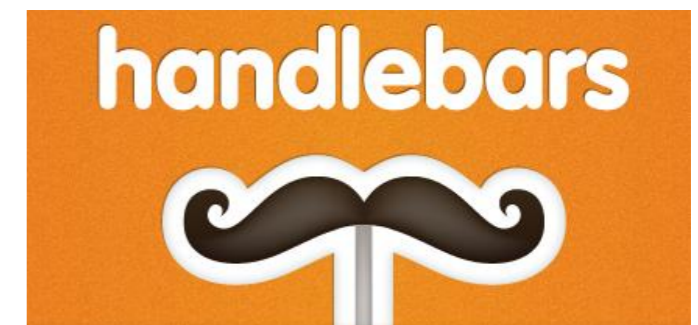


Web App Introduction



Starter App

- We will always start a new project by importing a template project from github.
- Playlist1 is a 'boilerplate' project providing foundation components for a starter application.
- Extend/repurpose this project to build interesting applications.

The screenshot shows the GitHub interface for the repository 'bmullally / playlist1'. At the top, there are navigation tabs for 'Code', 'Issues', 'Pull requests', 'Projects', 'Wiki', 'Insights', and 'Settings'. The repository has 1 commit, 1 branch, 0 releases, and 1 contributor. Below this, there are buttons for 'Branch: master', 'New pull request', 'Create new file', 'Upload files', 'Find file', and 'Clone or download'. The main content area displays a list of files and folders, all committed 7 days ago:

File/Folder	Commit	Time
controllers	playlist1	7 days ago
utils	playlist1	7 days ago
views	playlist1	7 days ago
.gitignore	playlist1	7 days ago
.jscsrc	playlist1	7 days ago
README.md	playlist1	7 days ago
package.json	playlist1	7 days ago
routes.js	playlist1	7 days ago
server.js	playlist1	7 days ago

Import from Github...

The screenshot shows a web browser window with the URL `https://glitch.com/edit/#!/quaint-dill?path=README.md:1:0`. The page title is "quaint-dill" and it has a "Live" indicator. On the left sidebar, under "Advanced Options", there are several buttons: "Make Private", "Open Console", "Import from Github", "Export to Github", "Download Project", and "Delete This Project". The "Import from Github" button is highlighted. A modal dialog box is open over the "Import from Github" button, titled "glitch.com says:". The dialog contains the following text: "Import from an existing GitHub Repo", "For best results repo size < 4MB, < 100 files, and no binary files", and "(NOTE: This will overwrite your current project)". Below the text is a text input field containing the repository name "bmullally/playlist1". At the bottom of the dialog are two buttons: "OK" and "Cancel".

glitch.com says:

Import from an existing GitHub Repo

For best results repo size < 4MB, < 100 files, and no binary files

(NOTE: This will overwrite your current project)

bmullally/playlist1

OK Cancel

Share

Markdown

Logs

+ New File

- assets
 - controllers/about.js
 - controllers/dashboard.js
 - controllers/start.js
 - utils/logger.js
 - views/layouts/main.hbs
 - views/partials/mainpanel.hbs
 - views/partials/menu.hbs
 - views/about.hbs
 - views/dashboard.hbs
 - views/start.hbs
- .env
- .gitignore
- .jscsrc
- README.md**
- package.json
- routes.js
- server.js

Welcome to the Gomix Template 1

A starter project for learning Gomix.

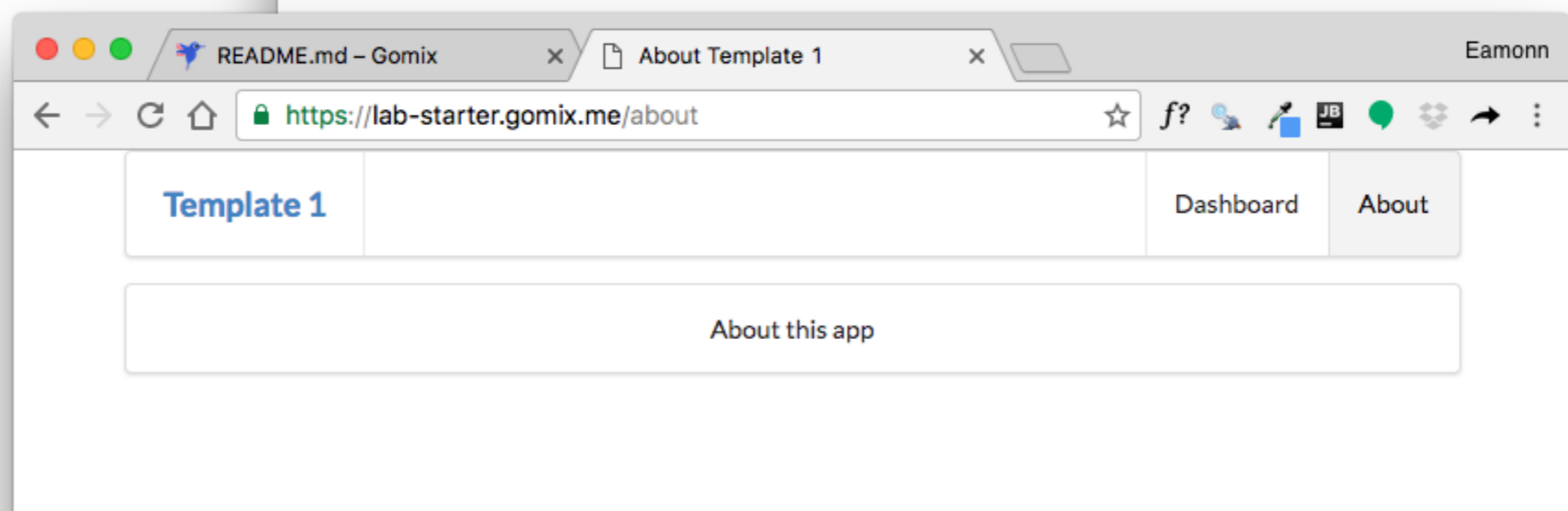
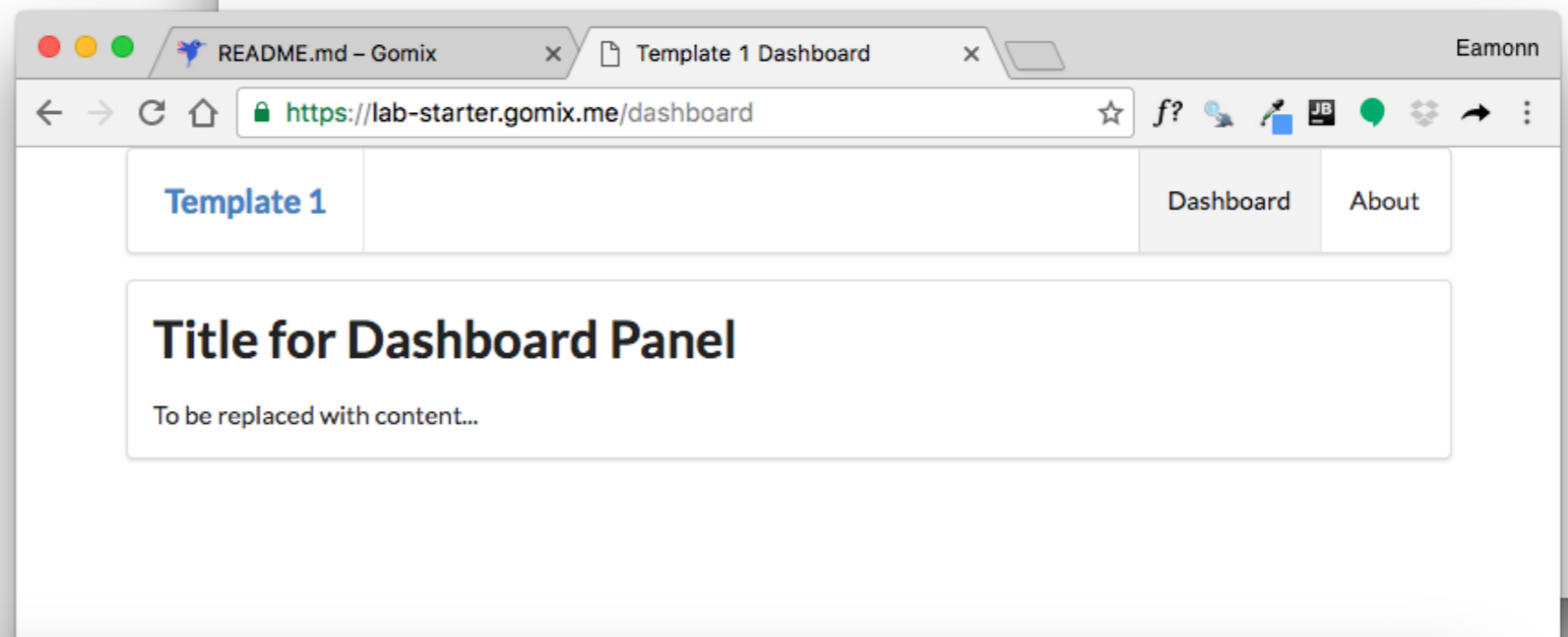
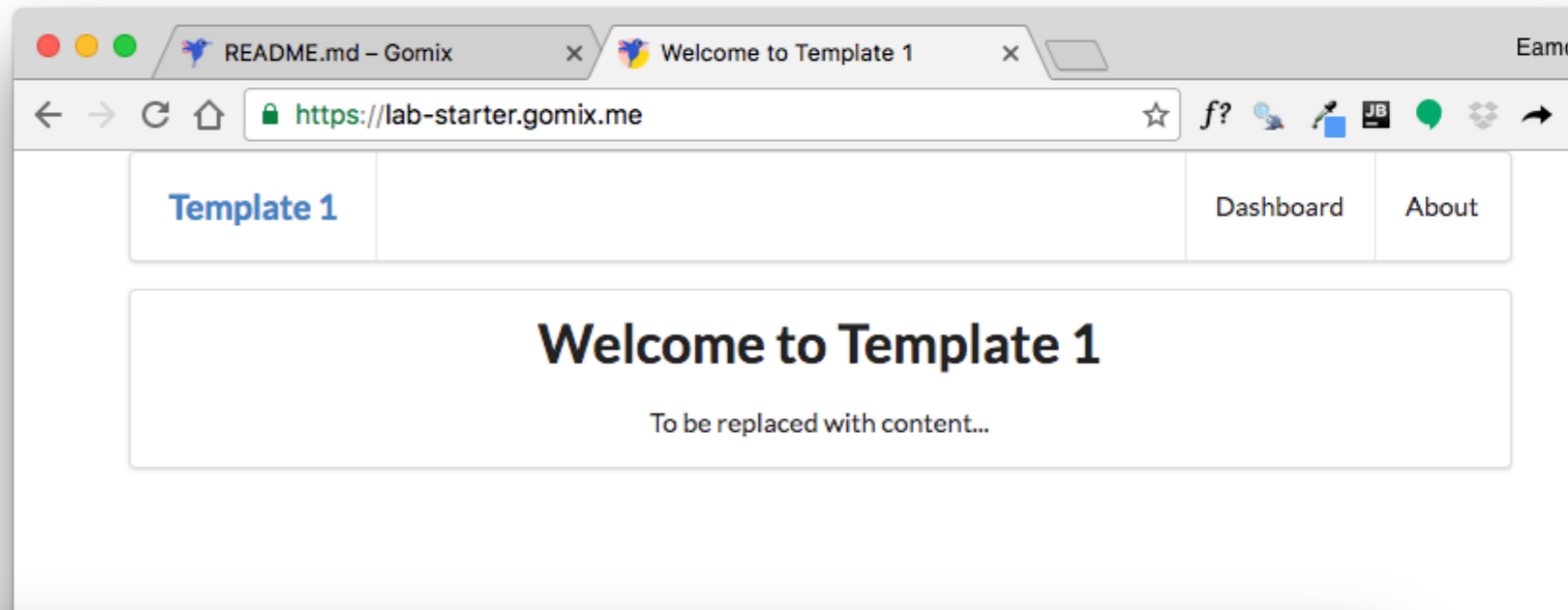
This is an Express.js project, designed to work well with the Gomix development environment. It includes basic express setup, templating, routing and session support.

Activity Log Clear

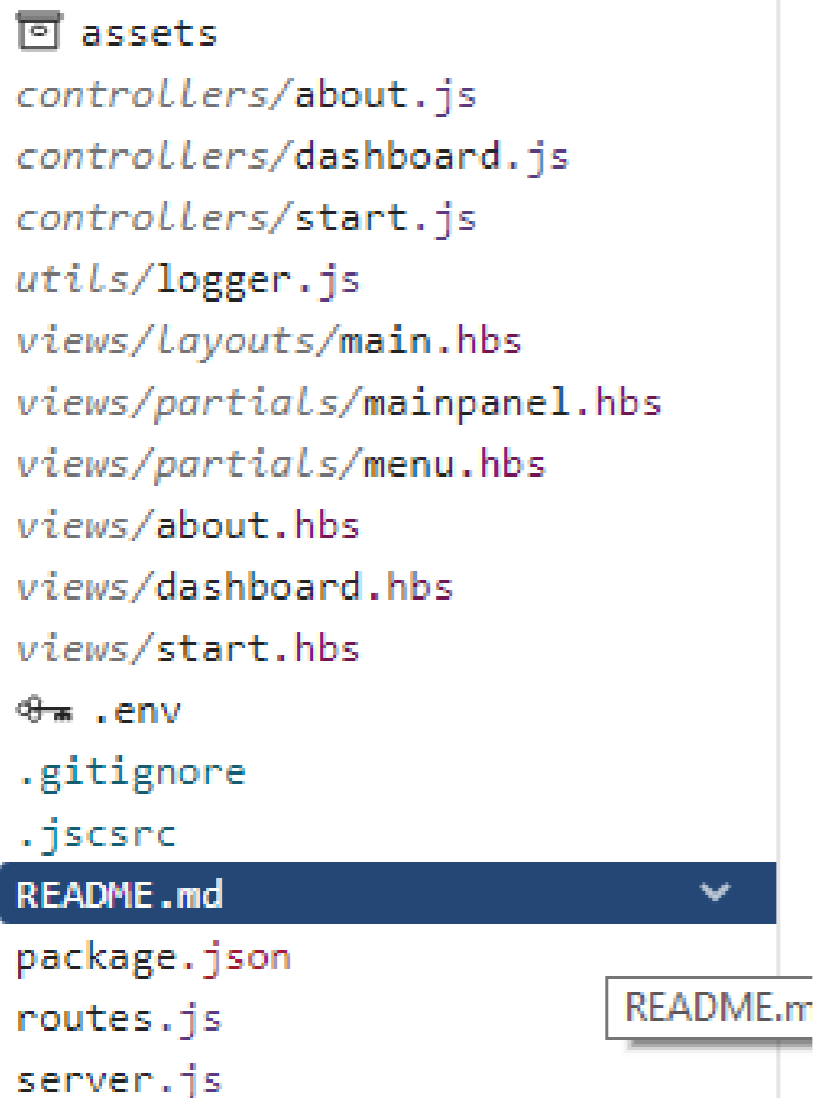
winston@2.3.1	2:29 PM
async@1.0.0	2:29 PM
colors@1.0.3	2:29 PM
cycle@1.0.3	2:29 PM
eyes@0.1.8	2:29 PM
isstream@0.1.2	2:29 PM
stack-trace@0.0.10	2:29 PM
npm WARN gomix-list@0.0.1 No license field.	

- The starter app has a menu + 3 'Views':

- Start
- Dashboard
- About



Application Structure



A screenshot of a file explorer showing the application structure. The files and folders are listed as follows:

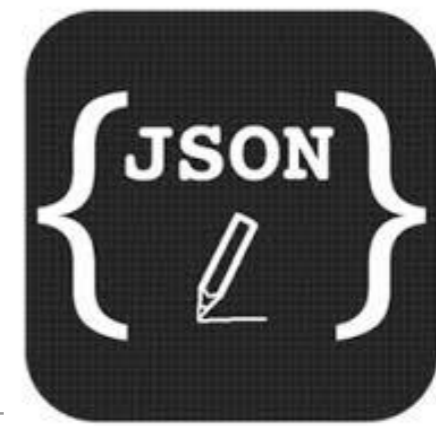
- assets
- controllers/about.js
- controllers/dashboard.js
- controllers/start.js
- utils/logger.js
- views/layouts/main.hbs
- views/partials/mainpanel.hbs
- views/partials/menu.hbs
- views/about.hbs
- views/dashboard.hbs
- views/start.hbs
- .env
- .gitignore
- .jscsrc
- README.md (highlighted)
- package.json
- routes.js
- server.js

A small tooltip labeled "README.m" is visible over the README.md file.

- App implements Routes + Model/View/Controller Architecture
- Back-end + Front-end collaborate to support structured, predictable application workflow

Back-end

JS



```
controllers/about.js  
controllers/dashboard.js  
controllers/start.js
```

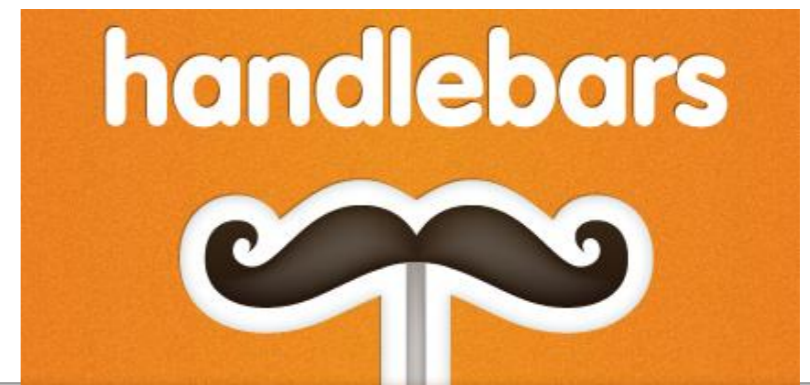
```
⌘ .env  
.gitignore  
.jscsrc
```

```
README.md
```

```
package.json  
routes.js  
server.js
```

- All written in Javascript + JSON
- Consists of:
 - **Server** - main entry point
 - **Routes** - supported urls
 - **Controllers** - objects to handle the routes
 - **Config** - .gitignore, .jscsrc, env, package.json, readme.md
- Will include **Models** later...

Front-end

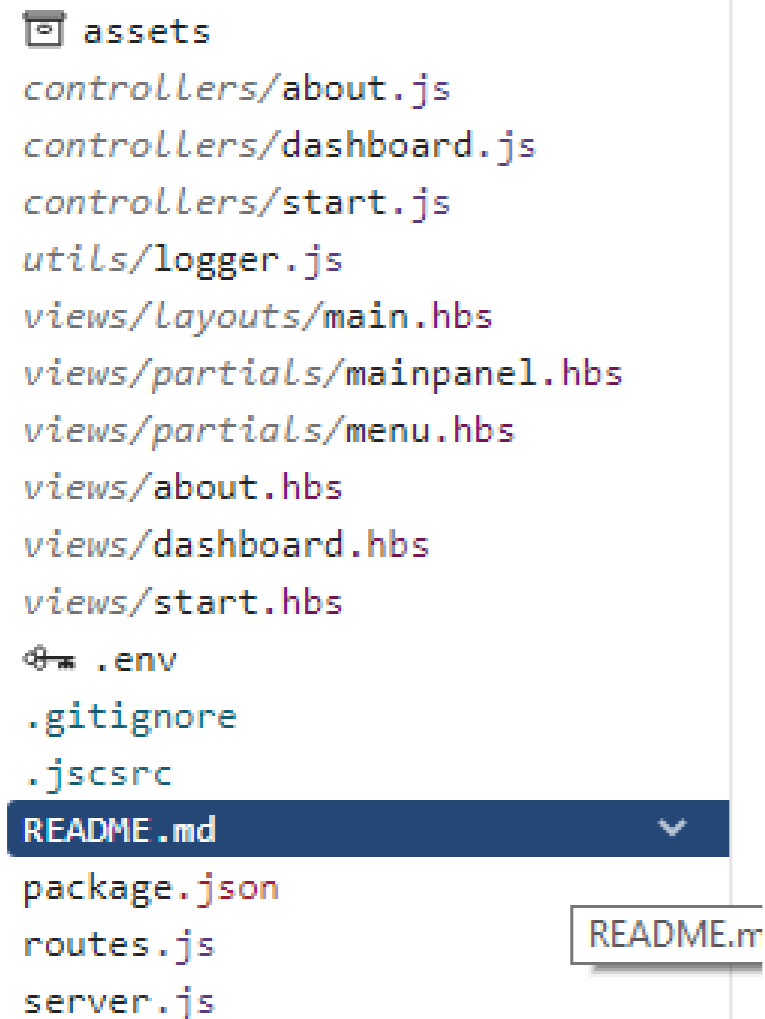


 `assets`

```
views/layouts/main.hbs
views/partials/mainpanel.hbs
views/partials/menu.hbs
views/about.hbs
views/dashboard.hbs
views/start.hbs
```

- All written in HTML + handlebars
- Handlebars: Templating language
- Similar to EJS, it supports:
 - **Layouts**
 - **Partials**
 - **Views**
- These are very similar to EJS equivalents

Back-end + Front-end



```
assets
controllers/about.js
controllers/dashboard.js
controllers/start.js
utils/logger.js
views/layouts/main.hbs
views/partials/mainpanel.hbs
views/partials/menu.hbs
views/about.hbs
views/dashboard.hbs
views/start.hbs
.env
.gitignore
.jscsrc
README.md
package.json
routes.js
server.js
```

README.m

- Building a new app:
 - Extend, modify, replace and create new routes + controllers
 - Rewrite handlebars templates to support new features, data and user experience.