

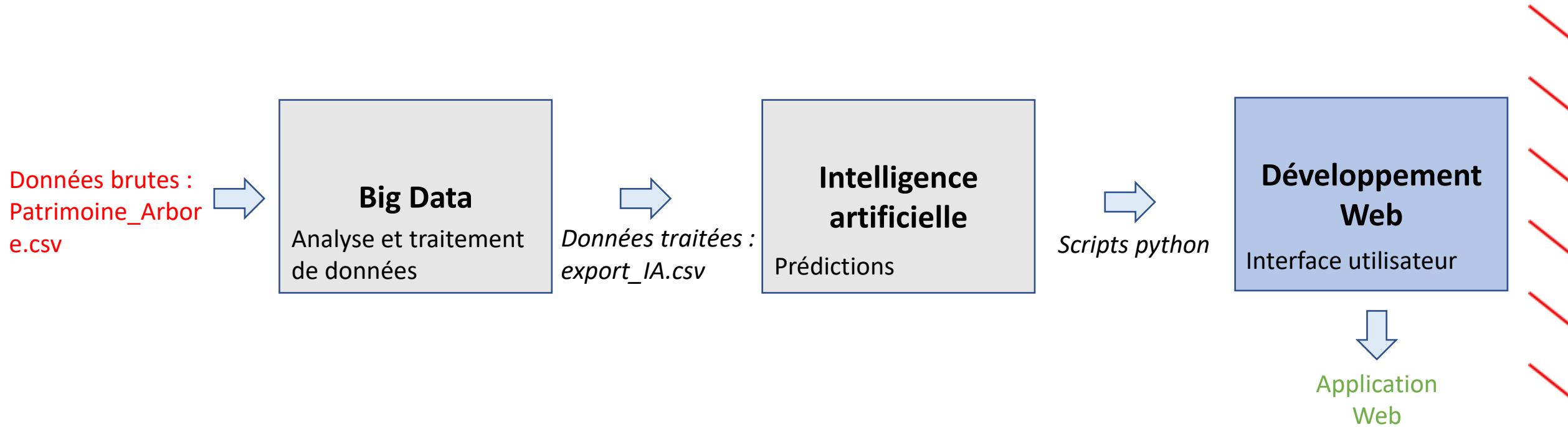
Projet Année 3 Big Data/IA/Web

Partie Développement Web



Contexte du projet

Déroulement du projet



Objectif

Concevoir et développer une application de gestion du patrimoine arboré : preuve de concept pour la ville de Saint-Quentin (Aisne)

Approfondir les compétences acquises dans les modules ***Big Data, Intelligence Artificielle, Développement Web et Base de Données*** à travers une application complète de traitements et de visualisation de données.

Objectifs de la partie Développement Web :

- Programmation web coté client (*front-end*) :
 - Créer une maquette visuelle d'un site web
 - Programmer les éléments de la maquette visuelle en *HTML*
 - Programmer le style de la maquette visuelle en *CSS*
 - Modifier le comportement de la page web en *JavaScript*
 - Manipuler *AJAX*
- Programmation web coté serveur (*back-end*) :
 - Créer un code *PHP* qui encapsule les requêtes permettant d'interagir avec la base de données
 - Traiter les réponses des requêtes en *PHP* et envoyer des réponses au client

Cahier des charges

Processus de développement

Analyse :

- Maquette du site web (*FIGMA, MockFlow*)
- Charte graphique
- Modèle Conceptuel de Donnée (*MCD*)
- Requêtes client-serveur associées aux pages présentées

Conception et développement :

- *HTML*
- *CSS*
- *JavaScript*
- *PHP*
- *SQL*

Test

Recette fonctionnelle

Cahier des charges

Technologies à utiliser

Partie *front-end* :

- Exclusivement avec les technologies *HTML*, *JavaScript* et *CSS*



- Les bibliothèques *Plotly* et *Mapbox*, et uniquement celles-ci, peuvent être utilisées



plotly



mapbox

Partie *back-end* :

- Exclusivement en *PHP*



- SGBD* de type *MySQL*



- Communications avec le *font-end* avec *AJAX*



- Échange de données avec le *font-end* en *JSON*



Cahier des charges

Technologies à utiliser

Mise en page :

Il faut que la mise en page du *front* prenne en compte un en-tête et un pied de page commun à toutes les pages.

Accès :

L'accès au site web se fera via un serveur web hébergé sur `projets.isen-ouest.fr`

L'application web doit être :

- Ergonomique : facilité d'utilisation, homogénéité des informations, respect des normes utilisées pour le Web
- Évolutive : possibilité de rajouter de nouvelles fonctionnalités

Le code doit être :

- Correctement architecturé
- Réexploitable : code lisible, code bien commenté, noms de variables / fonctions explicites
- Séparé en plusieurs fichiers (par fonctionnalités)

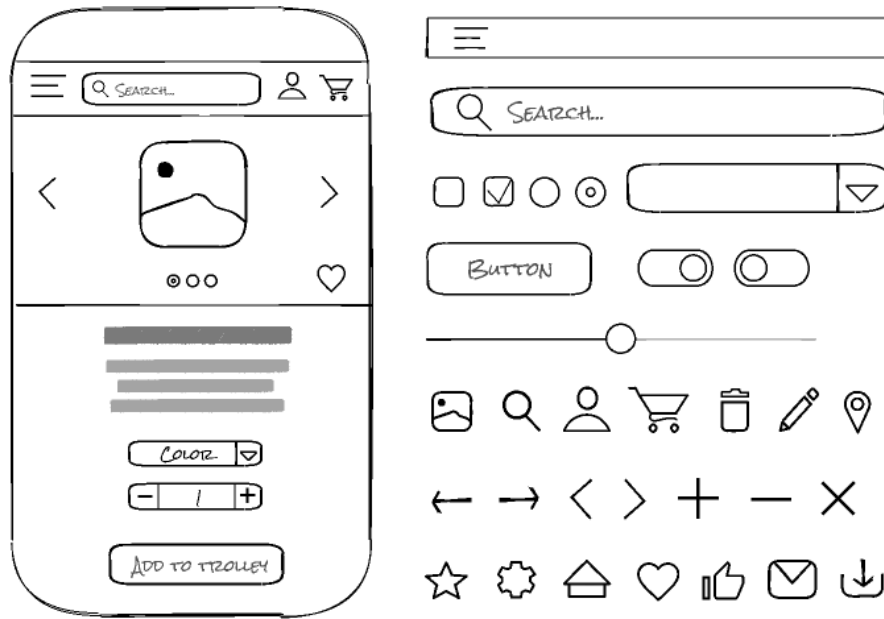
Partie 1 : Analyse

Analyse : Maquette de l'application web

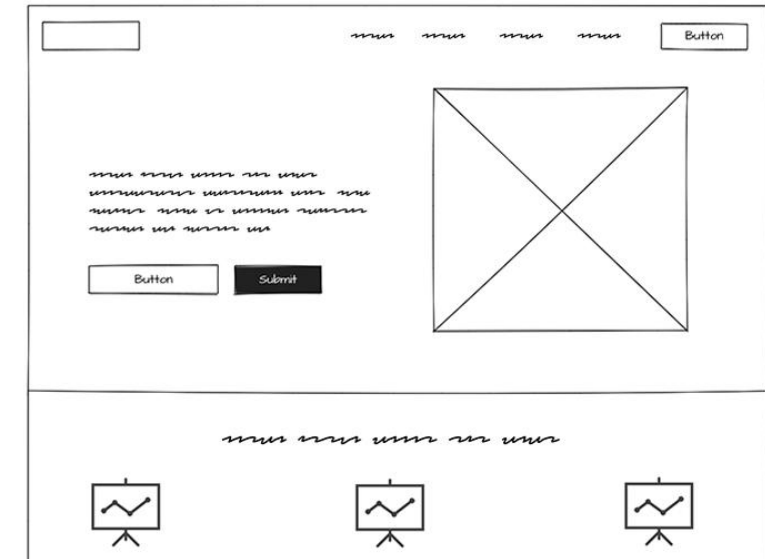
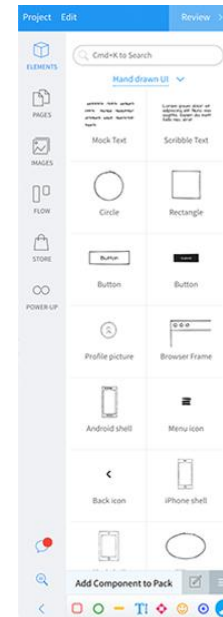
Figma, MockFlow

Préparation du squelette de votre application web :

Réalisation de la maquette en ligne avec *Figma* ou *MockFlow* :



Figma Handmade Wireframe Kit



MockFlow Wireframe

→ **Attendu : la maquette des différentes pages de votre application web en *PDF***

Analyse : Charte graphique

Choix des couleurs principales

Définition de la charte graphique de l'application web :

Choix des éléments de style :

- Couleurs
- Ombres
- Police de caractères

→ **Attendu : un document *PDF* présentant la charte graphique de votre application web**

Définition des ombres









Les caractéristiques des ombres sont les mêmes pour toutes les zones, à savoir :

Décalage en X : 5
Décalage en Y : 5
Flou : 10
Diffusion : 5
Intensité : 40%
Couleur : Noir



Figure 3 : Exemple d'utilisation des ombres

Caractéristiques des couleurs utilisées

	HEX : 12255B		HEX : FFEAEA
	HEX : 3A6BB7		HEX : FF3D3D
	HEX : 99AEE9		HEX : C3BDBD
	HEX : DDE0FB		HEX : 6F6767

Dans l'interface

Pour toute l'interface la police utilisée est **Assistant** avec différentes caractéristiques d'épaisseurs :

Light :

Bonjour Jeanine !

Regular :

Identifiant

Semi-Bold :

Se connecter Vous avez bu aujourd'hui :
CIVILITE

Date de naissance : 02/04/1944
Age : 77 ans Les champs indiqués

Bold :

Madame PICHON Jeanine 1.1L d'eau

INFORMATIONS UTILISATEUR

Dans les documents

GRAND TITRE Glacial Indifference, Bold

Sous-titres Open Sans, Bold

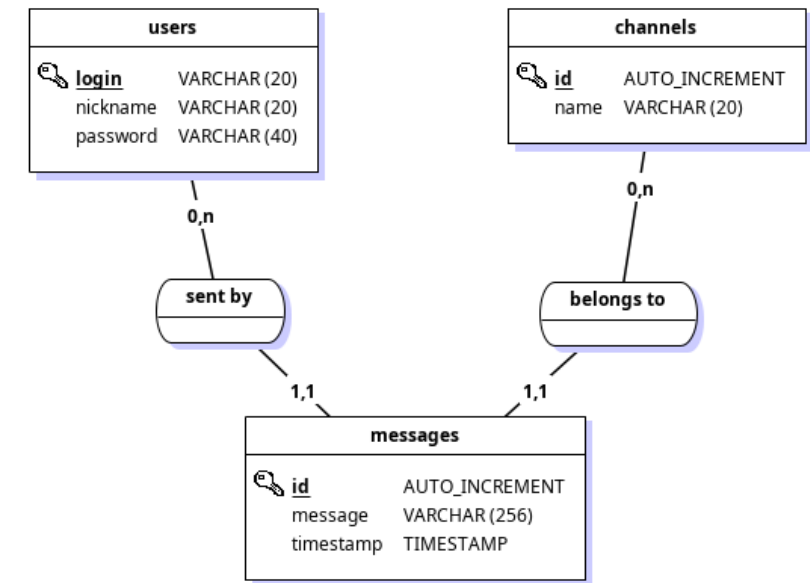
Paragraphe Open Sans

Conception de la base de données de l'application web :

Les nouveaux arbres ajoutés à l'aide de l'application web devront être stockés dans une base de données :

- Créez les entités nécessaires
- Ajouter les champs nécessaires
- Réfléchissez aux types et à la taille de chaque champ
- Créez les relations adéquates
- Réfléchissez aux cardinalités de vos relations
- Définissez vos clés primaires

→ **Attendus : le modèle conceptuel de données (MCD) de votre application web en PDF**



Attention

C'est à vous de définir les entités et les relations de votre modèle conceptuel de donnée en étudiant avec attention le cahier de charges fourni dans ce document.

Analyse : Interfaces client-serveur

Accès à la base de données

Définition des interfaces client-serveur de votre application web :

Réflexion autour des requêtes permettant au navigateur d'accéder à la base de données au travers de PHP :

GET

php/request.php/tweets/ Récupération des tweets

GET

php/request.php/tweets/?login=... Récupération des tweets d'un user

POST

php/request.php/tweets/ login=...&text=... Ajout d'un tweet

PUT

php/request.php/tweets/i login=...&text=... Modification du tweet

DELETE

php/request.php/tweets/i?login=... Suppression du tweet

Exemple d'interfaces client-serveur

[{"id": "1", "text": "Un premier tweet !!", "login": "cir2"},
{"id": "2", "text": "Un second tweet !!", "login": "cir3"}]

Exemple de réponse pour une requête de type GET

→ **Attendu : un document indiquant les interfaces client-serveur de votre application web en PDF**

Partie 2 : Conception

Conception : Fonctionnalités

5 pages

5 fonctionnalités principales sont attendues :

1. Page d'accueil
 2. Ajout d'arbres
 3. Visualisation dans un tableau et sur une carte des arbres de la base de données
 4. Prédiction du *cluster* des arbres entrés (à partir du script *Python* réalisé en IA)
 5. Prédiction de l'âge ou du risque de déracinement d'un arbre (à partir des scripts *Python* réalisé en IA)
- + Import des données initiales avec un script *Python* (pour les CIR uniquement ou en bonus)
 - + Choix de l'espèce avec complétion automatique
 - + Options de filtrage des arbres pour la visualisation (pour les CIR uniquement ou en bonus)
 - + Authentification pour l'ajout d'un arbre (en bonus)

Attention

Les arbres des fichiers CSV (utilisées en Big Data et IA) ne seront pas insérés dans votre base de données web. Sauf si le bonus 1 est réalisé.

Fonctionnalité 1

Page d'accueil

Ajout d'une page d'accueil :

- Ajout d'un menu pour naviguer entre les différentes pages du site
 - Descriptif rapide du projet
 - Image représentant le projet
- **Attendu : une page d'accueil qui s'affiche quand l'utilisateur entre l'url du site web**



Ajouter un nouvel arbre dans la base de données :

Le formulaire doit permettre d'ajouter un arbre avec au minimum les informations suivantes :

- L'espèce
- La hauteur totale, la hauteur du tronc et le diamètre du tronc
- Un indicateur permettant de savoir si l'arbre est remarquable
- La latitude et la longitude
- L'état de l'arbre (en place, supprimé...)
- Le stade de développement (jeune, adulte...)
- Le type de port (couronne, libre...)
- Le type de pied (terre, gazon...)

→ **Attendu : une page web contenant un formulaire permettant d'ajouter un nouvel arbre.**

Pour ces nouveaux arbres, l'âge et le déracinement ne doivent pas être fournis.

Attention

Les valeurs des différents champs devront être récupérés depuis la base de données et en adéquation avec les possibilités issues du fichier de données utilisé au début de la partie IA.

Fonctionnalité 3

Visualisation des arbres sur une carte

Visualisation des arbres de la base de données web :

Tous les arbres de la base de données web devront apparaître :

- Dans un tableau avec :
 - L'espèce
 - La hauteur totale, la hauteur du tronc et le diamètre du tronc
 - Un indicateur permettant de savoir si l'arbre est remarquable
 - La latitude et la longitude
 - L'état de l'arbre
 - Le stade de développement
 - Le type de port
 - Le type de pied
- Sur une carte :
 - La position de chaque arbre devra être visible sur la carte
 - Les détails d'un arbre seront visibles lors du survol du point avec la souris

→ **Attendu : une page web contenant un tableau et une carte avec les arbres de la base de données web**

Conseil

Il est fortement conseillé d'utiliser la bibliothèque *plotly* : <https://plotly.com/javascript/> pour l'affichage des données sur une carte.

Fonctionnalité 4

Prédiction du *cluster* des arbres

Prédiction du *cluster* des arbres entrés dans la base de données web :

Les clusters doivent pouvoir être prédit à la suite des actions suivantes :

- Clic sur un bouton « Prédire les clusters » en bas de la page web de visualisation des arbres
- Ouverture d'une nouvelle page
- Appel coté serveur de votre script *Python* permettant de prédire les *clusters*
- Affichage des arbres sur une carte
- Les arbres seront colorés en fonction de leur *cluster* d'appartenance

→ **Attendu : une page web affichant sur une carte les arbres avec leur *cluster* d'appartenance**

Information

Pour appeler votre script *Python* à partir de votre script PHP, vous pouvez utiliser la fonction `system` :
<https://www.php.net/manual/fr/function.exec.php>

Fonctionnalité 5

Prédiction de l'âge ou du risque de déracinement d'un arbre

Prédiction de l'âge ou du risque de déracinement d'un arbre de la base de données web :

L'âge ou le risque de déracinement d'un arbre doit être prédit, avec les méthodes de classification réalisées en IA à la suite des actions suivantes :

- Sélection d'un arbre dans le tableau à l'aide d'un bouton radio
- Clic sur un bouton « Prédire l'âge » ou « Prédire le déracinement » en bas de la page web de visualisation des arbres
- Ouverture d'une nouvelle page
- Appel coté serveur de vos scripts *Python* permettant de faire les prédictions
- Affichage du résultat comparatif des méthodes de classification réalisées en IA

→ **Attendu : une page web affichant la prédiction pour un arbre avec les différentes méthodes de classification**

Attention

Plusieurs scripts *Python* seront à appeler en fonction de la variable à prédire : âge ou risque de déracinement.

Livrables et évaluations

Format de l'archive :

Archive *ZIP*, *TGZ*, *7ZIP*, pas de *RAR* : projetweb_groupeX.zip (remplacer X par votre numéro de trinôme)

Le rendu final doit contenir :

- L'intégralité de vos codes sources commenté avec vos ressources (images, bases de données...) **ainsi que les données de l'IA** (scripts *Python*, modèles...) nécessaire au fonctionnement de votre application web
- Les visuels de votre maquette au format *PDF*
- Votre charte graphique au format *PDF*
- Votre *MCD* au format *PDF*
- Votre diagramme de *Gantt* en *PDF*
- *Une vidéo de fonctionnement de votre site*

Remarques :

- Malus possible sur l'un des membres du groupe si l'investissement est jugé trop faible
- Possibilité d'être interrogé durant le projet de façon individuelle
- Plagiat sévèrement sanctionné pour TOUS les membres du/des groupe(s)

Attention

Les livrables seront à poster sur l'intranet. Tout retard sera sanctionné (l'heure du réseau faisant foi).
Les fichiers au mauvais format ou avec un mauvais nommage seront pénalisés.

Présentation orale :

- Soutenance de 10 minutes (strict) + 5 minutes de questions
- Présentation en trinôme (pensez à vous répartir la parole)
- Présentez l'essentiel de votre projet

Code :

- Rendu de la partie analyse :
 - Maquette
 - Charte graphique
 - *MCD*
 - Interfaces client-serveur
 - Diagramme de *Gantt*
- Rendu de la partie conception :
 - Intégralité du code source
 - Scripts *SQL* (modèle + données)
 - *README* d'installation et de configuration

Barème indicatif : Soutenance 40% – Compétences 30% – Évaluation du code / maquette / MCD 30%

ISEN

ALL IS DIGITAL!



yncréa

MERCI
Des questions ?

