

PRÉSENTATION

Système de messagerie web

Filipe Fortunato & Jonathan Zaehringer

TECHNOLOGIE

FRONTEND

- vuejs
- vuetify
- axios
- moment



BACKEND



EXPRESS.JS

BASE DE DONNÉES

SQLITE 3

SCHEMA

SCHEMA Users

```
{  
  id: INTEGER PRIMARY,  
  username: VARCHAR UNIQUE,  
  first_name: VARCHAR,  
  last_name: VARCHAR,  
  digest_password: VARCHAR,  
  active: INTEGER,  
  deleted: INTEGER,  
}
```

Mot de passe hashé avec sha256

SCHEMA Messages

```
{  
  id: INTEGER PRIMARY,  
  from: INTEGER FOREIGN(users.id),  
  to: INTEGER FOREIGN(users.id),  
  subject: VARCHAR,  
  message: VARCHAR,  
  read: INTEGER,  
  timestamp: INTEGER,  
}
```

AUTHENTICATION

COOKIE

```
{  
  "user_id": Number,  
  "level": Number,  
  "iv": String,  
  "encryptedData": String,  
}
```

- encryptedData contient la session

SESSION

Authoritative pour le backend. Stocke :

- user_id
- level
- validity

AES256 CBC

Server key

MIDDLEWARE

Appelé à chaque requête

NAVIGATION

BACKEND Expressjs

API RESTful GET/POST/PUT/DELETE

ROUTER

- Router primitif liant une ressource → une fonction.
- Définition de route `RESTful`
- Accepte l'injection d'argument dans la ressource

FRONTEND *vuejs*

Framework *single page*

Sans rechargement

ROUTER

Système de routage fictif dans `router.js`

```
{  
  path: '/login',  
  name: 'login',  
  component: () => import('./views/Login.vue'),  
},
```

APPEL À L'API

Données envoyées/reçues de manière asynchrone

Grace à `Axios`

DÉMO