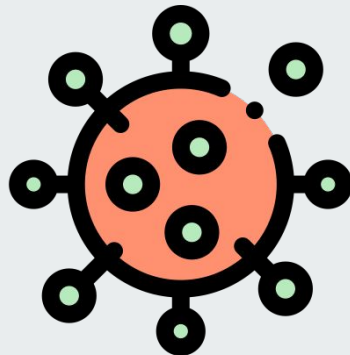




Info



COVID19

Barthélemy Passin-Cauneau

Théo Fafet

Anthony Choquard

Florent Robert

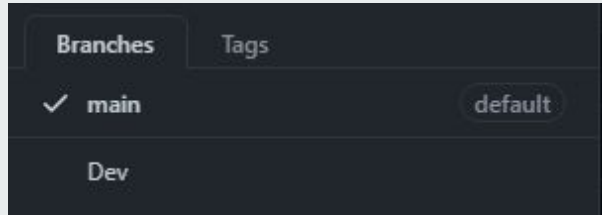


Plan

- Gestion de projet
- Démonstration
- Architecture serveur
- Architecture client
 - Tableau
 - Graphe
 - Map
- Difficultés rencontrées
- Conclusion



Gestion de projet



build : create front react,



Gestion de projet

- **Semaine 1** : Setup du back
- **Semaine 2** : Recherche de librairies pour le front
- **Semaine 3** : Implémentation des composants dans le front
- **Semaine 4** : Peaufinage & Fix



Gestion de projet

- **Anthony Choquard** : Component carte
- **Théo Fafet** : Architecture back, routes & filtres
- **Barthélemy Passin-Cauneau** : MongoDB, routes , Architecture front, component tableau
- **Robert Florent** : Component graphe



Démonstration



Server

App

```
const port = 8080;  
const hostname = 'localhost';  
  
const server = app.listen(port, hostname, () => {  
  console.log(`Server is listening on http://${hostname}:${port}`)  
})
```

MongoDB Router

Régions Router



Server

```
/**
 * Connexion to mongoDB :
 * dont forget to create a mongodb.config at the root project and write in it the url
 */
const url = fs.readFileSync("./mongodb.config", "utf-8")
const dbName = 'covidDB';
let db

MongoClient.connect(url, { useUnifiedTopology: true }, function(err, client) {
  console.log("Connected to mongoDB server");
  db = client.db(dbName);
});
```

[Données hospitalières relatives à l'épidémie de COVID-19 - data.gouv.fr](#)

▼ covidDB

heb-dep

heb-fra

heb-reg

quot-dep

quot-fra

quot-reg

```
_id: ObjectId("602a7c128886236c788b2ea2")
dep: 1
week: "2020-S21"
P: 2
cl_age90: 19
pop: 84665
```




Server

```
const paths = ['/heb/dep', '/heb/fra', '/heb/reg', '/quot/dep', '/quot/fra', '/quot/reg']
```

```
db.collection(path.replace("/", "").replace("/", "-")).find(filtering(req.query)).toArray()
```

```
const compareKeywords = [
  { key: ">=", value: "$gte" },
  { key: "<=", value: "$lte" },
  { key: ">", value: "$gt" },
  { key: "<", value: "$lt" },
  { key: "!", value: "$ne" }
];
```

▼ covidDB

heb-dep

heb-fra

heb-reg

quot-dep

quot-fra

quot-reg



Server

```
const paths = ['/heb/dep', '/heb/fra', '/heb/reg', '/quot/dep', '/quot/fra', '/quot/reg']
```

```
db.collection(path.replace("/", "").replace("/", "-")).find(filtering(req.query)).toArray()
```

```
const compareKeywords = [
  { key: ">=", value: "$gte" },
  { key: "<=", value: "$lte" },
  { key: ">", value: "$gt" },
  { key: "<", value: "$lt" },
  { key: "!", value: "$ne" }
];
```

/regions

```
[{
  "idf": 11,
  "cvl": 24,
  "bfc": 27,
  "nor": 28,
  "hdf": 32,
  "ges": 44,
  "pdl": 52,
  "bre": 53,
  "naq": 75,
  "occ": 76,
  "ara": 84,
  "pac": 93,
  "cor": 94
}]
```



Client

Composant fetch

```
▼ lib
  JS FetchFranceLiveGlobalData.js
  JS FetchLocationUserDep.js
  JS FetchServerInputData.js
  JS FetchServerMapData.js
```

Composant principal

JS App.js

Composant d'affichage des données

```
▼ components
  JS CasesNumberGraph.js
  JS DarkModeButton.js
  🔗 DarkModeButton.scss
  JS DepMap.js
  JS GraphColumn.js
  JS GraphCurve.js
  JS Grid.js
  🔗 Grid.scss
  JS Home.js
  JS Legend.js
  🔗 Legend.scss
  JS Map.js
  🔗 Map.scss
  JS RegMap.js
```



Client

Localisation

```
const API = "https://api-adresse.data.gouv.fr/reverse/";

export const FetchLocationUserDep = new Promise((resolve, reject) => {
  navigator.geolocation.getCurrentPosition(position => {
    fetch(API + "?lat=" + position.coords.latitude + '&lon=' + position.coords.longitude)
      .then(location => location.json())
```

Localisation

Client



Darkmode



```
export const DarkModeButton = ({ mode, onChange }) => (  
  <div className="DarkModeButton">  
    <img src={` ${mode ? Moon : Sun}`} ></img>  
    <Switch className="Switch"  
      checked={mode}  
      onChange={onChange}  
    />  
  </div>  
);
```

```
window.matchMedia("(prefers-color-scheme: dark)").matches)
```

```
const API = "https://api-adresse.data.gouv.fr/reverse/";  
  
export const FetchLocationUserDep = new Promise((resolve, reject) => {  
  navigator.geolocation.getCurrentPosition(position => {  
    fetch(API + "?lat=" + position.coords.latitude + '&lon=' + position.coords.longitude)  
      .then(location => location.json())
```

Localisation

Client



Darkmode



```
export const DarkModeButton = ({ mode, onChange }) => (  
  <div className="DarkModeButton">  
    <img src={` ${mode ? Moon : Sun}`} ></img>  
    <Switch className="Switch"  
      checked={mode}  
      onChange={onChange}  
    />  
  </div>  
);
```

```
window.matchMedia("(prefers-color-scheme: dark)").matches)
```

```
const API = "https://api-adresse.data.gouv.fr/reverse/";  
  
export const FetchLocationUserDep = new Promise((resolve, reject) => {  
  navigator.geolocation.getCurrentPosition(position => {  
    fetch(API + "?lat=" + position.coords.latitude + '&lon=' + position.coords.longitude)  
      .then(location => location.json())
```

Responsive

```
@media screen and (max-width: 1000px) {  
  position: initial;  
  
  table, td {  
    margin-left: auto;  
    margin-right: auto;  
  }  
}
```



Tableau

Données en France datant du 27 février 2021

Guéris	257 258
Décès	61 450
Hospitalisés	24 989



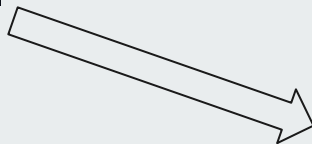
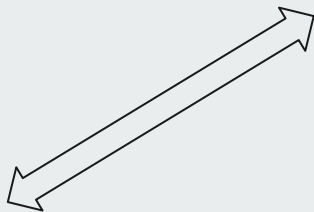
Tableau

App

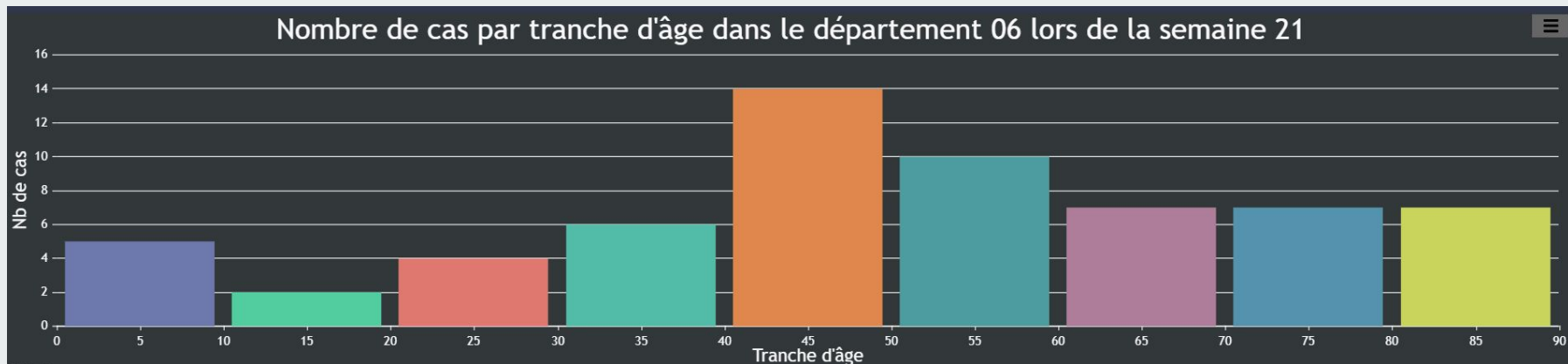
```
useEffect(() => {  
  FetchFranceLiveGlobalData.then(data => setRealtimeData(data));  
});
```

Composant
FetchFranceLiveGlobalData

Composant
Grid



Graphe





Graphe



Graphe

CasesNumberGraph.js

```
let path = "http://localhost:8080/covid_data/heb/dep?dep="+departement.defaultDepartement.value;  
FetchServerInputData(path).then(data => {updateCurveGraph(data);})
```

Sélectionnez un département

06 | ▾

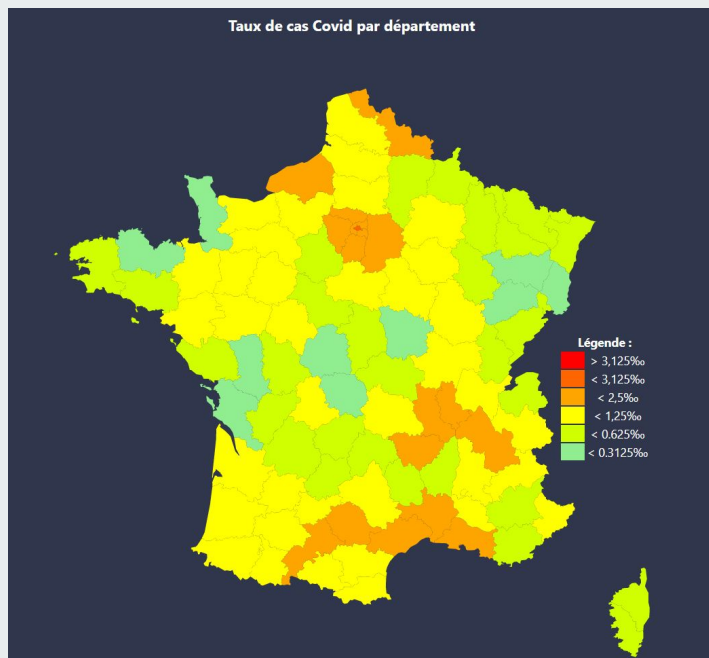
Backend
Filtres



Composant
FetchServerInputData

Composant
Graphe

Carte



```
[{  
  "idf": 11,  
  "cvl": 24,  
  "bfc": 27,  
  "nor": 28,  
  "hdf": 32,  
  "ges": 44,  
  "pdl": 52,  
  "bre": 53,  
  "naq": 75,  
  "occ": 76,  
  "ara": 84,  
  "pac": 93,  
  "cor": 94  
}]
```



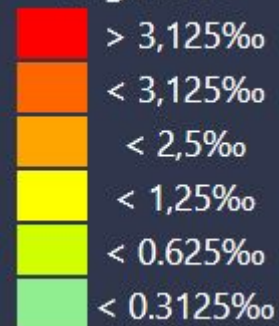
Carte



Carte

Départements ☒ Régions

Légende :



Semaine du 30 Septembre



A horizontal timeline slider with a series of small dots and a larger blue dot indicating the current position.



Difficultés rencontrées

```
export const Grid = ({data, mode}) => {  
  if (data.length > 0) {  
    return (  

```



Difficultés rencontrées

```
const Grid = lazy(  
  () => new Promise((resolve, reject) =>  
    FetchFranceLiveGlobalData.then(data => {  
      setRealtimeData(data);  
      resolve(import("../components/Grid"));  
    })  
  )  
);
```

```
<Suspense fallback={<div className="loadGrid"><p className="loadingText">Accès aux données en direct en cours...</p><div className="loader"></div></div>>  
  <Grid data={realtimeData} mode={displayMode} />  
</Suspense>
```




Difficultés rencontrées

```
export const Grid = ({data, mode}) => {  
  if (data.length > 0) {  
    return (  

```



```
const Grid = ({data, mode}) => {  
  return (  

```



Difficultés rencontrées

Recherche de librairie map



Conclusion



Merci de votre attention