Tooling Choices

Octopeer for Bitbucket

Our tooling as of now consists of a build tool (gulp), a layer around javascript (TypeScript), a testing toolkit (Jasmine) and coverage (Istanbul).

To get started we had to use a build tool, the major two build tools for javascript are Grunt and Gulp, due to Arthur's experience with Gulp and lack of experience within the group with the other tool, we started using **Gulp**.

We write most - if not all - of our code in **TypeScript**, which basically adds type checking to Javascript. Helping us catch issues early on and maintain good code quality. It transpiles to Javascript, which requires some extra work for getting coverage on the TypeScript files.

For testing we could either use Karma (which launches a web browser and runs other frameworks inside, such as the ones afterwards), Mocha (testrunner only) + ChaiJS (assertions) + SinonJS (mocking), or Jasmine (all in one, test-runner, assertions and mocking).
We went with **Jasmine**, as having an all-in-one solution would save time configuring the entire toolkit.

For coverage not many tools are available, the primary and most compatible tool we could find was **Istanbul**. Since this tool only gives our coverage on the transpiled version of our source files, we have to convert it back, this is done using **Istanbul-remap** and **gulp-sourcemaps**.

For linting we chose to use **TS Lint**. This tools assists us when it comes to using proper coding style. There are no competing linting tools for TypeScript, so this tooling choice was straightforward.