

# Société Lepetit

Projet station météorologique

# Le problème

- Quitter le modèle actuel de solution sur le Cloud pour une solution locale.
- Diminuer les coûts d'hébergement et production

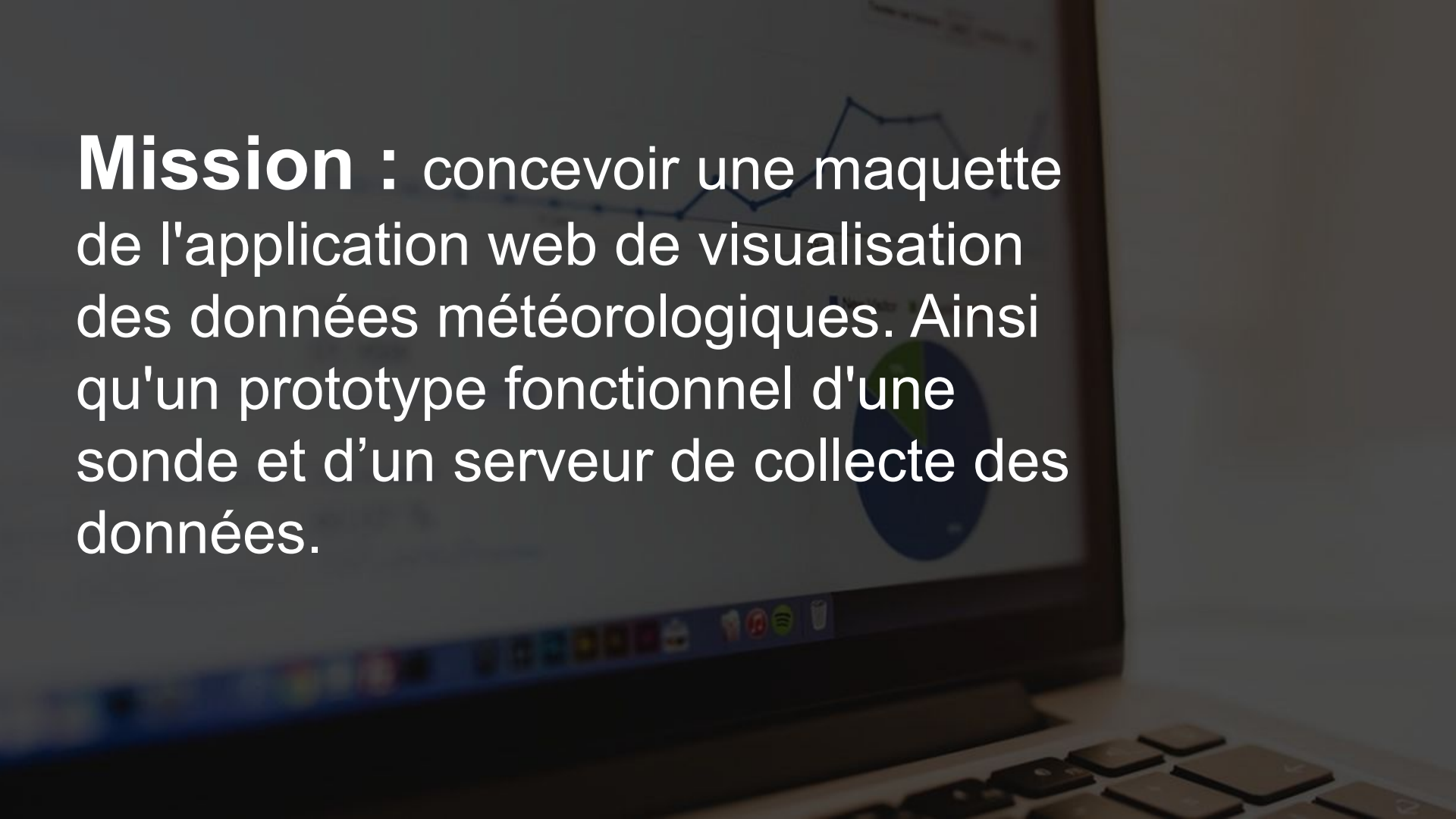


A laptop screen is shown in the background, displaying a line graph with a blue line and a pie chart with a blue and green segment. The text 'La solution' is overlaid on the screen. The line graph has a date '19 av.' visible. The pie chart has a legend with 'New Visitor' and 'Returning Visitor'.

# La solution

Mettre en place une station  
météorologique en local  
dont les informations  
s'affichent sur un site  
sécurisé

=> Diminution de coût plus  
dépendant d'hébergeur

The background image shows a laptop screen. On the screen, there is a line graph with a blue line and a globe. The text is overlaid on the left side of the screen.

**Mission** : concevoir une maquette de l'application web de visualisation des données météorologiques. Ainsi qu'un prototype fonctionnel d'une sonde et d'un serveur de collecte des données.

# Grandes étapes du projet



# Début du projet

## Étude du cahier des charges

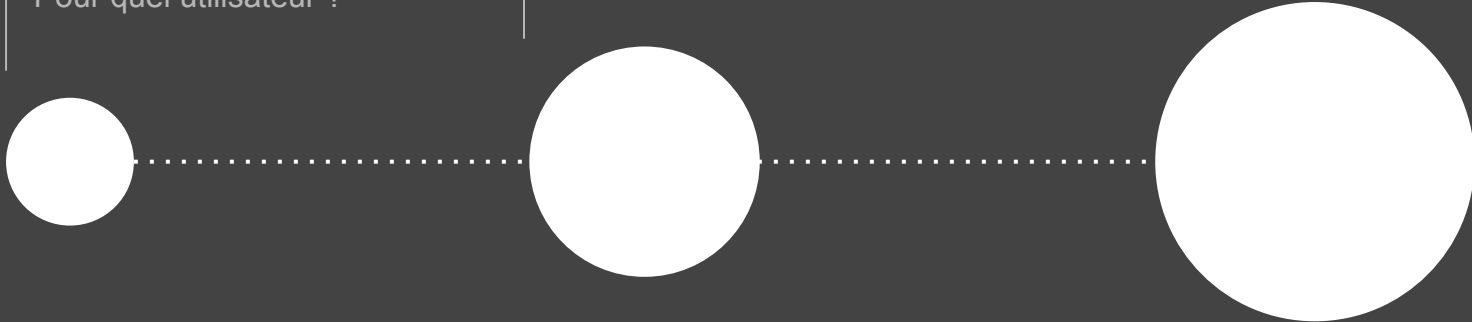
Quelle est la demande ?  
Pour quel utilisateur ?

## Trello et priorisation des tâches

Délimitation des objectifs.

## Utilisation de GitHub

Versioning, travail collaboratif.



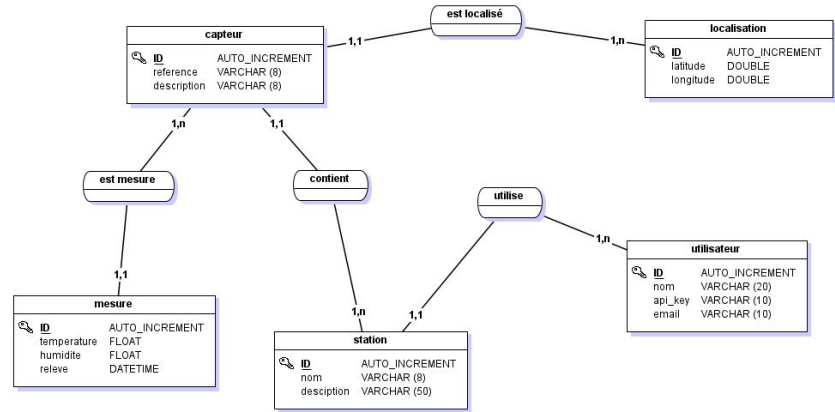
# Mise en place du back

1

Serveur local Apache sous Debian

2

Élaboration du MCD et base de données mySQL



# Conception de l'API REST



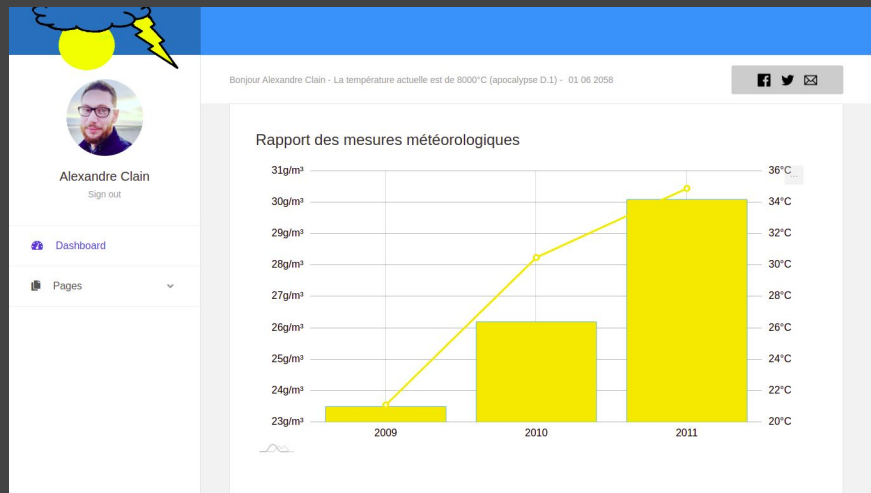


# Conception du front

1

## Adaptation du template

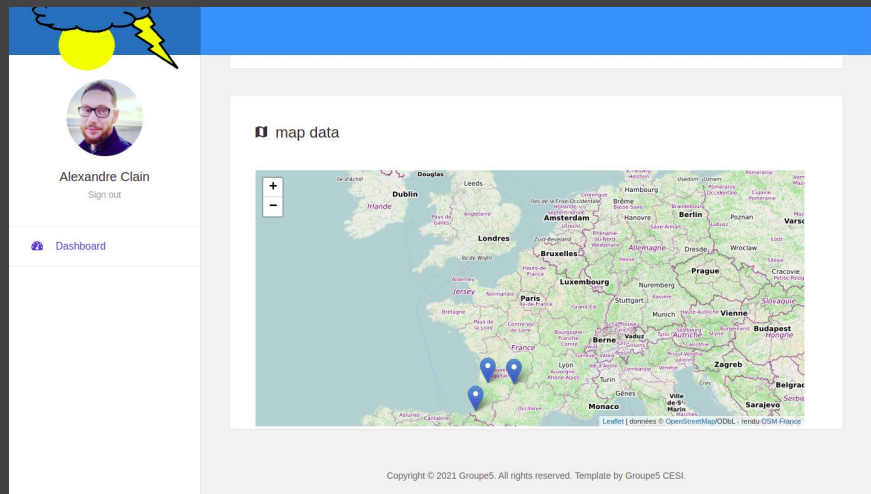
En HTML, CSS et PHP



2

## intégration du graphique et de la carte

AmCharts et OpenStreetMap  
en JavaScript et PHP



# Démonstration

# Problèmes rencontrés

1

Distanciel et  
manque de matériel

2

API en C++

```
void Database::generate_hash( AUser* who )
{
    std::string tmp_s;
    static const char alphanum[] =
        "0123456789"
        "ABCDEFGHIJKLMNOPQRSTUVWXYZ"
        "abcdefghijklmnopqrstuvwxyz"; // array of char to take (62)

    srand( (unsigned)time( NULL ) ); // setrand to the current timestamp. making random really random

    tmp_s.reserve( 32 ); // we reserve 32 char

    for ( int i = 0; i < 32; ++i )
        tmp_s += alphanum[rand() % (sizeof( alphanum ) - 1)]; // remove 1 cause null terminator in c*/]; // select random char

    std::string formatted_string;
    #if _MSC_VER >= 2017051 // vs studio 16.10 c++20 update
        formatted_string = std::format( "UPDATE 'utilisateur' SET 'api_key' = '{}'' WHERE 'utilisateur'.id = {};", tmp_s.c_str(), who->id ); //c++20 be like
    #else
        char* buffer = new char[256]; // create a new buffer
        sprintf( buffer, "UPDATE 'utilisateur' SET 'api_key' = '%s' WHERE 'utilisateur'.id = %i;", tmp_s.c_str(), who->id );
        formatted_string = buffer;
        delete[] buffer;
    #endif

    sql::Statement* stmt; // statement ptr
    stmt = get_connection()->createStatement(); // init statement
    stmt->executeUpdate( formatted_string ); // execute update (not query since it return value and if no value error)

    stmt->close(); // close statement
    delete stmt; // delete dynamically allocated class

    who->token = tmp_s; // Important! set manually the token or it will bug cause cache is not updated
    tmp_s.clear(); // clear the std::string
```

## Points forts

## Points faibles

## Points de vigilance

## Améliorations

Faible coût des composants.

Accessibilité de l'application.

Performances de l'application

Authentification utilisateur.

Ergonomie de l'application.

Sécurité.

Accessibilité de l'application pour les handicaps

Sécurité de l'application

Test en Réel avec le Raspberry

# Conclusion

## ***Solution proposée***

*API fonctionnelle, qui assure la connection entre la base de données et l'interface utilisateur.*

*Station météo à moindre coût*

*Hébergement local*

Nous espérons avoir participé à la prospérité de l'entreprise Lepetit et espérons que vous referez appel à nos services pour vos futurs projets.

An aerial photograph of New York City at dusk. The Empire State Building is the central focus, its top illuminated with red and green lights. The city's dense skyline of skyscrapers is visible, with many windows glowing from interior lights. The Hudson River and the New York Harbor are visible in the background under a dark, cloudy sky.

Merci :)