

Create An Account In Flight API

Date	05 Nov 2022
Team ID	PNT2022TMID32270
Project Name	Developing A Flight Delay Prediction Model Using Machine Learning

Flight API:

To make the model available via API, We created an endpoint in sagemaker as the entry point for the inference. The endpoint sets up a Flask Web Server that responds to incoming inference requests via HTTP. We then made it available to external users with AWS Lambda and a serverless framework, and coded a simple go function to perform sanity checks on the user inputs and call the inference endpoint. Thus, the Flight Delay Prediction API was born! Now anyone can call the API and incorporate ML-trained flight delay prediction.

Account:

The screenshot displays the RapidAPI interface for the 'Flight Data API'. The page title is 'Flight Data API Documentation' with a 'FREE' badge. Below the title, it says 'Travelpayouts Data API - the way to get travel insights for your site or blog. Get flight price trends and find popular destinations for your customers.'

The main content area shows the 'GET Non-stop tickets' endpoint. It includes a description: 'Returns the cheapest non-stop tickets for the selected route with departure/return date filters.' Below this, there are fields for 'Personal Account' (37_Pavithra ks), 'RapidAPI App' (default-application_6860341), and 'Request URL' (rapidapi.com). There is also a section for 'Header Parameters' with 'X-Access-Token'.

On the right side, there is a 'Code Snippets' section with a '(Node.js) Axios' snippet. The code is as follows:

```
const axios = require('axios');

const options = {
  method: 'GET',
  url: 'https://travelpayouts-travelpayouts-flight-data-v1.p.rapidapi.com/v1/prices/direct/',
  params: {destination: 'LED', origin: 'MOW'},
  headers: {
    'X-Access-Token': 'undefined',
    'X-RapidAPI-Key': 'b029114482msh7bac39ec7c2b474p1fae20jsnf785ca872c40',
    'X-RapidAPI-Host': 'travelpayouts-travelpayouts-flight-data-v1.p.rapidapi.com'
  }
};

axios.request(options).then(function (response) {
  console.log(response.data);
}).catch(function (error) {
  console.error(error);
});
```