

BeamNG apps

Introduction

The user interface (apps as well) of BeamNG.drive are written using [AngularJS](#) framework. Some familiarity with the framework is useful, although not necessary. In this context, apps are just simple directives in the module *beamng.apps*. The 3 important parts of each app are

1. An *app.js* file with all the code used by the app
2. An *app.json* file that contains information about the app
3. An *app.png* image file that shows up in the app selector

1. the app.js file

The structure of a BeamNG app directive generally follows this structure:

```
angular.module('beamng.apps')
.directive('myApp', ['StreamsManager', function (StreamsManager) {
  return {
    template: '[Some HTML Content]',
    replace: true,
    restrict: 'EA',
    link: function (scope, element, attrs) {
      var streamsList = /*An optional list of streams that will be used in the app*/;

      // Make the needed streams available.
      StreamsManager.add(['sensors']);

      // Make sure we clean up after closing the app.
      scope.$on('$destroy', function () {
        StreamsManager.remove(streamsList);
      });

      scope.$on('streamsUpdate', function (event, streams) {
        /* Some code that uses the streams' values */
      });
    }
  };
}]);
```

The example above is an example of app that uses the vehicle's streams (which is by far the most common case). However this is not always the case, the apps can be used for just about anything. Again, the code must be included in the "link" function of the app.js file, for example

```
angular.module('beamng.apps')
.directive('myApp', ['StreamsManager', function (StreamsManager) {
  return {
    template: '<button ng-click="hello()">Click Me</button>',
    replace: true,
    restrict: 'EA',
    link: function (scope, element, attrs) {
      scope.hello = function () {
        // do something here.
      };
    }
  };
}]);
}]])
```

2. The app.json file

This file is really simple and just holds information about the app, useful for loading it. However, one must be very careful with the fields' names, as typos or missing fields might affect the proper execution of the app. The file contents have the form

```
{
  "name" : "My App",
  "author": "Me",
  "version": "0.1",
  "description": "Just a tutorial app",
  "directive": "myApp",
  "domElement": "<my-app></my-app>",
  "css": { "width": "150px", "height": "150px", "top": "200px", "left": "200px" }
}
```

The required fields are:

- name The display name for the app
- author Your name
- version A version number
- description A small description of the app
- directive The name of the directive (the same as in the app.js file)
- domElement The DOM element that will actually hold the app

The domElement is determined from the *directive* name, and is just a conversion from "camelCase" to "lisp-case", so in our example it would be 'myApp' -> If you are not sure about that, going through the [docs](#) might be useful.

- css The default CSS attributes to be used when the app is first launched. These are of course width and height and a top/bottom, left/right property that also states the screen corner to which the app will be aligned.

3. The app.png file

No explanation needed for this one, it's the image that shows up in the app selection view. Just make sure its size is 250 x 120.

Use your app

In order to make your app visible from the game, move the folder with the app's files to your mods directory (usually Documents\BeamNG.drive\ui\modules\apps). The app will then be available from the selection list.

Bonus

If you feel that this is a tedious and/or error-prone task, we agree with you! There is a [yeoman generator](#) available to guide you through the process and scaffold a skeleton app for you.

Make sure you have [nodejs](#) and [yeoman](#) in your system. NodeJS can be downloaded from [here](#) and installing yeoman is just a matter of running

```
npm install -g yo
```

in a console.

Then, you can get the generator by running

```
npm install -g generator-beamng-app
```

To generate your app, run

```
yo beamng-app
```