

The Problem

Deliver a solution that will take a number [0 - 999,999,999,999] and convert it into its spoken equivalent. For example, 167 would convert to "one hundred and sixty seven"

My Solution

Possibly overkill for this problem, but I've used a JS application boilerplate that I put together last year using Backbone.js and Require.js to organise the code. This allows me to separate view and business logic in a familiar (mvc like) way, and write/utilise simple unit tests for the model code.

Important Aspects of the Solution:

- **app/js/models/NumberModel.js** contains the number class/object that constructs the string for the number
- **app/js/tests/numberTest.js** contains the simple unit tests (using QUnit) used during development
- The application can be run by installing into web host and navigating to **http://application_root** in a browser
- Unit tests can be run by navigating to **http://application_root#/tests** in a browser
- View controllers are located in **app/js/views**
- View output templates are located in **app/js/templates**

About the Application Framework

- **backbone.js** - Allows separation of project structure (MVP/C style)
- **underscore.js** - Dependency of backbone.js. utility function library
- **require.js** - file and module loader allows views. models etc to be split out into multiple files.
- **jQuery** - javascript library!
- **QUnit** - simple unit testing

APPLICATION STRUCTURE

```
| --app
|   | --css
|   | --js
|   |   | --collections
|   |   | --models
|   |   | --tests
|   |   | --vendor
|   |   | --views
|   |   | --router.js
|   | --templates
|   | --bootstrap.js
| --index.html
```