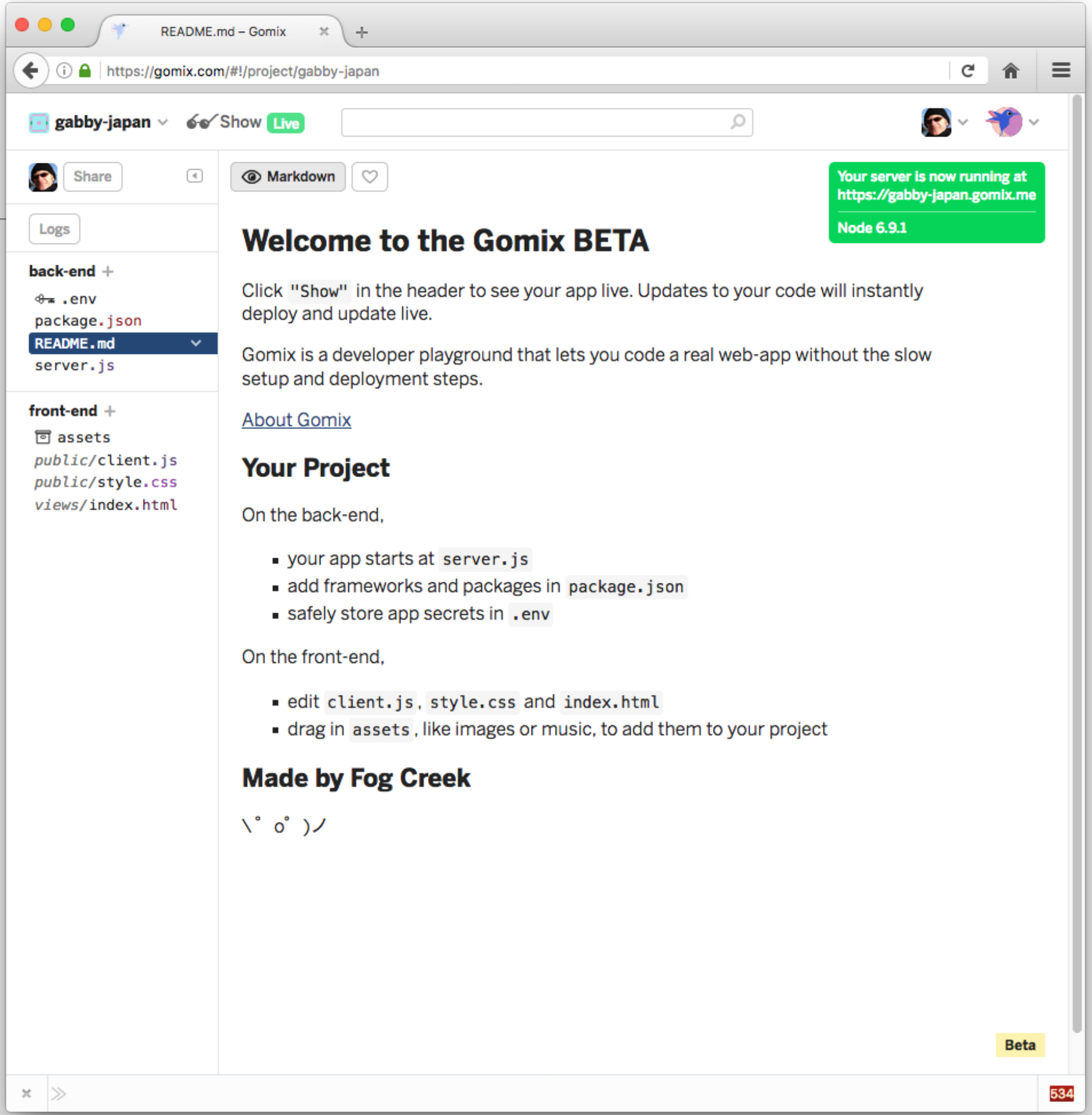
Gomix Tour

Prerequisite tools on your Workstation

- none!
- (apart from a browser + a github account)

- First screen is the “source” for a running, live web project

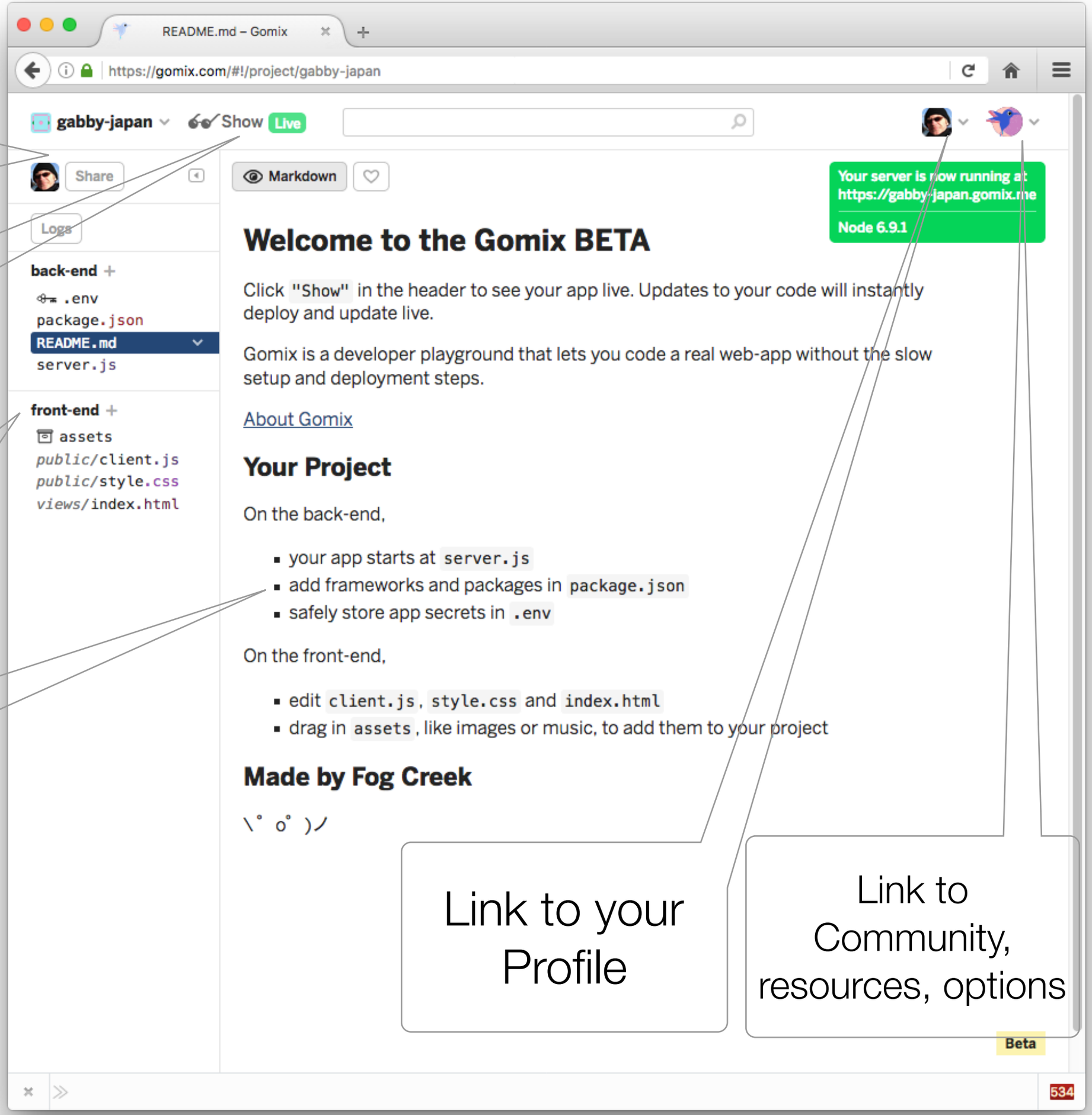


Project name (automatically generated)

Link to running app (to share)

Files in the project

Current File (editable)



Link to your Profile

Link to Community, resources, options

README.md – Gomix

https://gomix.com/#!/project/gabby-japan

gabby-japan · Show Live

Share

Logs

back-end +
- .env
- package.json
- README.md
- server.js

front-end +
- assets
- public/client.js
- public/style.css
- views/index.html

Markdown

Welcome to the ix Bl

Click "Show" in the header to see your app deploy and update live.

Gomix is a developer playground that lets y setup and deployment steps.

About Gomix

Your Project

On the back-end,

- your app starts at server.js
- add frameworks and packages in pacl
- safely store app secrets in .env

On the front-end,

- edit client.js, style.css and inde
- drag in assets, like images or music,

Made by Fog Creek

\˚ o˚)ノ

README.md – Gomix

Welcome to Gomix!

https://gabby-japan.gomix.me

A Dream of the Future

Oh hi,

Tell me your hopes and dreams:

Dreams!

Submit

- Find and count some sheep
- Climb a really tall mountain
- Wash the dishes

Remix this in Gomix

- Project is always running live (provided there are source errors)

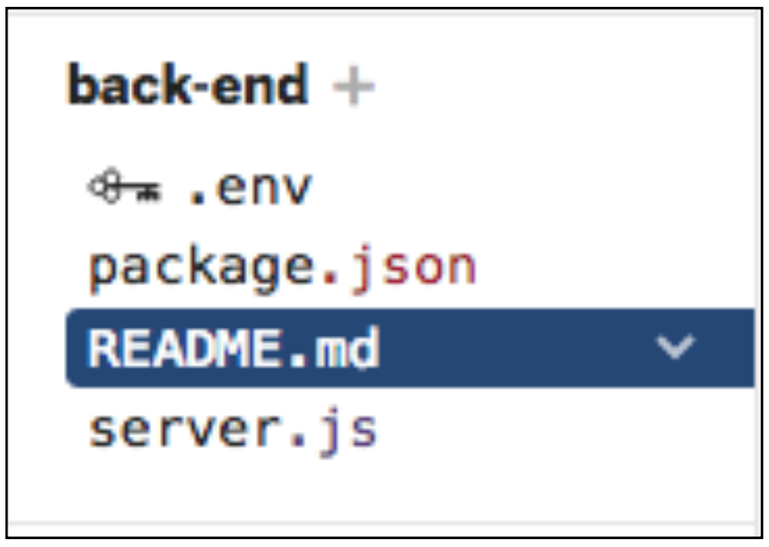
Project Structure

- Gomix projects not just web sites!
- They are web apps, divided into:
  - Back-end files
  - Front-end files

Front End

- Comparable to the web site you have been developing to date:
  - html files + stylesheets + images
- Templating also possible.
- Also, access to the server side is implicit.
- This means you can build apps that have behaviour + state (much more on this later)

Back end



- A new departure over web site dev 1.
- An application - written in javascript - and hosted in the cloud.
- Many types of application supported.
- We will focus on Javascript applications written using node.js
- This is the default toolkit for gomix - but other variants are planned.

The Starter App

index.html – Gomix

Eamonn

https://gomix.com/#!/project/actually-specialist

actually-specialist

Show Live

Share

Logs

back-end +

.env
package.json
README.md
server.js

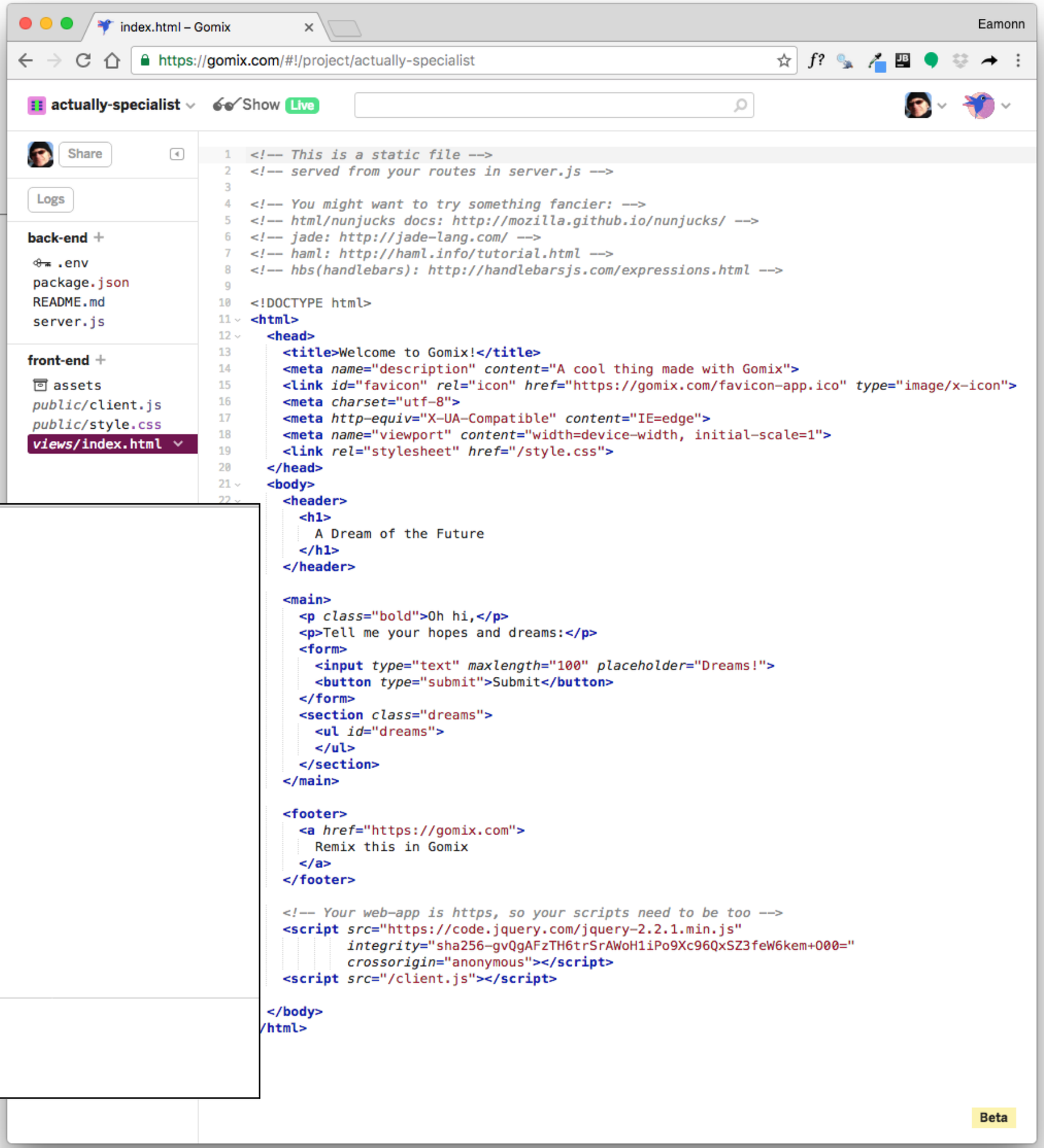
front-end +

assets
public/client.js
public/style.css
views/index.html

```
<!-- This is a static file -->
<!-- served from your routes in server.js -->

<!-- You might want to try something fancier: -->
<!-- html/nunjucks docs: http://mozilla.github.io/nunjucks/ -->
<!-- jade: http://jade-lang.com/ -->
<!-- haml: http://haml.info/tutorial.html -->
<!-- hbs(handlebars): http://handlebarsjs.com/expressions.html -->

<!DOCTYPE html>
<html>
  <head>
    <title>Welcome to Gomix!</title>
    <meta name="description" content="A cool thing made with Gomix">
    <link id="favicon" rel="icon" href="https://gomix.com/favicon-app.ico" type="image/x-icon">
    <meta charset="utf-8">
    <meta http-equiv="X-UA-Compatible" content="IE=edge">
    <meta name="viewport" content="width=device-width, initial-scale=1">
    <link rel="stylesheet" href="/style.css">
  </head>
  <body>
    <header>
      <h1>
        A Dream of the Future
      </h1>
    </header>

    <main>
      <p class="bold">Oh hi,</p>
      <p>Tell me your hopes and dreams:</p>
      <form>
        <input type="text" maxlength="100" placeholder="Dreams!">
        <button type="submit">Submit</button>
      </form>
      <section class="dreams">
        <ul id="dreams">
        </ul>
      </section>
    </main>

    <footer>
      <a href="https://gomix.com">
        Remix this in Gomix
      </a>
    </footer>

    <!-- Your web-app is https, so your scripts need to be too -->
    <script src="https://code.jquery.com/jquery-2.2.1.min.js"
            integrity="sha256-gvQgAFzTH6trSrAWoH1iPo9Xc96QxSZ3feW6kem+O00="
            crossorigin="anonymous"></script>
    <script src="/client.js"></script>

  </body>
</html>
```

Beta

The Starter App



A Dream of the Future

Oh hi,

Tell me your hopes and dreams:

Dreams!

Submit

- Find and count some sheep
- Climb a really tall mountain
- Wash the dishes

Remix this in Gomix

```
<body>
  <header>
    <h1>
      A Dream of the Future
    </h1>
  </header>

  <main>
    <p class="bold">Oh hi,</p>
    <p>Tell me your hopes and dreams:</p>
    <form>
      <input type="text" maxlength="100" placeholder="Dreams!">
      <button type="submit">Submit</button>
    </form>
    <section class="dreams">
      <ul id="dreams">
      </ul>
    </section>
  </main>

  <footer>
    <a href="https://gomix.com">
      Remix this in Gomix
    </a>
  </footer>
```

html

```html
<body>
  <header>
    <h1>
      A Dream of the Future
    </h1>
  </header>

  <main>
    <p class="bold">Oh hi,</p>
    <p>Tell me your hopes and dreams:</p>
    <form>
      <input type="text" maxlength="100" placeholder
      <button type="submit">Submit</button>
    </form>
    <section class="dreams">
      <ul id="dreams">
      </ul>
    </section>
  </main>

  <footer>
    <a href="https://gomix.com">
      Remix this in Gomix
    </a>
  </footer>
```

client side javascript

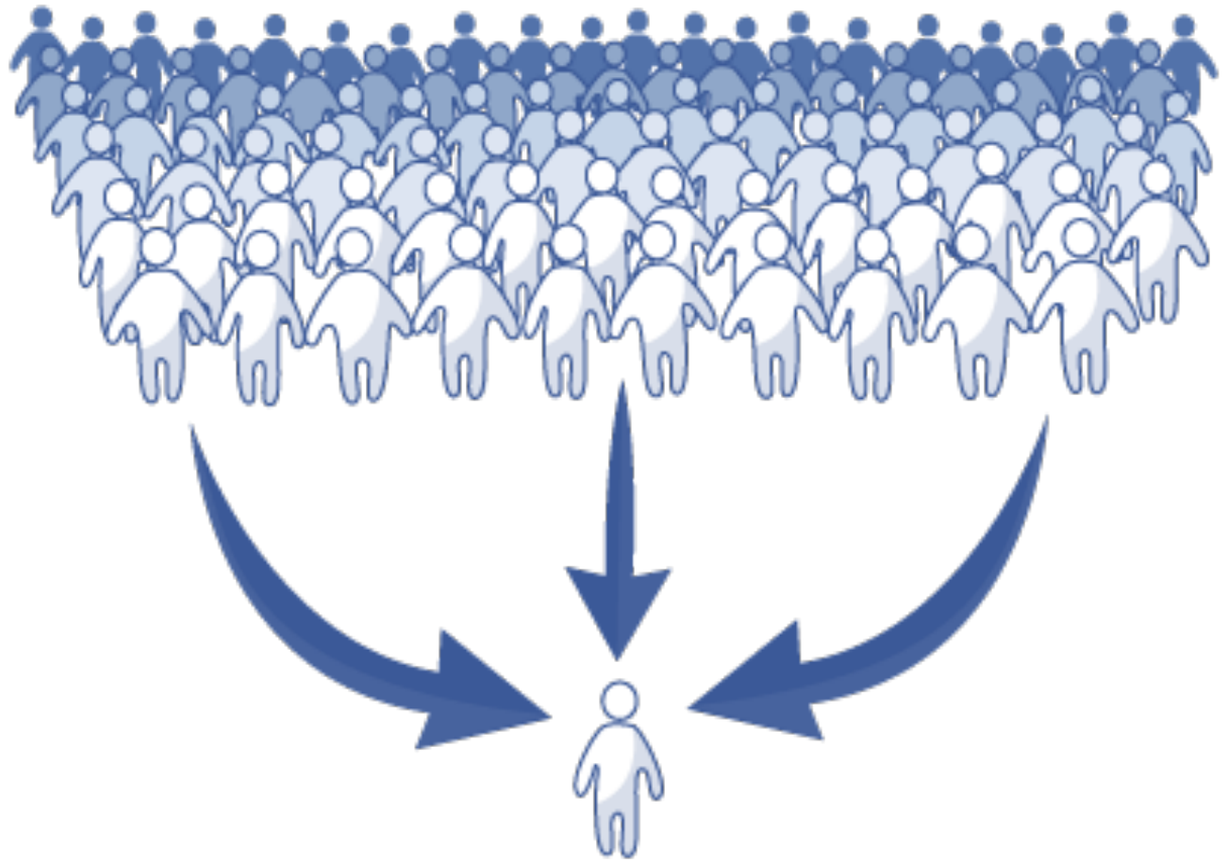
```javascript
// client-side js
// run by the browser each time your view template is loaded

// by default, you've got jQuery,
// add other scripts at the bottom of index.html

$(function() {
  console.log('hello world :o');

  $.get('/dreams', function(dreams) {
    dreams.forEach(function(dream) {
      $('<li></li>').text(dream).appendTo('ul#dreams');
    });
  });

  $('form').submit(function(event) {
    event.preventDefault();
    dream = $('input').val();
    $.post('/dreams?' + $.param({dream: dream}), function() {
      $('<li></li>').text(dream).appendTo('ul#dreams');
      $('input').val('');
      $('input').focus();
    });
  });

});
```

server side javascript

```
// server.js
// where your node app starts

// init project
var express = require('express');
var app = express();

// we've started you off with Express,
// but feel free to use whatever libs or frameworks you'd like through `package.json`.

// http://expressjs.com/en/starter/static-files.html
app.use(express.static('public'));

// http://expressjs.com/en/starter/basic-routing.html
app.get("/", function (request, response) {
  response.sendFile(__dirname + '/views/index.html');
});

app.get("/dreams", function (request, response) {
  response.send(dreams);
});

// could also use the POST body instead of query string: http://expressjs.com/en/api.html#req.body
app.post("/dreams", function (request, response) {
  dreams.push(request.query.dream);
  response.sendStatus(200);
});

// Simple in-memory store for now
var dreams = [
  "Find and count some sheep",
  "Climb a really tall mountain",
  "Wash the dishes"
];

// listen for requests :)
var listener = app.listen(process.env.PORT, function () {
  console.log('Your app is listening on port ' + listener.address().port);
});
```

- Client side javascript runs in each users browser

```
$('form').submit(function(event) {
  event.preventDefault();
  dream = $('input').val();
  $.post('/dreams?' + $.param({dream: dream}), function() {
    $('<li></li>').text(dream).appendTo('ul#dreams');
    $('input').val('');
    $('input').focus();
  });
});
```



```
// could also use the POST body instead of query string: http://expressjs.com/en/api.html#req.body
app.post("/dreams", function (request, response) {
  dreams.push(request.query.dream);
  response.sendStatus(200);
});
```

- A node runs the server side javascript. All browsers connected to this node

Skills for this Course

- Web App Development 1
    - Basic Javascript knowledge
    - Back end development in Javascript
- Front end javascript development is deferred for a future course

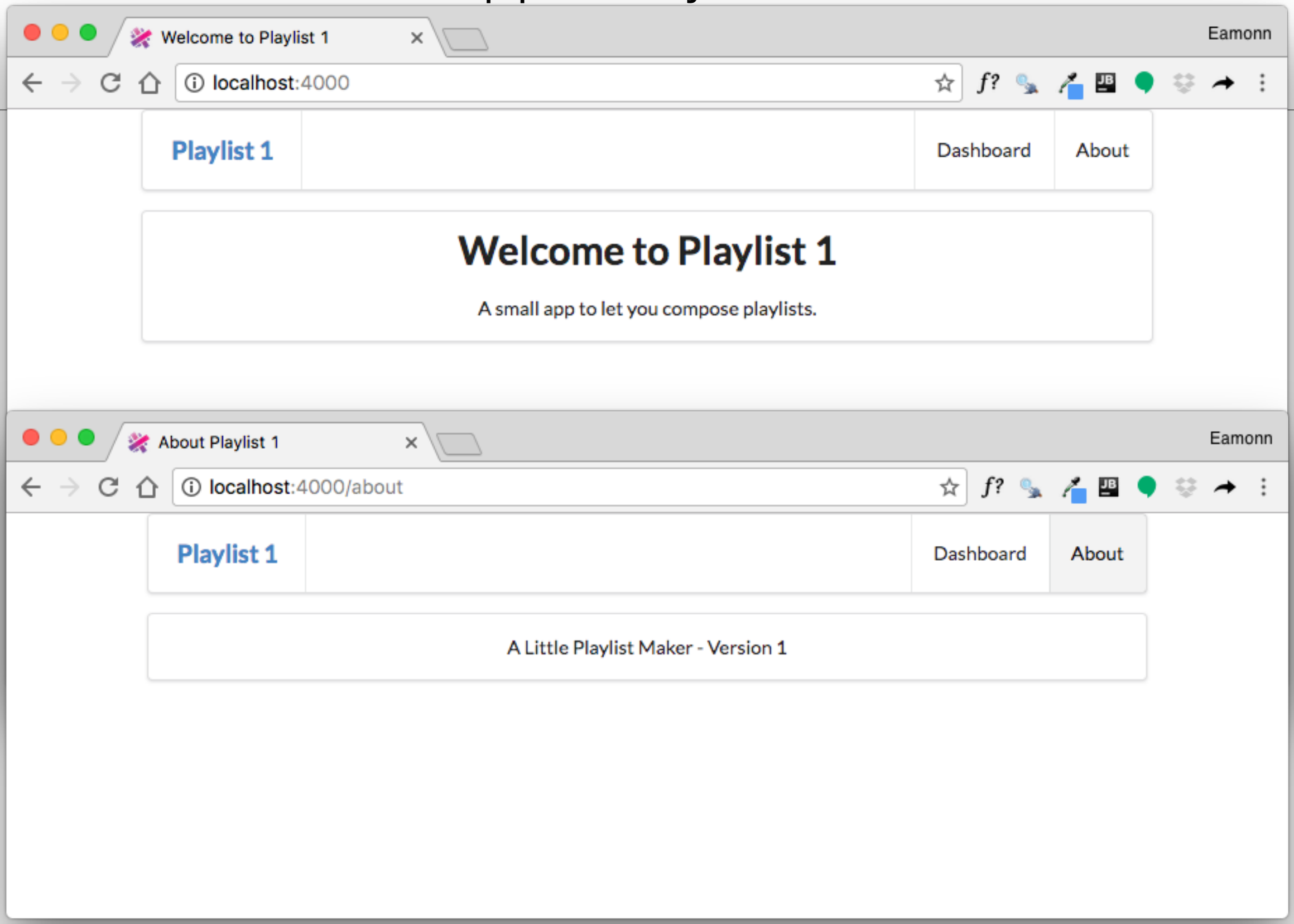
```
// server.js
// where your node app starts

// init project
var express = require('express');
var app = express();

// we've started you off with Express,
// but feel free to use whatever libs or frameworks you'd like through `package.json`.

// http://expressjs.com/en/starter/static-files.html
app.use(express.static('public'));

// http://expressjs.com/en/starter/basic-routing.html
app.get("/", function (request, response) {
  response.sendFile(__dirname + '/views/index.html');
});

app.get("/dreams", function (request, response) {
  response.send(dreams);
});

// could also use the POST body instead of query string: http://expressjs.com/en/api.html#req.body
app.post("/dreams", function (request, response) {
  dreams.push(request.query.dream);
  response.sendStatus(200);
});

// Simple in-memory store for now
var dreams = [
  "Find and count some sheep",
  "Climb a really tall mountain",
  "Wash the dishes"
];

// listen for requests :)
var listener = app.listen(process.env.PORT, function () {
  console.log('Your app is listening on port ' + listener.address().port);
});
```

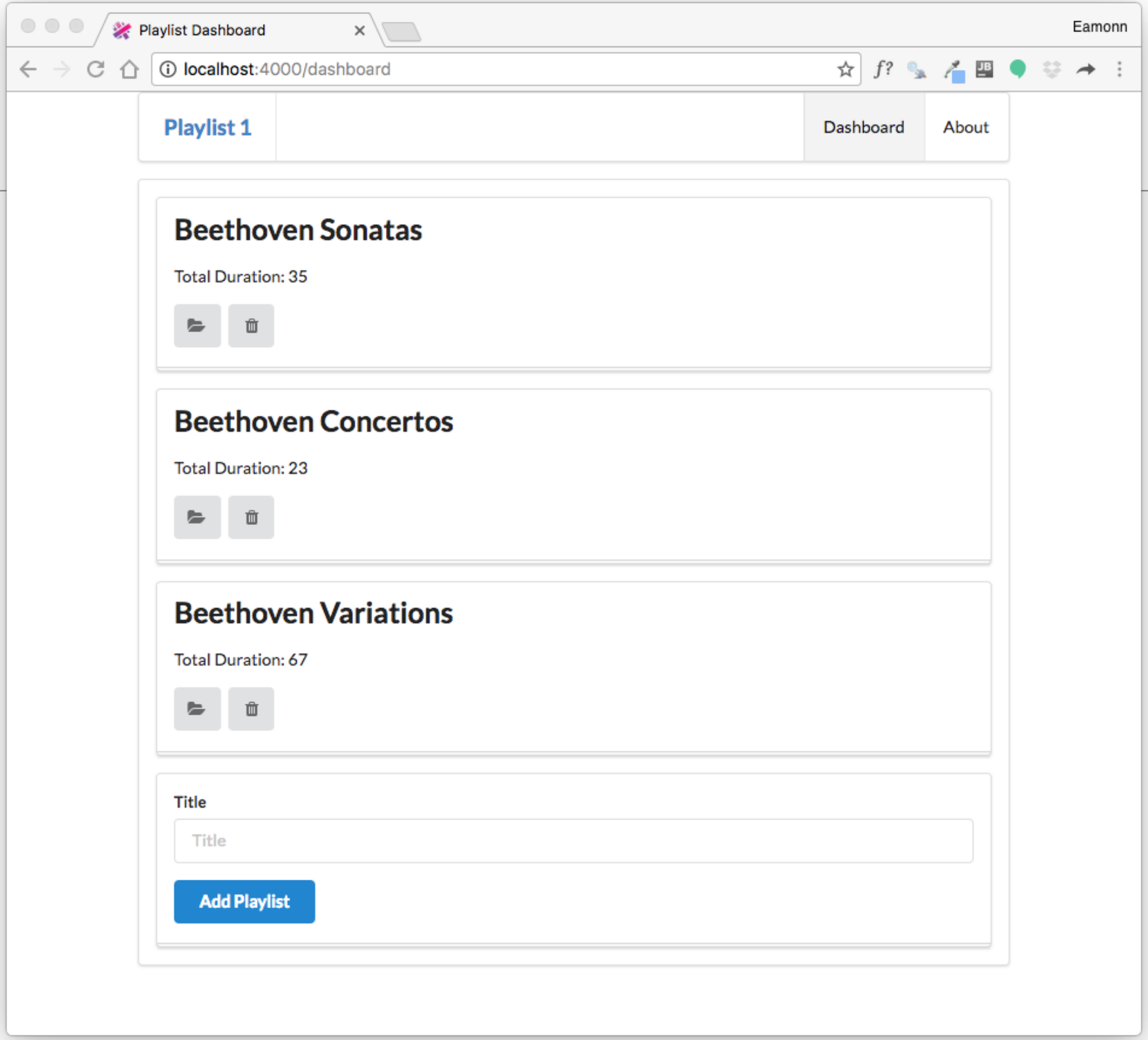
We will learn what all of this means.

- \+ how to build a fully featured web app including:
  - templating (like semester 1)
  - forms to submit information
  - How store information in models
  - create user accounts, and tie account to a each user

All of this requires intermediate level Javascript skills

A tour of our first app - Playlist




Playlist Dashboard
Eamonn
localhost:4000/dashboard
Playlist 1
Dashboard
About
Beethoven Sonatas
Total Duration: 35
Beethoven Concertos
Total Duration: 23
Beethoven Variations
Total Duration: 67
Title
Title
Add Playlist

Playlist Labs

- We will do three playlist labs
  - Playlist 1: simple rendering of static playlist
  - Playlist 2: render multiple playlists, ability to delete playlists
  - Playlist 3: ability to create playlists. Store playlists long term.

Playlist

Eamonn

localhost:4000/playlist/01

Playlist 1

Dashboard

About

Beethoven Sonatas

Song	Artist	
Piano Sonata No. 3	Beethoven	
Piano Sonata No. 7	Beethoven	
Piano Sonata No. 10	Beethoven	

Title

Title

Artist

Artist

Add Song