CCChat Documentation

Leonid Immel and Oliver Schneider

22. März 2018

Inhaltsverzeichnis

1 Up next

1. routes
2. login

2 Node Modules (Serverside)

Module	Benefit
express	Web application framework that includes several useful features for web applications.
passport	Authentication middleware for session management.
socket.io	Realtime event-based bidirectional communication.
winston	Logger.
body-parser	Handles http post requests. Extracts body of http to req.body.
serve-favicon	Handles the requests from browsers for the favicon and caches the favicon.
handlebars	Builds semantic templates
mongoose	For MongoDB modeling. Saves all app data.
redis	For Redis Connection. Saves the cookies.
mocha	Testframework
chai	Helper for testframework

3 Vendor Libraries (Clientside)

Library	Benefit
bootstrap	Responsive website programming for mobile-first implementation.
jQuery	Easier handling of javascript

4 Project Structure

- app
  - public (all static served files)
    - js
    - css
    - html
    - libs
  - services
    - «servicename».route.js (Route)
    - «servicename».views.js (Handlebars view)
    - «servicename».spec.js (Test)
    - «servicename».module.js (Module/Logic)
  - middleware (processes incoming requests before they are passed to the routes (loginauthentication))
  - helpers (functionality and code shared in the whole app (regex- or renderoperations))
  - tests (applicationtests)

5 Documentation

modules The functionality of modules is described within a comment on top of the js-file.

functions Functions with complex implementations or unclear names (should never happen!) need a commentation
Simpler functions are not commented to produce more readable code

variables Variables require clear and good naming, never comment a variable unless absolutely necessary.
You should think a littel longer and find a better name.

6 Formatting

We use the Eclipse JS Formatter standart for our code.

7 SSL

7.1 Create certificate

```
openssl genrsa -out client-key.pem 2048
openssl req -new -key client-key.pem -out client.csr
openssl x509 -req -in client.csr -signkey client-key.pem -out client-cert.pem
```

7.2 Allow Certificate in browser for Dev

7.2.1 Chromium / Chrome 64

Option 1:

1. Go to Settings. Hit the „Advanced“ button on bottom of the page
2. Click the „Manage certificates“ button in the „Privacy and security“ Tab
3. Click the „Authorities“ Tab
4. Click on „Import“

5. Navigate to the cert file within your app source code folder.
6. Click open and allow what you want to allow

Option 2:

1. Open the side explitetly with the url „https://localhost:3000“
2. Click the advanced Button
3. Allow the certificate
4. Continue with Finish (see a little further down)
5. follow option 2 until point 3.
6. find your certificate in the list of authorities.
7. click expand on the certificate
8. click the 3 dots that appear under the certificate
9. click „edit“
10. allow what you want to allow

7.2.2 Firefox 59

1. Open the side explitetly with the url „https://localhost:3000“
2. Click the „Advanced“ Button
3. Click „Add Exception“
4. Click „Confirm Security Exception“

8 Logger

9 Error Handling

10 Useful websites

- Simple chat with socket.io

- Handlebars
- Combine Bootstrap and ReactJS
- Mongoose
- Example of secure chat in production
- Add a SSL-Cert in Chrome