

Overview

Heroku app to handle long running request: importing ticket audits.

Please see [US258](#) for more information.

Running the app:

Locally:

Run the following command locally and open your browser to <http://localhost:5000/canvas>

```
. setupConfigVars.sh
```

then run the following:

```
heroku local web
```

To run the worker

```
heroku local worker
```

Heroku:

<https://ticket-scale-import.herokuapp.com/canvas>

Visualforce (using signed request)

/apex/TEST__TicketScaleImport

Connected App

[Connected App 1](#)

[Connected App 2](#)

Debugging the app

run node-inspector in a separate browser window

then run heroku local webdebug

@TODO - include more info

Fixing permissions

Sometimes people have different umask settings on their machines, meaning files generated appear modified when instead they just have different permissions.

Navigate to the base of the repo and run this shellscript:

```
git config --global --add alias.permission-reset '!git diff -p -R --no-color | grep -E "^(diff|(old|new) mode)" --color=never | git apply'
```

Now, you can run this git command

```
git permission-reset
```

All files are reset back to the permission known within git.

Common Links

For more information about using Node.js on Heroku, see these Dev Center articles:

- [Getting Started with Node.js on Heroku](#)
- [Heroku Node.js Support](#)
- [Node.js on Heroku](#)
- [Best Practices for Node.js Development](#)
- [Using WebSockets on Heroku with Node.js](#)
- [Platform connect](#)
- [JSForce](#)
- [Lightning Design System - LDS](#)

how was this made

- [nodejs buildpack](#)
- [canvas developer guide](#)
- [made a canvas app](#)
- [authorize the canvas app for users/profiles/perm sets](#)
- [expose out the canvas app in vf](#)
- [demo on dev0](#)

Example canvas tag in vf

```
<apex:canvasApp applicationName="TEST Canvas Demo"
  namespacePrefix="" onCanvasAppLoad="onCanvasLoad"
  height="600px" width="100%"
  parameters="{ eventId: '{! EventDateTime__c.Event__c }', eventDateTimeId: '{! Event
DateTime__c.Id }', eventName: '{! JSENCODE( EventDateTime__c.Event__r.EventName__c ) }
' }"
/>
```