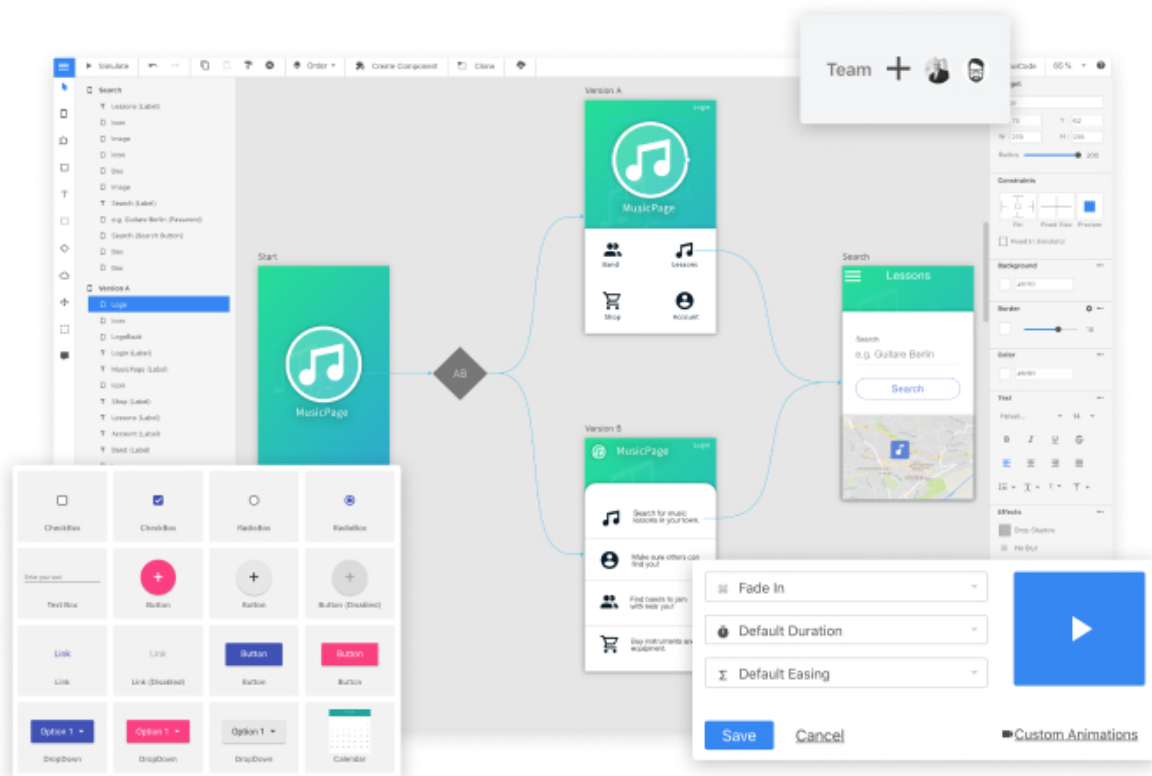


Quant-UX - Prototype, Test and Learn

Quant UX is a research, usability and prototyping tool to quickly test your designs and get data driven insights. This repo contains the front end. You can find a working demo at <https://quant-ux.com/#/>



Development setup

```
1 npm install
```

Compiles and hot-reloads for development

```
1 npm run serve
```

Compiles and minifies for production

```
1 npm run build
```

Run your unit tests

```
1 npm run test:unit
```

Lints and fixes files

```
1 npm run lint
```

Installation

The easiest way to get your own installation up and running is using the prebuild Docker images by Brian McGonagill. You can find the repo and instructions at <https://github.com/bmcgonag/quant-ux-docker/>

Manual Installation

Quant-UX has two components. A front-end (this package) and a backend (qux-java). The front-end needs Node.js (> 12) installed. The backend needs a Mongo DB, a Mail Server (SMTP) and Java (> 1.8). The front-end comes with it's own mini web server, which also include a proxy that redirects all request to the correct backend.

Docker

The easiest way to get your own Quant-UX installation running is using the Docker images.

- 1) Create a docker compose file (`docker-compose.yml`) and set the environment variables.

```
1 version: '3'
2
3 services:
4   mongo:
5     restart: always
6     container_name: quant-ux-mongo
```

```

7     image: mongo
8     volumes:
9         - ./data:/data/db          # path for the data to be stored and kept
                                   on your host machine is on the left side of the ":"
10    qux-fe:
11        restart: always
12        container_name: quant-ux-frontend
13        image: klausenschaefersinho/quant-ux
14        environment:
15            - QUX_PROXY_URL=http://quant-ux-backend:8080      # this is the
                                   path the front end uses to talk tot he backend
16            - QUX_AUTH=qux
17            - QUX_KEYCLOAK_REALM=
18            - QUX_KEYCLOAK_CLIENT=
19            - QUX_KEYCLOAK_URL=
20            - QUX_WS_URL=ws://127.0.0.1:8086      # change to where the
                                   websocket server is deployed for external access
21        links:
22            - mongo
23            - qux-be
24        ports:
25            - 8082:8082      # change the left side port if your host
                                   machine already has 8082 in use
26        depends_on:
27            - qux-be
28    qux-be:
29        restart: always
30        container_name: quant-ux-backend
31        image: klausenschaefersinho/quant-ux-backend
32        volumes:
33            - ./quant-ux-data:/app-data
34        environment:
35            - QUX_HTTP_HOST=http://quant-ux-frontend:8082      # this is the URL
                                   included in the mails, e.g. password resets
36            - QUX_HTTP_PORT=8080      # This is the port the backend will use
37            - QUX_MONGO_DB_NAME=quantux      # the database / collection name in
                                   mongodb
38            - QUX_MONGO_TABLE_PREFIX=quantux      # table / document prefix in
                                   mongodb
39            - QUX_MONGO_CONNECTION_STRING=mongodb://quant-ux-mongo:27017      #
                                   this assumes your mongodb container will be called "quant-ux-
                                   mongo" in the docker-compose file
40            - QUX_MAIL_USER=mail_admin@example.com      # this should be
                                   your smtp email user
41            - QUX_MAIL_PASSWORD=sTr0ngPa55w0Rd      # this should be your
                                   smtp email password
42            - QUX_MAIL_HOST=mail.example.com      # this should be your
                                   smtp host address
43            - QUX_JWT_PASSWORD=some-long-string-of-mix-case-chars-and-nums
                                   # you should change this to a real JWT secret
44            - QUX_IMAGE_FOLDER_USER=/app-data/qux-images      # this folder

```

```

45         should mapped in the volume
46     - QUX_IMAGE_FOLDER_APPS=/app-data/qux-image-apps          # this
47       folder should mapped in the volume
48     - TZ=America/Chicago          # change to your timezone
49     - QUX_AUTH_SERVICE=qux
50     - QUX_KEYCLOAK_SERVER= # just the keycloak host & port
51     - QUX_KEYCLOAK_REALM=
52     - QUX_USER_ALLOW_SIGNUP=true # set the false to not allow users
53       to signup
54     - QUX_USER_ALLOWED_DOMAINS=* # comma separated list of domains, e
55       .g. 'my-server.com' or '*' for all
56 depends_on:
57     - mongo
58 qux-ws:
59     restart: always
60     container_name: quant-ux-websocket-server
61     image: klausenschaefersinho/quant-ux-websocket
62     environment:
63         - QUX_SERVER=http://quant-ux-backend:8080/
64         - QUX_SERVER_PORT=8086
65     ports:
66         - 8086:8086
67     links:
68         - qux-be
69     depends_on:
70         - qux-be

```

Make sure to update `QUX_JWT_PASSWORD` the ENV variable to make sure your installation is secure. Update `QUX_HTTP_HOST`, `QUX_MAIL_USER`, `QUX_MAIL_PASSWORD` and `QUX_MAIL_HOST` to sure correct mail handling

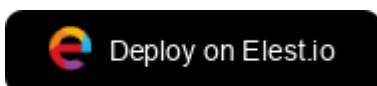
2) Start the containers with the following command

```
1 docker compose up
```

One-Click deployment

Elestio

You can deploy an instance of Quant UX with few clicks and minimal configuration on cloud service provider of your choice.



Kubernetes

You can find a kubernetes configuration here <https://github.com/engmsilva/quant-ux-k8s/tree/master/k8s>

Backend

- Install Mongo DB (> 4.4)
- Install Java (1.8)
- Checkout the backend

```
1 git clone https://github.com/KlausSchaefer/qux-java.git
```

- This contains already a compiled version of the backend in the release folder
- Edit the matc.conf file to setup the correct mongo and mails server details. More details can be found here: <https://github.com/KlausSchaefer/qux-java>
- Start the server, or install as a service in Linux.

```
1 java -jar release/matc.jar -Xmx2g -conf matc.conf -instances 1
```

Front-end

- Install Node.js (> 12)
- Clone repo

```
1 git clone https://github.com/KlausSchaefer/quant-ux.git
```

- Install all dependencies:

```
1 npm install
```

- Build

```
1 npm run build
```

Config front-end

- Set the proxy server url as an ENV variable

```
1 export QUX_PROXY_URL=https://your.quant-ux.server.com // backend host
2
3 export QUX_WS_URL= wss.quant-ux.server.com // web socket server
```

- Start

```
1 node server/start.js
```

Reverse Proxy

Now you should have a running system. It is not secure yet. The best is to put both behind a NGINX reverse proxy, which handles SSL.

- <https://www.scaleway.com/en/docs/tutorials/nginx-reverse-proxy/>

You can use <https://letsencrypt.org/> to create SSL certificates