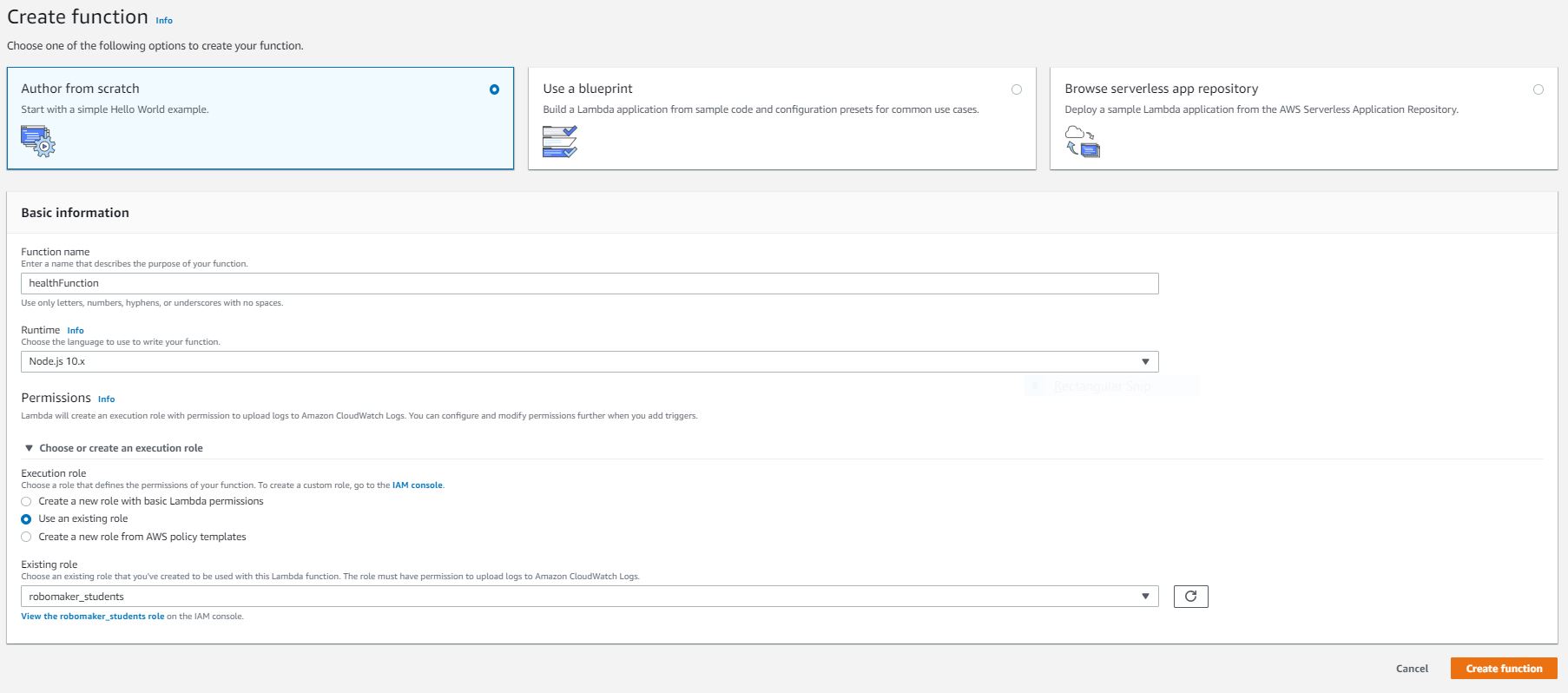
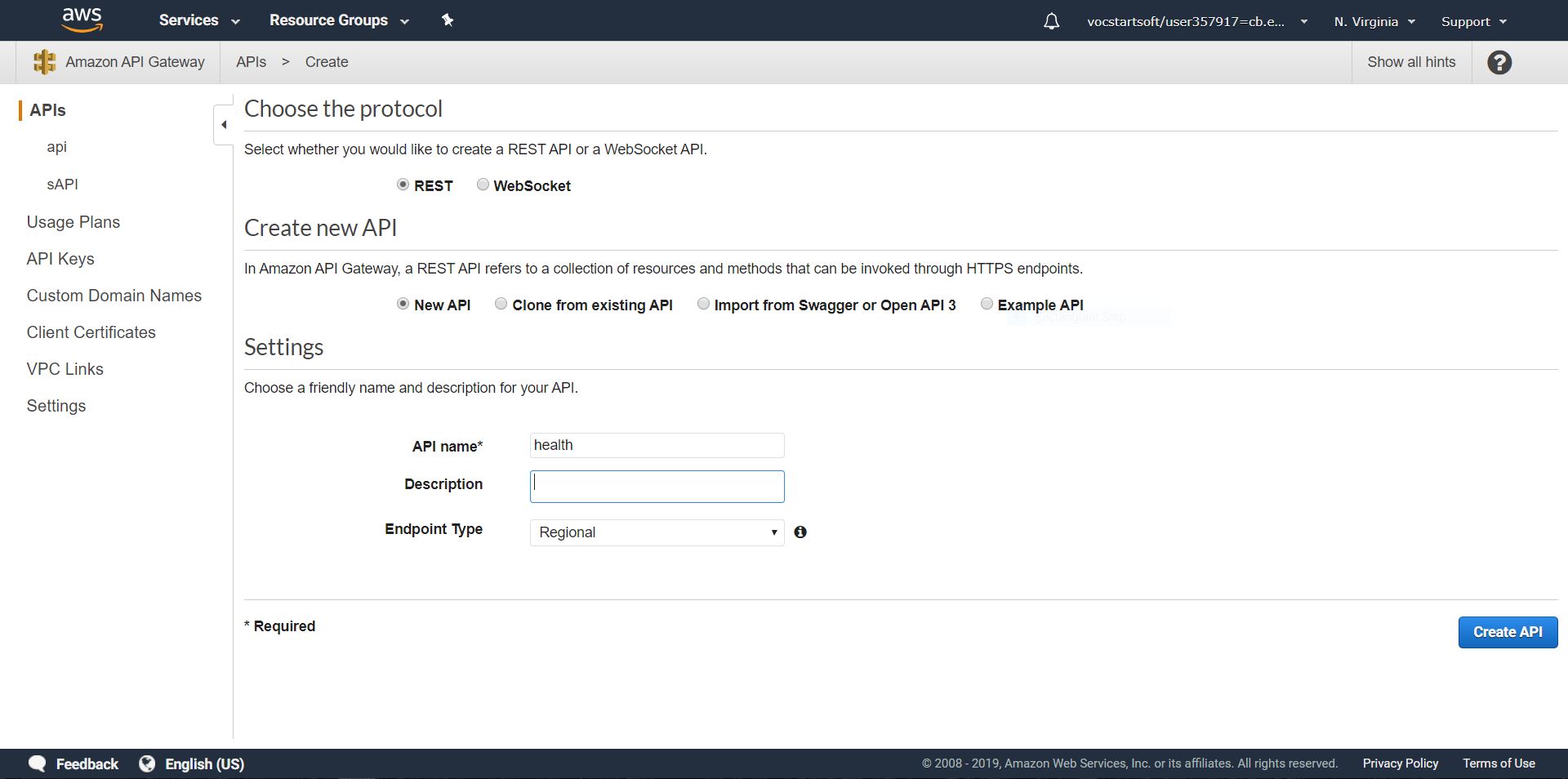
**Health Records**

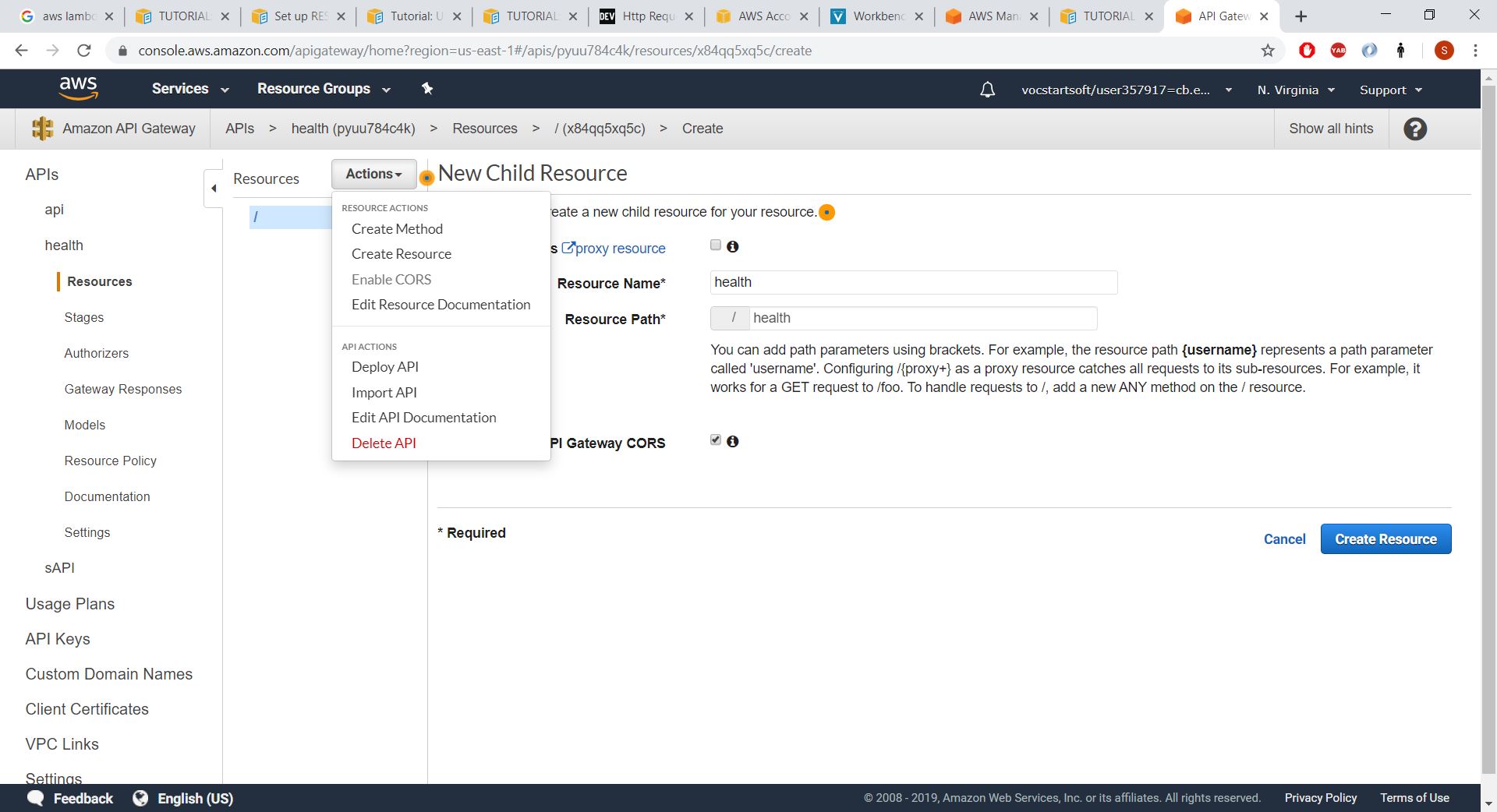
* Creation of Lambda Function:
* Choose Create function.
* Choose Author from scratch.
* Under Basic information, do the following:
* In Function name, enter Healthfunction.
* From the Runtime dropdown list, choose Node.js 8.10.
* Under Permissions, expand Choose or create an execution role. From the Execution role dropdown list, choose Choose exsisting
* Choose Create function.



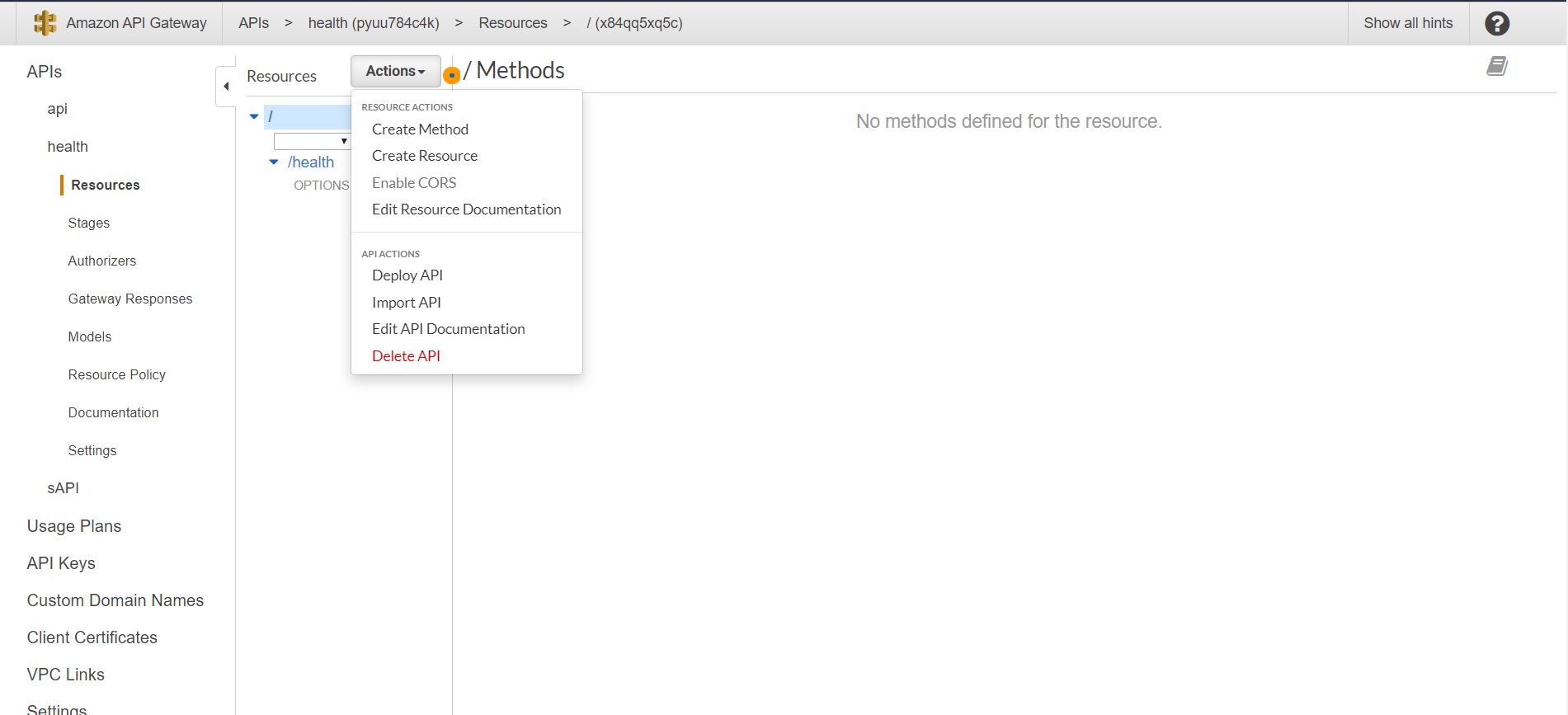
* Creation of API gateway:
* To create an API, choose Create new API (for creating the first API) or Create API (for creating any subsequent API). Next, do the following:
* Choose New API.
* Type a name in API Name.
* Optionally, add a brief description in Description.
* Choose Create API

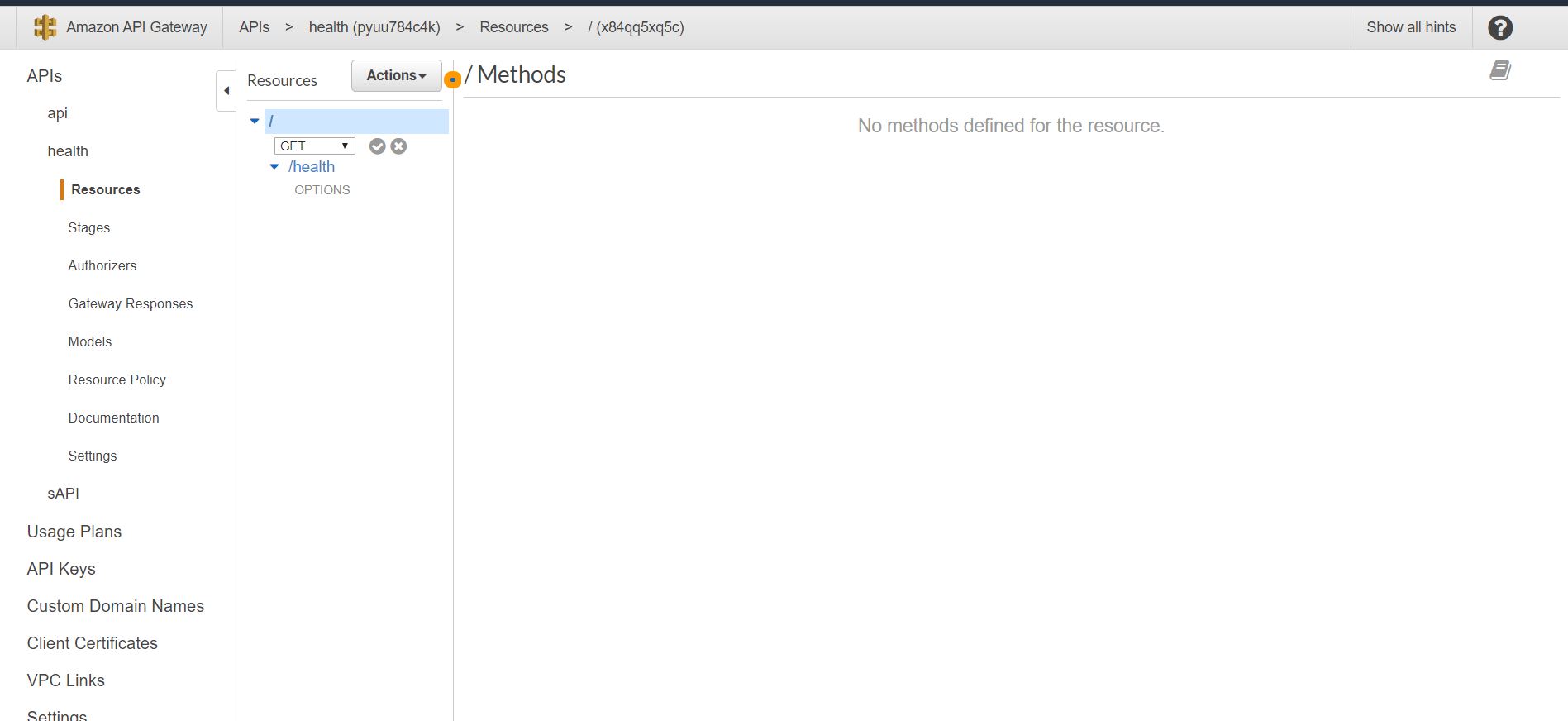


* As a result, an empty API is created. The Resources tree shows the root resource (/) without any methods. In this exercise, we will build the API with the HTTP custom integration of the Healthcare website, choose Actions and then choose Create Resource.
* Type in Resource Name, leave the Resource Path value as given, choose Enable API Gateway CORS, and choose Create Resource.

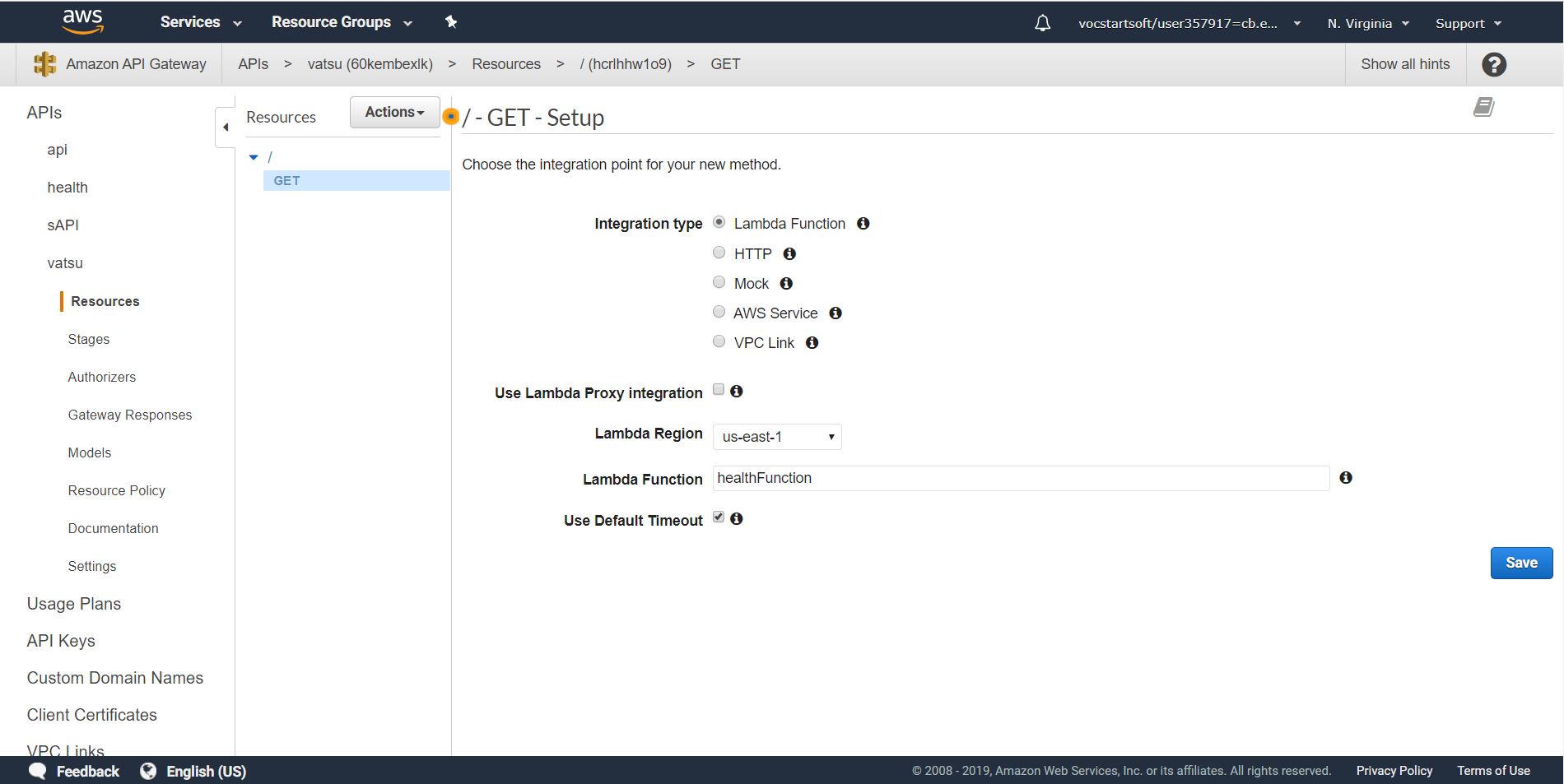


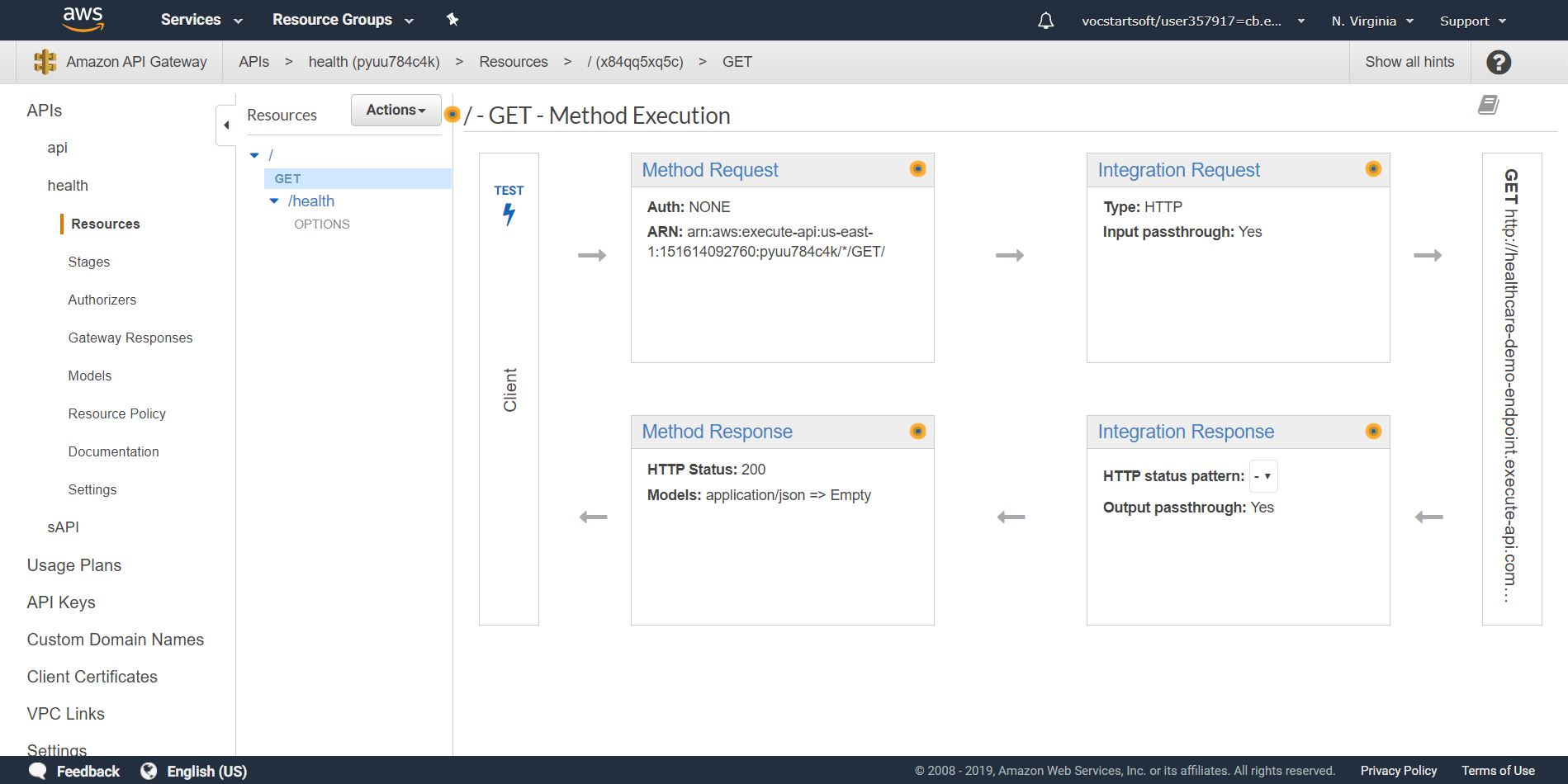
* To expose a GET method on the /pets resource, choose Actions and then Create Method.



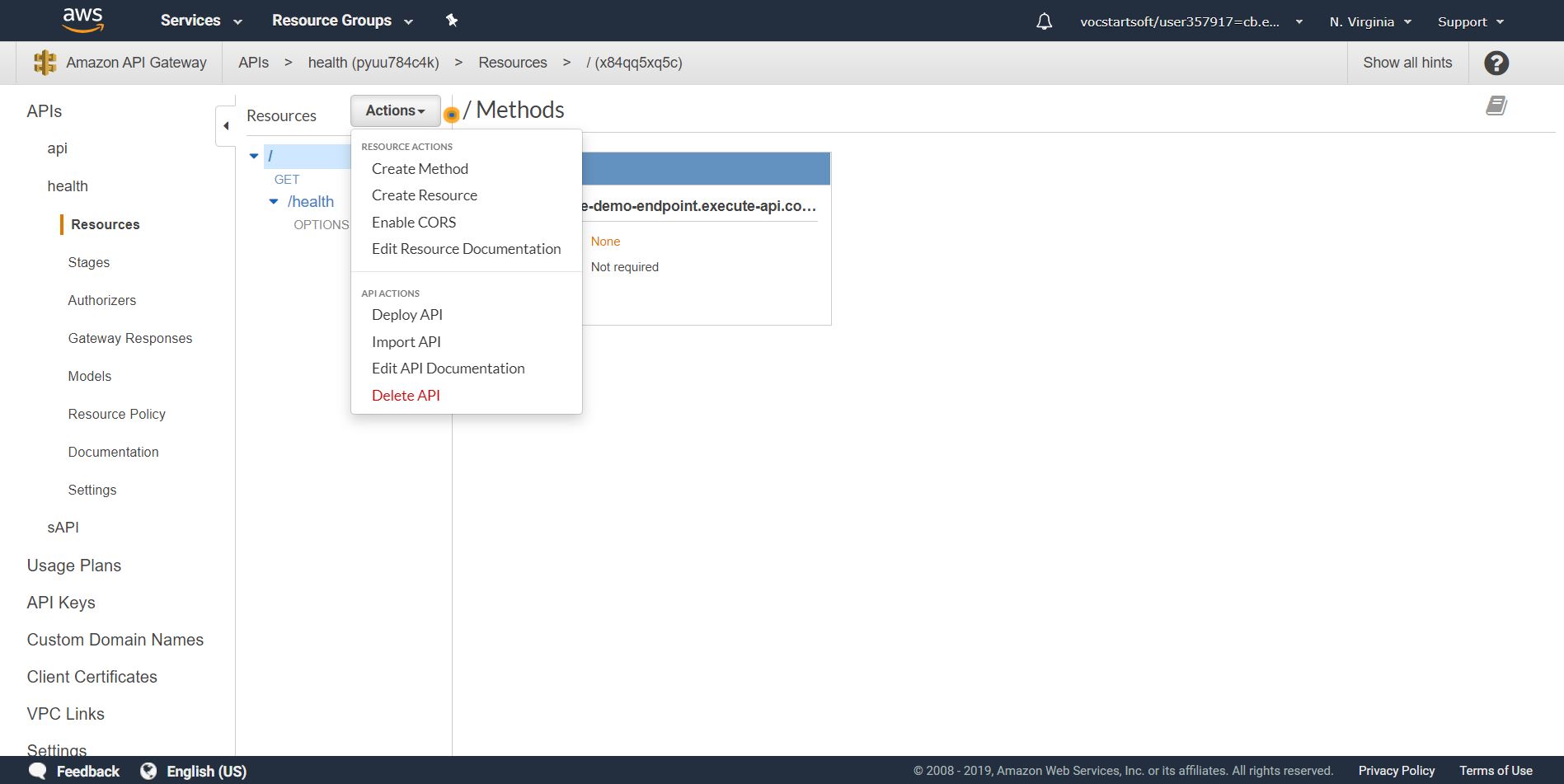


* In the method's Setup pane, select Lambda Function for Integration type, select GET, type the Endpoint URL value, leave all other settings as default, and then choose Save.

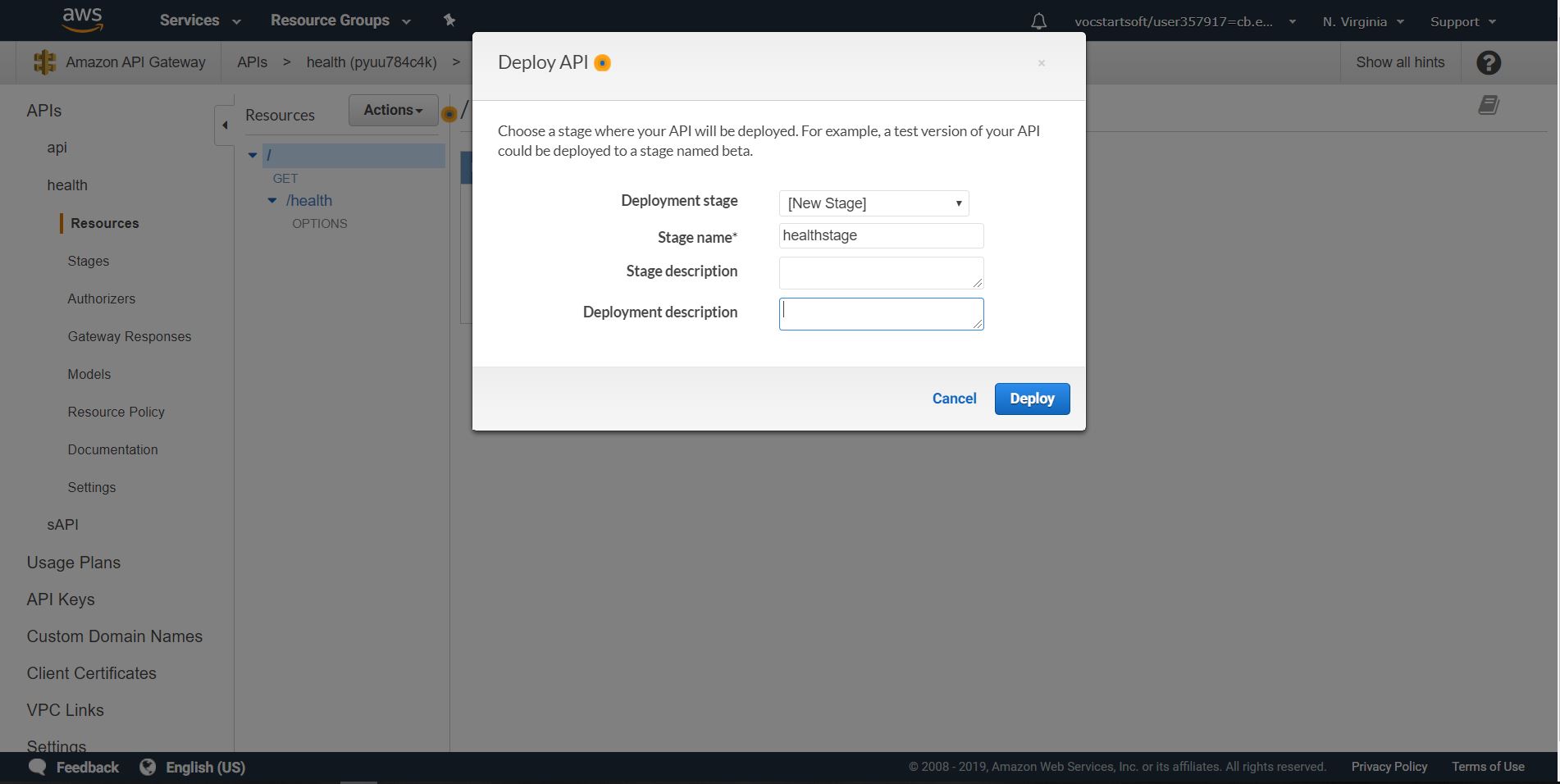




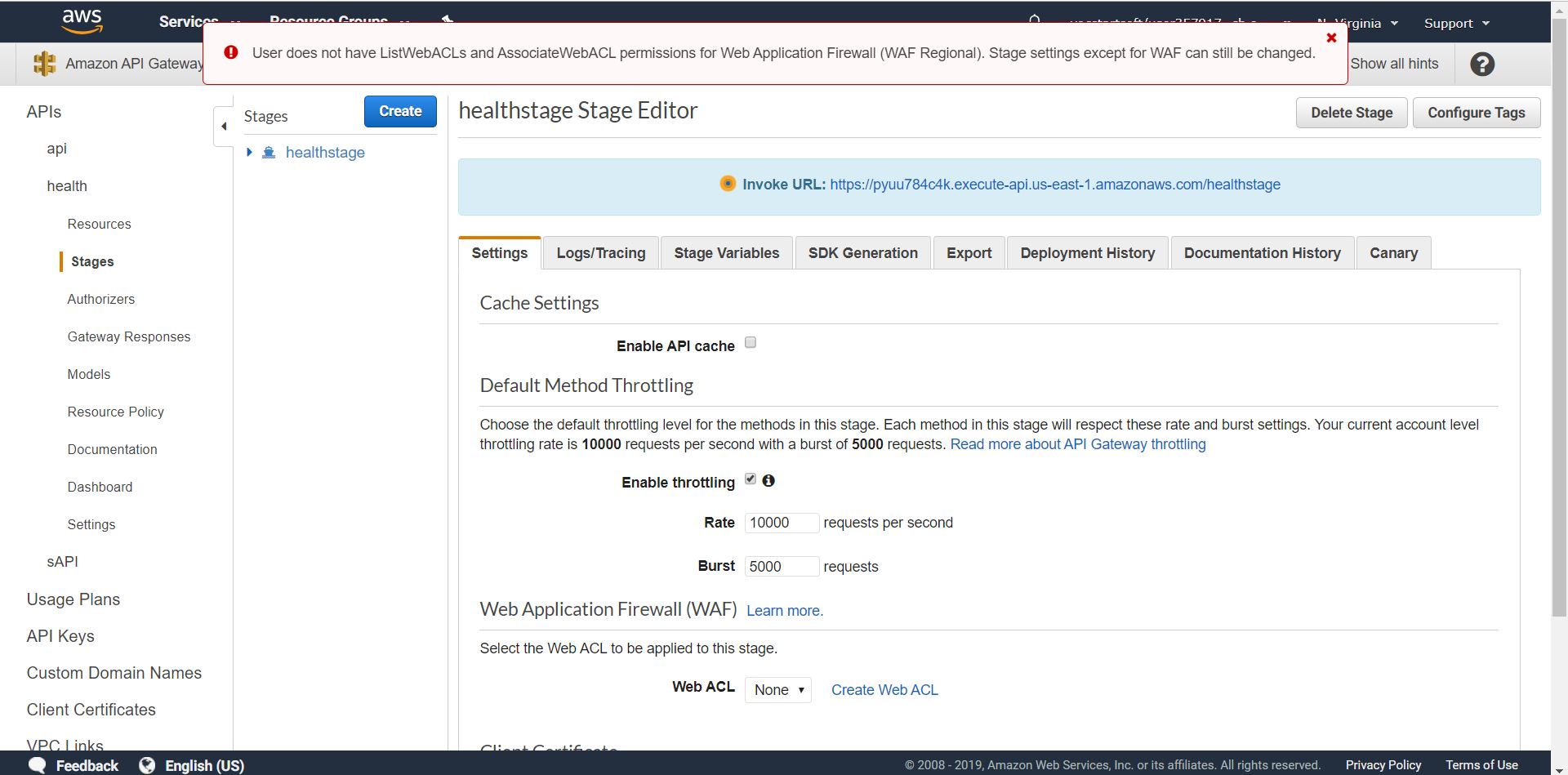
* To deploy the API, select the API and then choose Deploy API from the Actions drop-down menu.



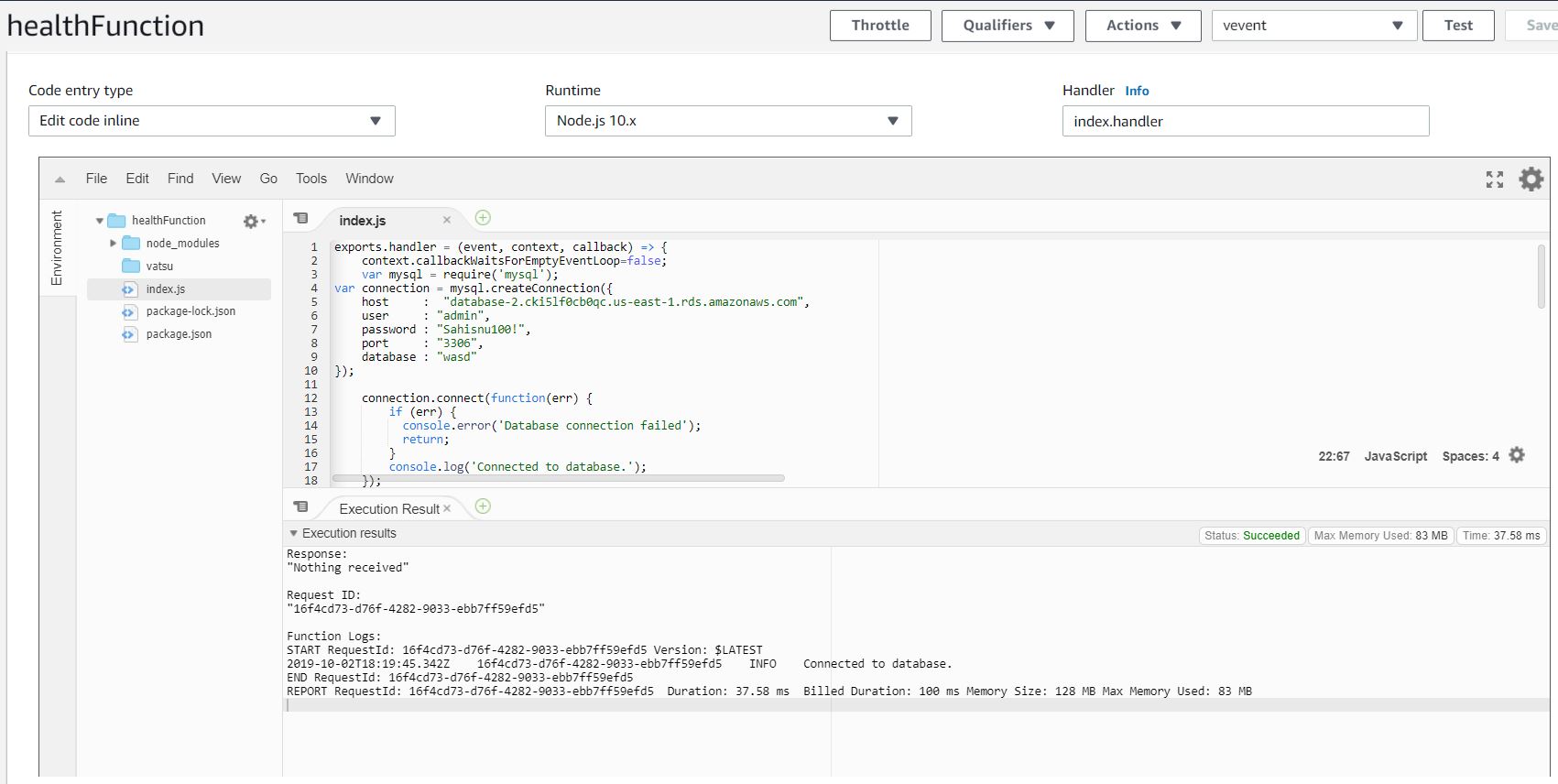
* In the Deploy API dialog, choose [New Stage] enter a name Healthstage in the Stage name input field; optionally, provide a description in Stage description and/or Deployment description; and then choose Deploy.



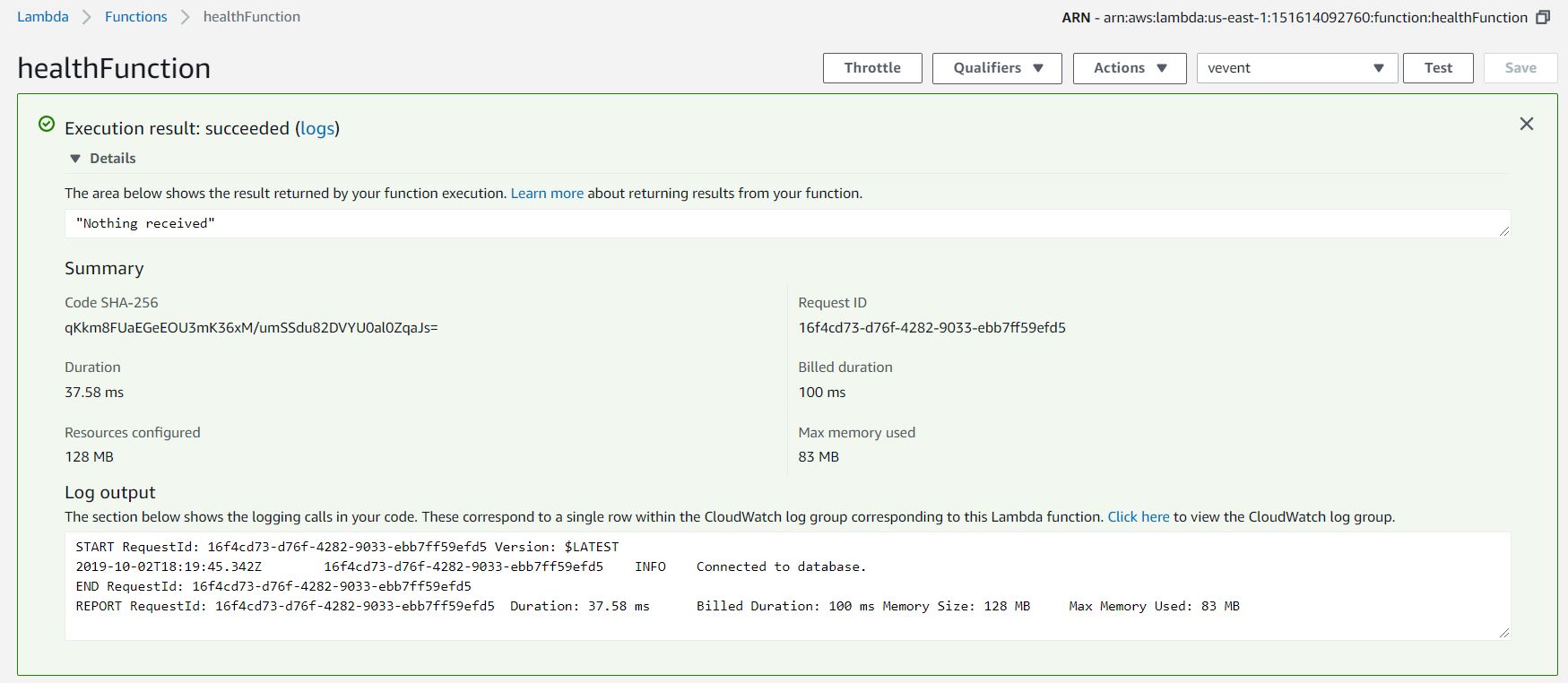
* Now the url will be displayed:



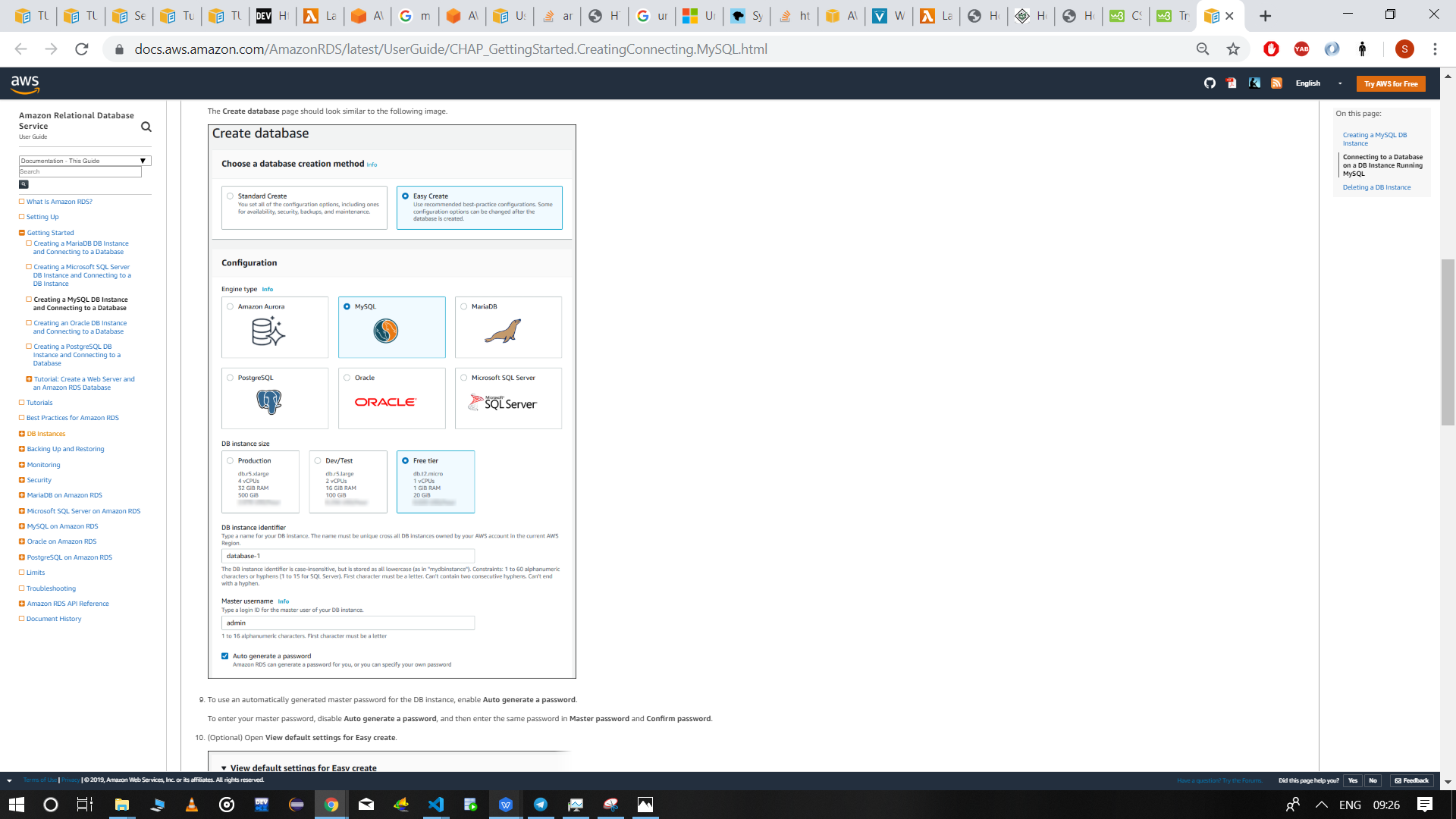
* lambda-code; Upload Zip file and have this in your index.js file,then click save to upload the file in lambda . After saving click test to check the output of your code



* Execute function result will be displayed as



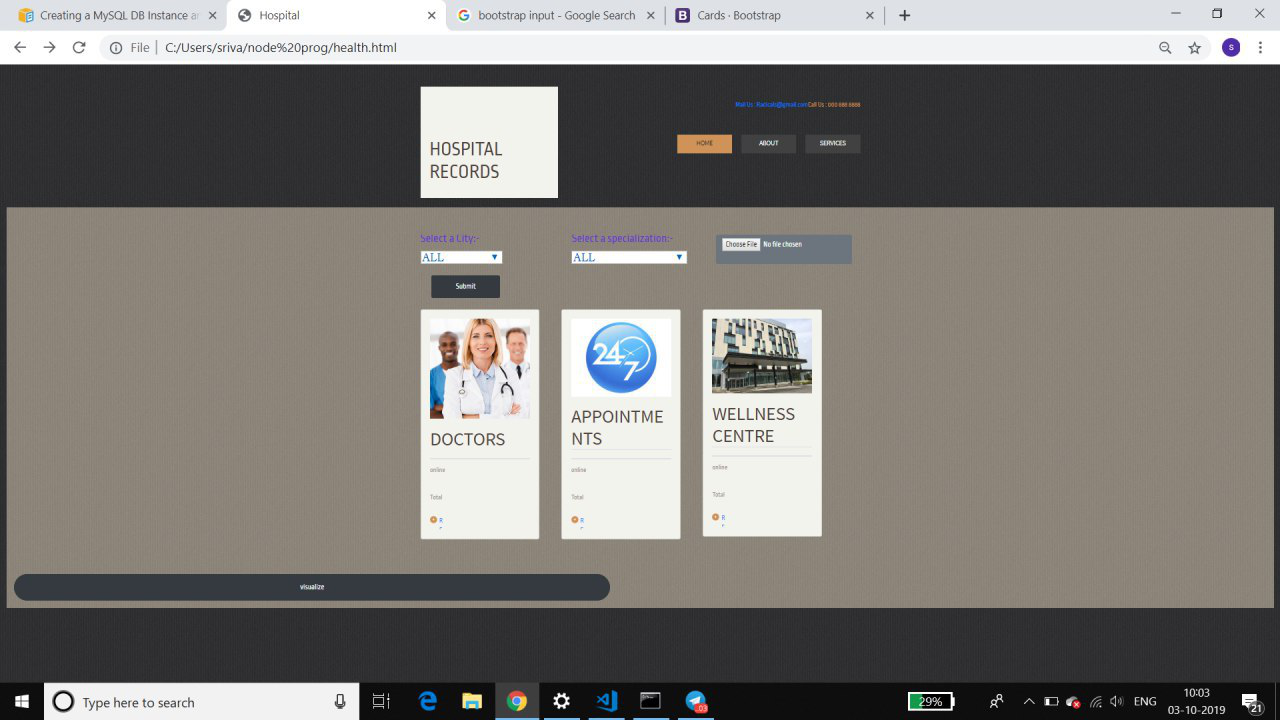
* Creation of RDS:
* To create a MySQL DB instance with Easy Create enabled
* Sign in to the AWS Management Console and open the Amazon RDS console .
* In the upper-right corner of the Amazon RDS console, choose the AWS Region in which you want to create the DB instance.
* In the navigation pane, choose Databases.
* Choose Create database and make sure that Easy Create is chosen.



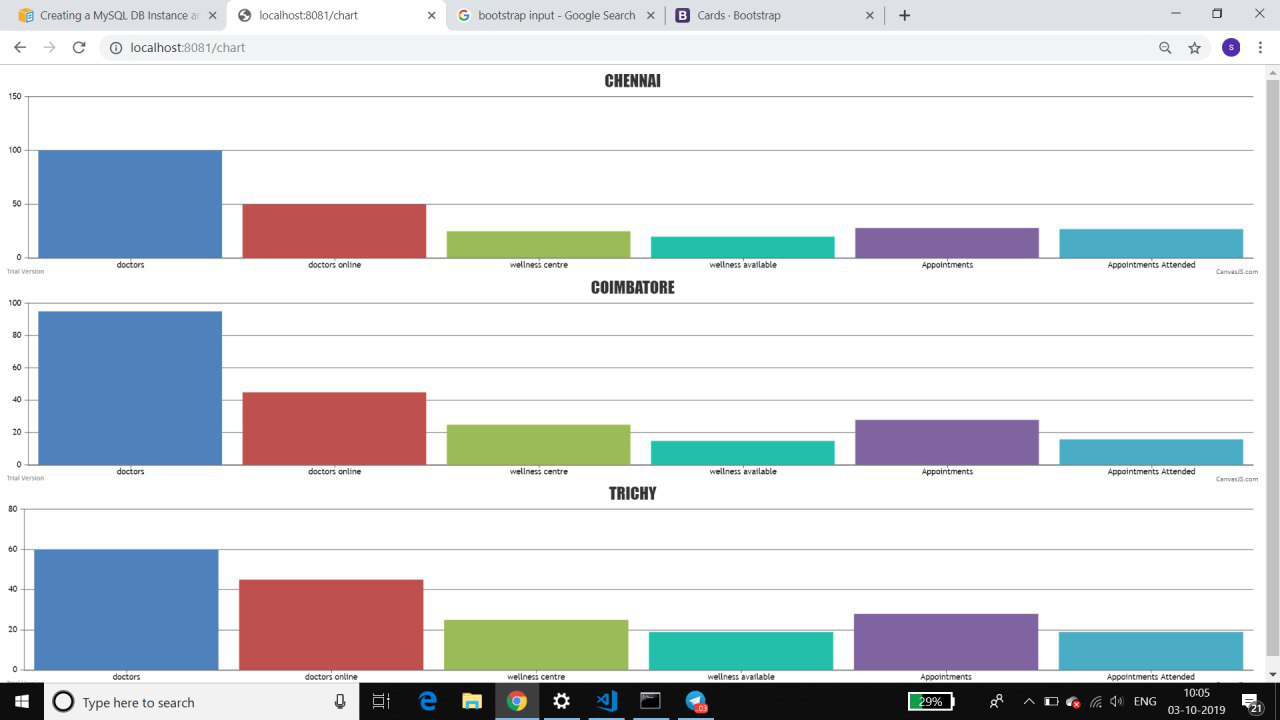
In Configuration, choose MySQL.

* For DB instance size, choose Free tier.
* For DB instance identifier, enter a name for the DB instance, or leave the default name.
* For Master username, enter a name for the master user, or leave the default name.
* The Create database page should look similar to the following image.
* To use an automatically generated master password for the DB instance, enable Auto generate a password.
* To enter your master password, disable Auto generate a password, and then enter the same password in Master password and Confirm password.
* (Optional) Open View default settings for Easy create
* You can examine the default settings used when Easy Create is enabled. If you want to change one or more settings during database creation, choose Standard Create to set them. The Editable after database creation column shows which options you can change after database creation. To change a setting with No in that column, use Standard Create. For settings with Yes in that column, you can either use Standard Create or modify the DB instance after it is created to change the setting.
* Choose Create database.database will be created succesfully;

Html Design(Front-End);-



Visulaization:-



Html Design (Back-End);-

