GO Shopping Application Supporting Documentation

RYAN FINLAYSON

April 11, 2021

1 Design Decisions

The lambda function uses AWS API Gateway to receive API calls to the endpoint. The lambda function contains the go code of the back-end server required to process API calls. The file was uploaded to AWS lambda as a zip file.

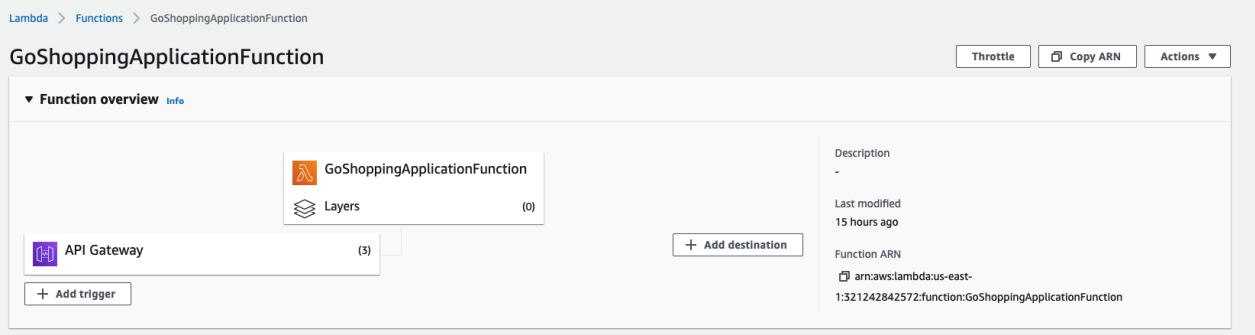

Figure 1: AWS Lambda Setup

2 API Functionality

The Shopping Application supports two API methods: *GET* and *POST*.

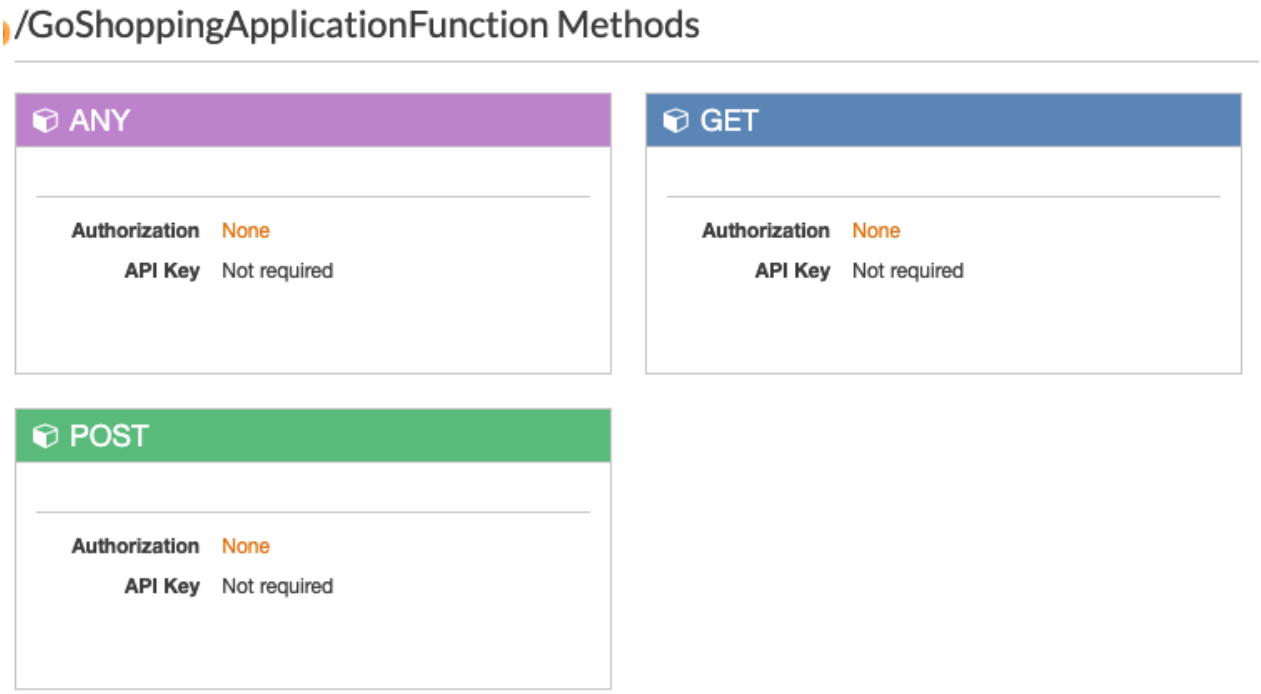

Figure 2: Supported API Methods

GET	**Requirements**	**Purpose**
/GoShoppingApplicationFunction		Get all products stored by API
/GoShoppingApplicationFunction?id=<id_value>		Get a specific product by ID
POST	**Requirements**	**Purpose**
/GoShoppingApplicationFunction	Request Body	Add a new product

Here is an example of the request body required for the POST request:

```
{
    "id": 0,
    "name": "Product Name",
    "description": "Example Description",
    "price": 10.95,
    "image_url": "<img_url>"
}
```

3 Application

The application consists of two main parts: main product page and settings page. Ideally, the settings page would not be visible to the user, as the user is not expected to add items themselves to the site, however, for demonstration purposes to exercise the API, an add product feature has been implemented.

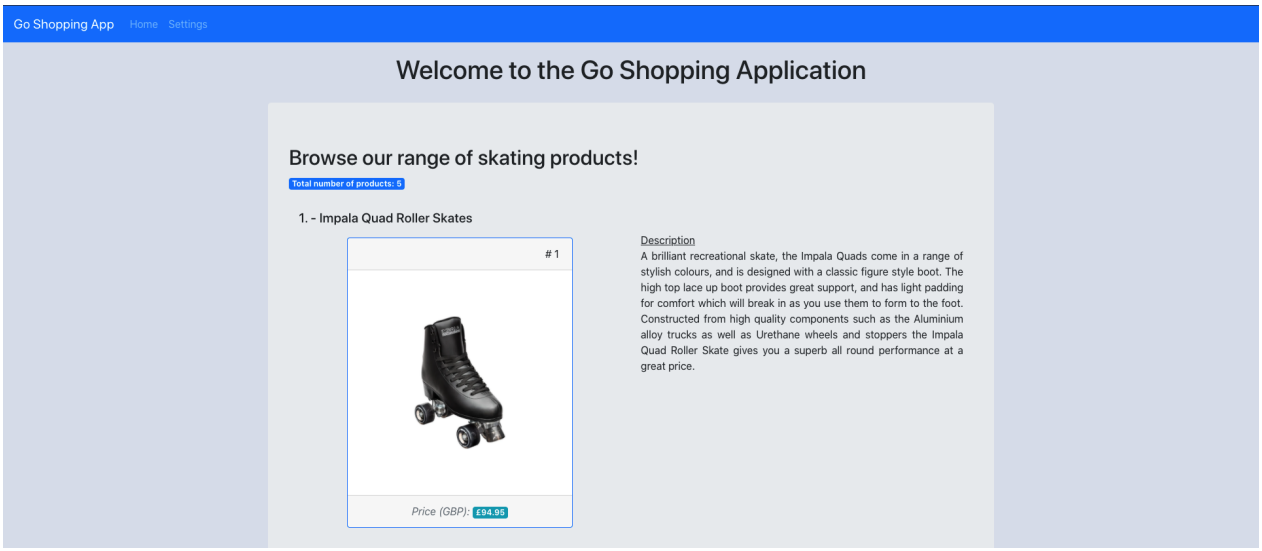

Figure 3: Main Product Page

The main application page displays all the products by performing a GET request using Reacts functional components rather than Reacts class component alternative. The GET request uses the default root to retrieve *all* products stored by the API.

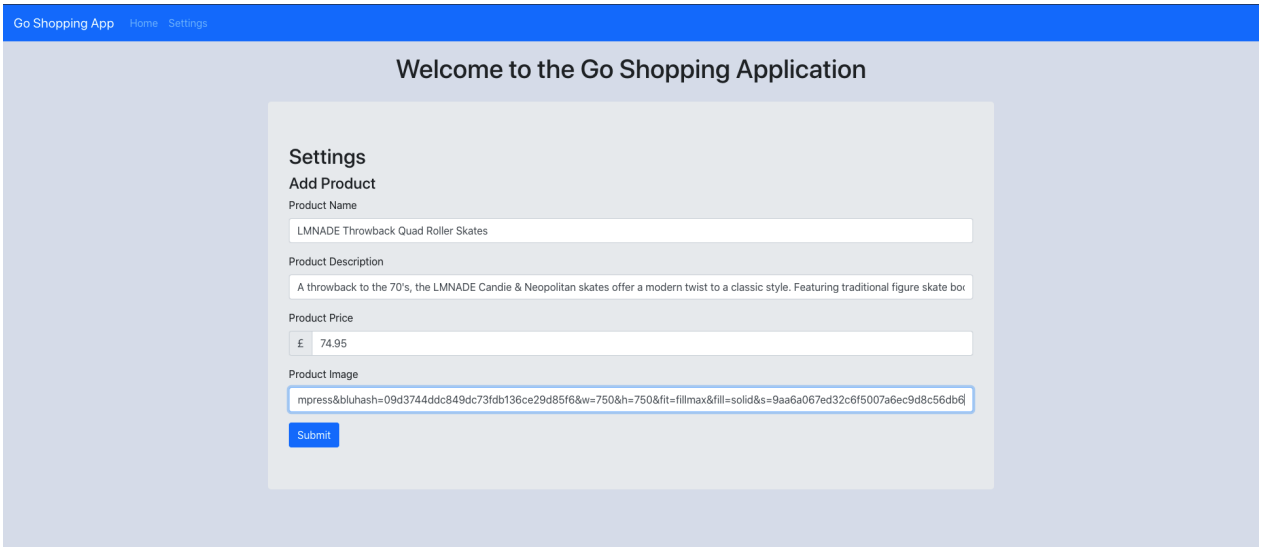

Figure 4: Add Product in Settings Page

By completing the react-bootstrap form, data is used to populate the main products page, by performing a POST request with the AWS API. The information in the form is converted into a JSON request body.