

Projet Année 3 Big Data/IA/Web

Partie Développement Web



Contexte du projet

Objectif

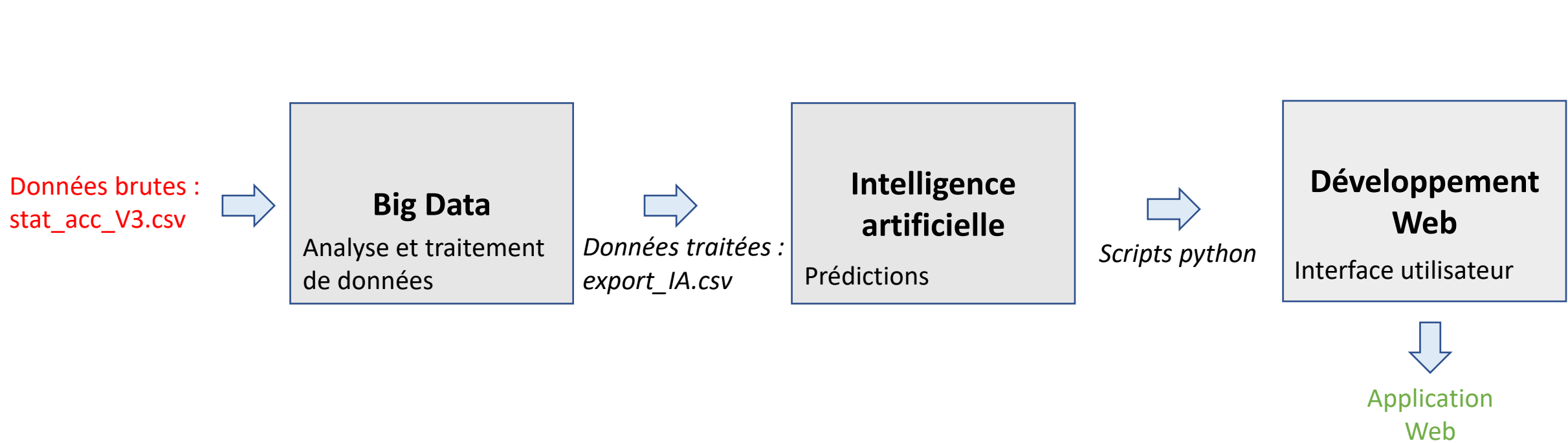
Concevoir et développer une application d'étude des accidents de la route

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 concernant les accidents corporels de la circulation routière en France.

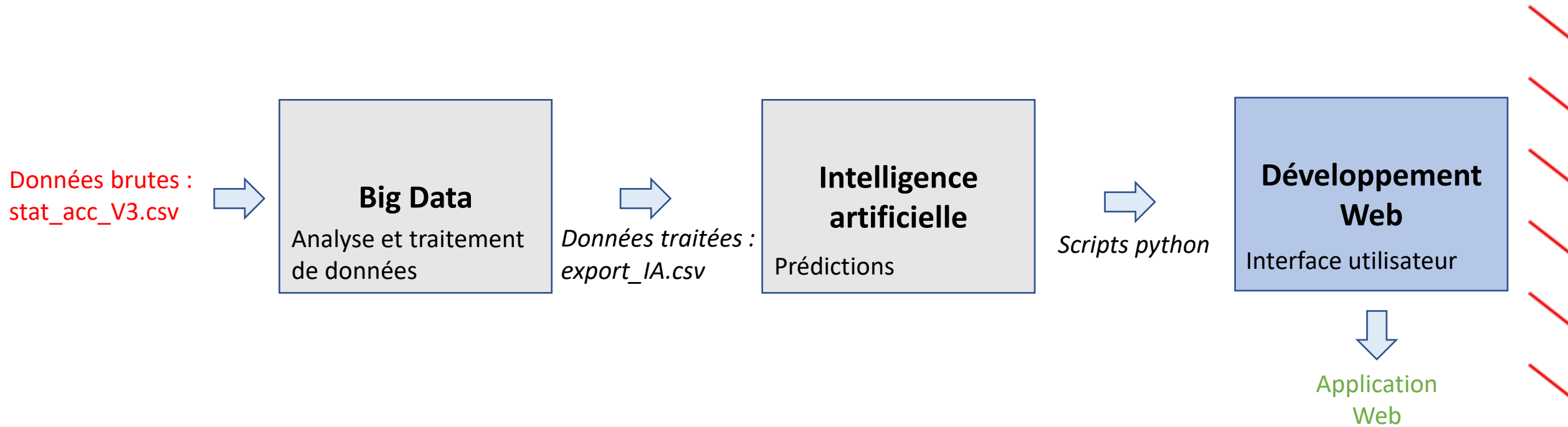
Objectifs de la partie Développement Web :

- 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

Déroulement du projet



Déroulement du projet



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

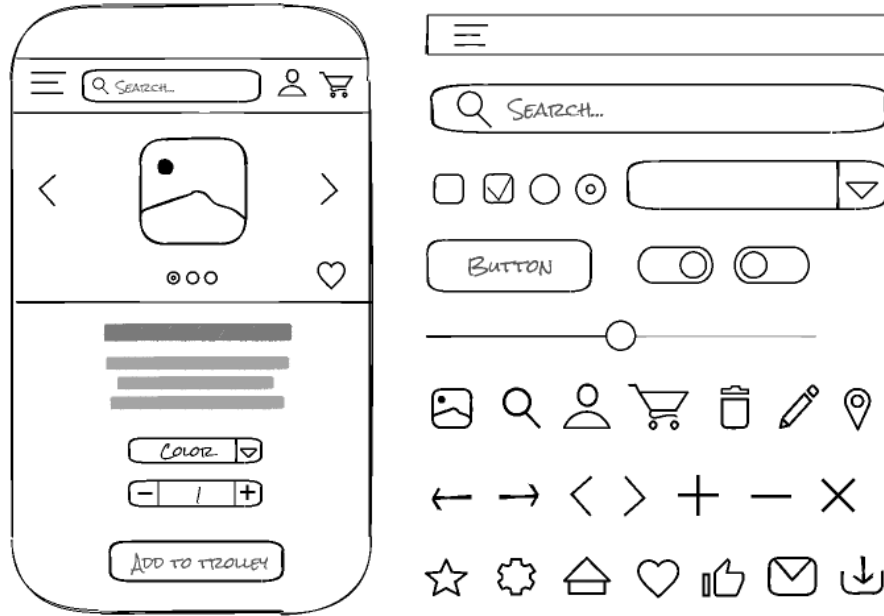
Cahier des charges

Maquette de l'application web

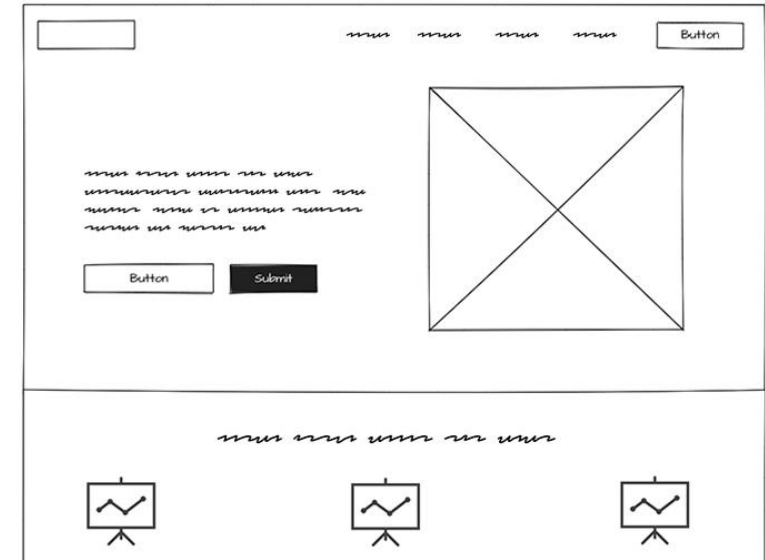
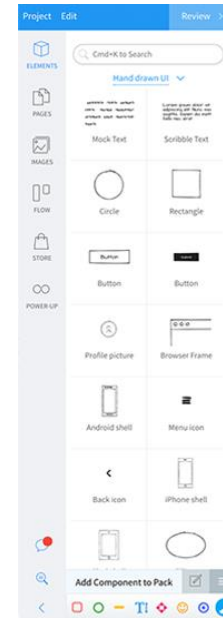
Figma, MockFlow

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**

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 :

CAVILITE

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

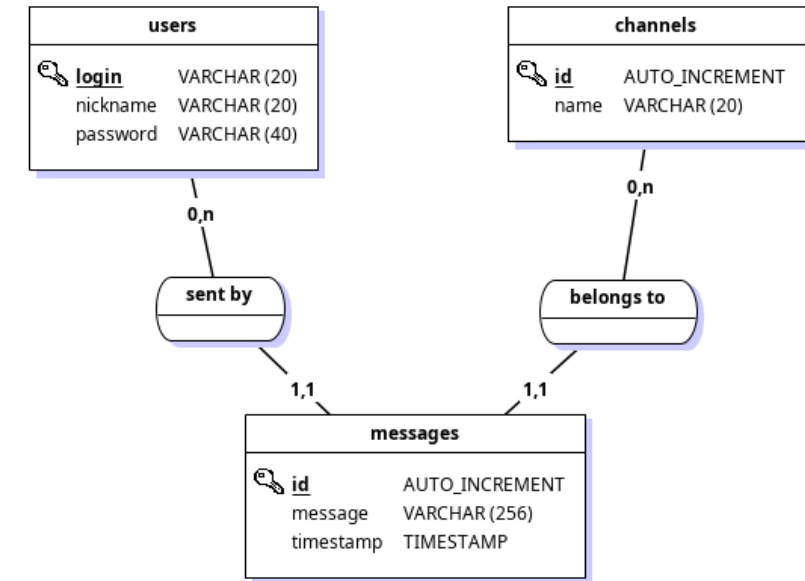
Open Sans

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

Les nouveaux accidents 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.

5 fonctionnalités principales sont attendues :

1. Page d'accueil
 2. Ajout d'accidents
 3. Visualisation des accidents ajoutés dans un tableau et sur une carte
 4. Prédiction du *cluster* des accidents entrés (à partir du script *Python* de l'IA)
 5. Prédiction de la gravité d'un l'accident (à partir des scripts *Python* de l'IA)
- + Import des données initiales avec un script Python (pour les CIR uniquement ou en bonus)
 - + Options de filtrage des accidents pour la visualisation (pour les CIR uniquement ou en bonus)
 - + Authentification pour l'ajout d'un accident (en bonus)

Attention

Les accidents 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 accident dans la base de données :

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

- L'âge du conducteur
- La date et l'heure
- La ville
- La latitude et la longitude
- Les conditions atmosphériques
- La luminosité de la scène
- L'état de la route
- L'état d'utilisation de la ceinture de sécurité

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

Pour ces nouveaux accidents, la gravité ne doit pas être fournie.

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 accidents ajoutés sur une carte

Visualisation des accidents de la base de données web :

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

- Dans un tableau avec :
 - L'âge du conducteur
 - La date et l'heure
 - La ville
 - La latitude et la longitude
 - Les conditions atmosphériques
 - La luminosité de la scène
 - L'état de la route
 - L'état d'utilisation de la ceinture de sécurité
- Sur une carte :
 - La position de chaque accident devra être visible sur la carte
 - Les détails d'un accident seront visibles lors du survol du point avec la souris

→ **Attendu : une page web contenant un tableau et une carte avec les accidents 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 accidents

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

Les clusters doivent pouvoir être prédit suite aux actions suivantes :

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

→ **Attendu : une page web affichant sur une carte les accidents 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 la gravité d'un l'accident

Prédiction de la gravité d'un accident entré dans la base de données web :

La gravité d'un accident doit être prédite, avec la méthode *KNN* et les méthodes de « haut niveau », suite aux actions suivantes :

- Sélection d'un accident dans le tableau à l'aide d'un bouton radio
 - Clic sur un bouton « Prédire la gravité » en bas de la page web de visualisation des accidents
 - Ouverture d'une nouvelle page
 - Appel coté serveur de vos scripts *Python* permettant de prédire la gravité
 - Affichage de la gravité de l'accident avec un comparatif des méthodes : *KNN*, *SVM*, Random Forest, *MLP*
- **Attendu : une page web affichant la gravité d'un accident avec les différentes méthodes de classification**

Attention

Plusieurs scripts *Python* seront à appeler en fonction de la méthode à utiliser. Méthode simple : *KNN* et méthodes de « haut niveau » : *SVM*, Random Forest, *MLP*

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



Partie *back-end* :

- Exclusivement en *PHP*



- SCBD* 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)

Calendrier prévisionnel

Déroulement

Calendrier prévisionnel

Organisation du projet au jour le jour :

Demi-journée 1	Demi-journée 3	Demi-journée 5	Demi-journée 7
<ul style="list-style-type: none">Présentation de la partie Web du projetGestion de projetMaquetteCharte graphique	<ul style="list-style-type: none">Gestion de projetBase de données (MCD)Définition de l'API en PHP	<ul style="list-style-type: none">Gestion de projetFront pour la visualisation d'accidentsCréation de l'API en PHP pour la visualisation	<ul style="list-style-type: none">Front pour la prédiction sur un accidentCréation de l'API en PHP pour l'appel des scripts IADocumentation du codePréparation de la soutenance
Demi-journée 2	Demi-journée 4	Demi-journée 6	Demi-journée 8
<ul style="list-style-type: none">Gestion de projetBase de données (MCD)Interface JavaScript/PHP	<ul style="list-style-type: none">Gestion de projetFront pour l'ajout d'accidentsCréation de l'API en PHP pour l'ajout	<ul style="list-style-type: none">Gestion de projetFront pour la prédiction sur un accidentCréation de l'API en PHP pour l'appel des scripts IA <p>QCM de 30 minutes (individuel)</p>	<p>Recette (sur le serveur) de 10 minutes (strict) + 5 minutes de questions (trinôme)</p> <p>Dépôt des livrables sur l'ENT (trinôme)</p>

Travail en trinôme :

- Chaque étudiant dans le trinôme connaît l'ensemble du projet : **utiliser le gestionnaire de code *Git***
- Attention à bien se répartir le travail en prévoyant les tâches de chacun avec un **diagramme de Gantt**

Ressources externes :

- Tous les documents sont autorisés
- Attention à utiliser avec une grande précaution tout document extérieur : site de vulgarisation, forum, code d'autrui

Documentation du projet :

- Au fur et à mesure
- Standardisée
- Livraison de code ou de documents :
 - Ne pas attendre la dernière minute pour poster un livrable
 - Préparer des livrables intermédiaires (surtout pour les sources)
 - Sauvegarder régulièrement vos données

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*

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

QCM :

- QCM de 30 minutes à points négatifs
- Évaluation individuelle

Code :

- Rendu de l'intégralité de vos codes sources avec les ressources associées
- Rendu de votre maquette
- Rendu de votre charte graphique
- Rendu de votre *MCD*

Barème indicatif : Soutenance 40% – QCM individuel 20% – Évaluation du code/maquette/MCD 40%

ISEN

ALL IS DIGITAL!



yncréa

MERCI
Des questions ?

