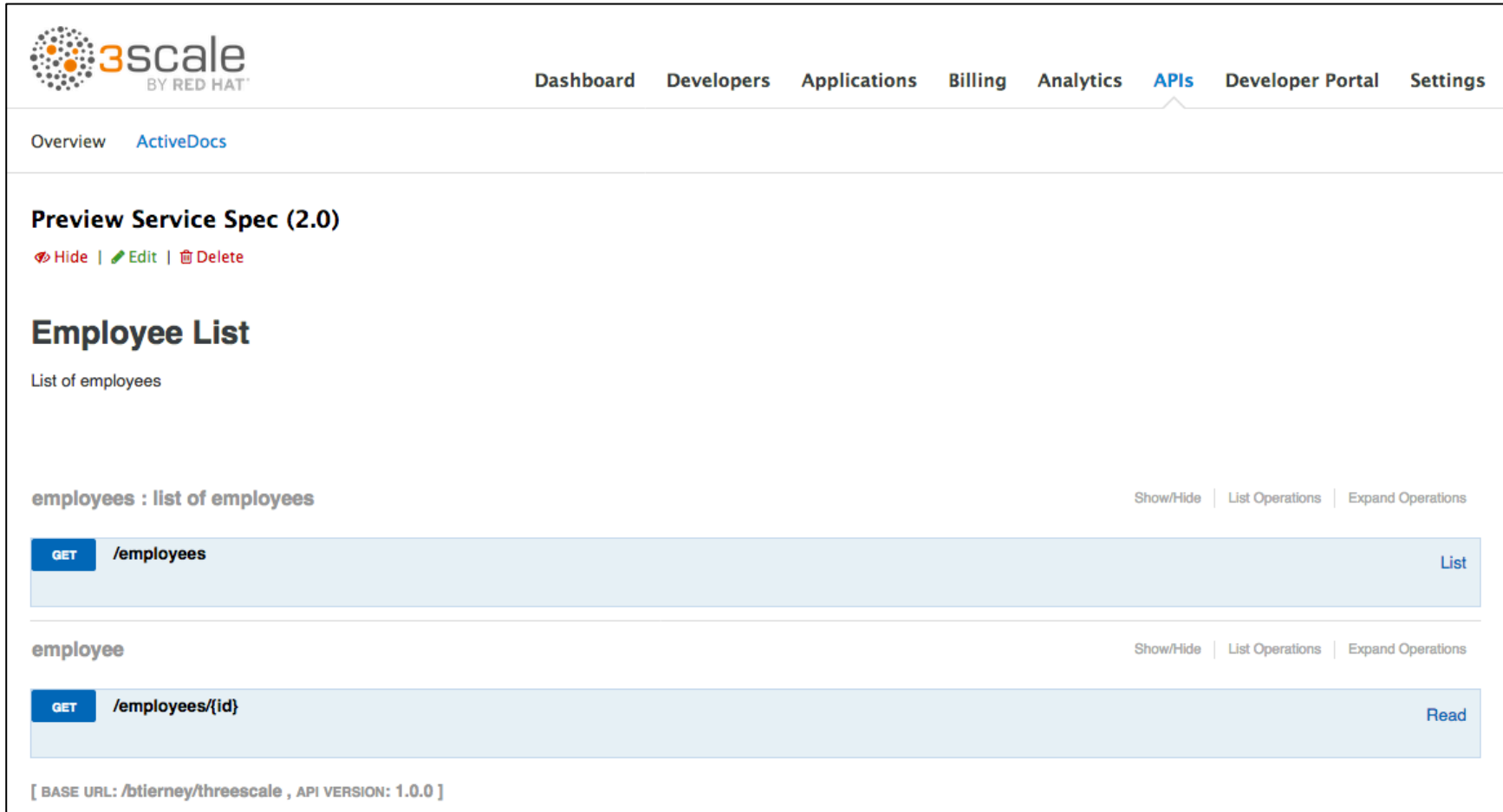
Import Swagger Docs (cont'd)

1. Go to the following link and copy the emp_swagger.json file to your clipboard
https://github.com/btierney/threescale/blob/master/emp_swagger.json
2. Paste the contents of the clipboard into the API JSON Spec* text field as shown below
3. Click the 'Update Service' button at the bottom of the screen.

API JSON Spec*

```
1 {  
2   "swagger": "2.0",  
3   "info": {  
4     "title": "Employee List",  
5     "description": "List of employees",  
6     "version": "1.0.0"  
7   },  
8   "host": "my-json-server.typicode.com",  
9   "basePath": "/btierney/threescale",  
10  "schemes": [  
11    "http"  
12  ],  
13  "consumes": [  
14    "application/json"  
15  ],
```

Import Swagger Docs (cont'd)



The screenshot displays the 3scale API management interface. At the top, the 3scale logo (BY RED HAT) is on the left, and a navigation menu includes Dashboard, Developers, Applications, Billing, Analytics, APIs (highlighted), Developer Portal, and Settings. Below the navigation, there are tabs for Overview and ActiveDocs. The main content area is titled 'Preview Service Spec (2.0)' and includes links for Hide, Edit, and Delete. The service is named 'Employee List' with the description 'List of employees'. Below this, the service is listed as 'employees : list of employees' with options to Show/Hide, List Operations, and Expand Operations. The first operation is a GET request to '/employees', which is highlighted in a light blue box with a 'List' button on the right. The second operation is a GET request to '/employees/{id}', also highlighted in a light blue box with a 'Read' button on the right. At the bottom, a footer indicates the base URL is '/btierney/threescale' and the API version is '1.0.0'.

3scale
BY RED HAT™

Dashboard Developers Applications Billing Analytics **APIs** Developer Portal Settings

Overview **ActiveDocs**

Preview Service Spec (2.0)
🔍 Hide | ✎ Edit | 🗑 Delete

Employee List
List of employees

employees : list of employees Show/Hide | List Operations | Expand Operations

GET /employees **List**

employee Show/Hide | List Operations | Expand Operations

GET /employees/{id} **Read**

[BASE URL: /btierney/threescale , API VERSION: 1.0.0]

Test the service by clicking 'GET' on the '/employees'.

Import Swagger Docs (cont'd)

GET **/employees** [List](#)

Response Class (Status 200)
An array of employees

Model

Example Value

```
[
  {
    "id": 0,
    "name": "string"
  }
]
```

Response Content Type application/json

Parameters

Parameter	Value
user_key	<input type="text" value="aa5b7f12c2e40b1a47d1e6a78cd020c2"/>

Try it out!

First user key from latest 5 applications

USC Organization's App - USC Service API aa5b7f12c2e40b1a47d1e6a78cd020c2

USC Organization's App - API 69786a8eff168563f9f061c1d09fdbb1

Developer's App - API 07d88581a19c26c52edd5827a39892ce

Hover over the 'user_key' field. Notice how the system will conveniently offer some choices
Click the 'Try it out!' button to see the results

Import Swagger Docs (cont'd)

Request URL

http://my-json-server.typicode.com/btierney/threescale/employees?user_key=7d795ac1bb57441fb352b95bed079ec7

Response Body

```
[
  {
    "id": 1,
    "name": "Awalt, Frank",
    "title": "CEO"
  },
  {
    "id": 2,
    "name": "Brennenstuhl, Susan"
  },
  {
    "id": 3,
    "name": "Boulds, Zach"
  },
  {
    "id": 4,
    "name": "Bankhead, Elissa"
  },
  {
    "id": 5,
```

Response Code

200

Developer Portal

1. Download image assets from: <http://>
2. Click the 'Developer Portal' link at the top of the page.
3. Select 'New File' from the drop down on the right and then enter the values as follows:
4. Section* **images**
5. Path* **/images/logo-usc.jpg**
6. Click 'Create File'

Upload File

Section*

|— images

Path*

/images/logo-usc.jpg

☐ Downloadable

Checked sets the content-disposition to 'attachment.'

Attachment

Browse...

logo-usc.jpg

Tag list

New File

Create File

Developer Portal (cont'd)

[Dashboard](#) [Developers](#) [Applications](#) [Billing](#) [Analytics](#) [APIs](#) [Developer Portal](#) [Settings](#)

[Visit Developer Portal](#)

New page

[New Page](#)

Title*

US Cellular Test Page

Section*

. Root

Path*

/usctestpage

Does not depend on a selected section.

Layout

Main layout

▼ Advanced options

System name

Content type

text/html

Can be [HTML](#), [CSS](#), [Javascript](#) or an arbitrary MIME type.

☒ Liquid enabled

Process Liquid tags and drops?

Select the 'Developer Portal' link above
Click on the 'New Page' drop-down box
Enter values as follows:

Title* **US Cellular Test Page**
Section* **Root**
Path* **/usctestpage**

Click the '**Liquid Enabled**' checkbox

Developer Portal (cont'd)

From the link below, copy the source for the swagger javascript into the Text Box as shown. Click the 'Create Page' button on the bottom right of the page. Then click **Publish** and **Save**.

Portal Script:

<https://github.com/btierney/threescale/blob/master/uscellular/portalscript>

```
1 <h3>Active Docs/Swagger 2.0 Documentation</h3>
2
3 {% active_docs version: "2.0" services: "employee-swagger" %}
4
5 <script type="text/javascript">
6 $(function () {
7
8   window.swaggerUi.load();
9 });
10 </script>
```

Create Page

Delete

Revert

Preview

Publish

Save

