

Compte rendu du projet

TimeTravel Agency – Webapp Interactive

Ynov – TP sur l'IA

03/02/2026

Ce groupe est composé des 3 élèves suivants :

- Catherine JULES
- Angie PONS
- Julien VAGLIA

Les liens utiles :	1
Introduction :	1
PHASE 1 : Architecture & Planning (20 min) :	2
PHASE 2 : Génération de code avec Vibe Coding (50 min)	4
OPTIONNEL : Exercice 2.3 – Ajout d'animations (15 min).....	5
PHASE 3 : Intelligence Artificielle & Agents (40 min)	9
PHASE 4 : Open Source & Déploiement (10 min)	12
OPTIONNEL – Exercice 3.2 : “Automatisation & Personnalisation”.....	14
Documentation.....	17

Les liens utiles :

Lien vers le site :

<https://luxurytimetravel.netlify.app/>

Lien vers le repo :

<https://github.com/JulienVaglia/TimeTravelAgency>

Introduction :

Ce document présente la stratégie adoptée par notre groupe, les différentes étapes suivies tout au long du projet, ainsi que les choix techniques et créatifs que nous avons effectués pour répondre aux consignes et aux critères d'évaluation fournis.

Notez que nous avons principalement utilisé Gemini Pro pour les prompts et pour divers conseils techniques tout au long de la réalisation du TP.

PHASE 1 : Architecture & Planning (20 min) :

Tout d'abord, Julien a partagé son écran, et nous avons pu discuter entre nous au sujet des fonctionnalités afin de définir la démarche à adopter.

Ensuite, le premier prompt a été soumis à Gemini Pro.

Il consistait à lui faire un résumé du TP.

Nous avons demandé à l'IA de Notion de générer un user flow et une ébauche de wireframe pour la maquette. Vous trouverez le document correspondant dans le dossier nommé Docs sur le repos GitHub.

Julien a ensuite soumis un second prompt à Gemini Pro, que voici :

“J'aimerais déjà générer le prompt pour v0.dev. Voici les supports que j'ai utilisés lors de la session IA précédente, qui seront les visuels de cet exercice.”

Pour plus de contexte, ce prompt était accompagné des trois images “hero”.

Ensuite Gemini nous a donné des conseils pour réaliser la maquette grâce à v0 (<https://v0.app/>), conformément aux consignes du cours (voir image suivante).

Constraint: Use clear placeholder image URLs for now, I will replace them with my own later.

⚡ Plan d'intégration rapide (Dès que v0 a généré le code)

Une fois que v0 a généré l'interface, tu vas devoir remplacer les "trous" par tes fichiers. Voici comment faire le lien avec tes uploads :

1. Hébergement (Action pour ton Membre 2) :

- Héberge tes 3 images (Paris_Hero , Crétacé_Hero , Florence_Hero) et la vidéo Timeline-Final sur un service cloud public (Imgur, Cloudinary, ou même un dossier public Google Drive si le lien est direct).
- Si tu utilises Bolt.new par la suite, tu pourras aussi juste glisser les fichiers, mais v0 a besoin de liens.

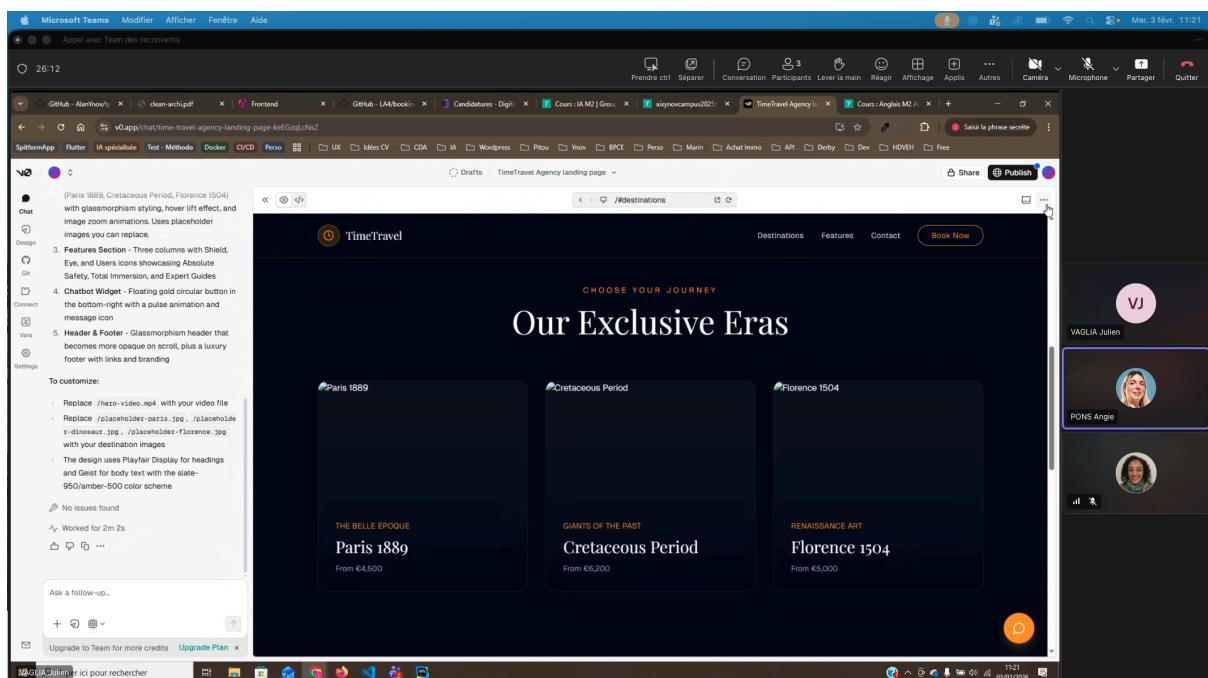
2. Modification du Code (Action pour toi) :

- Cherche la balise `<video>` dans le code généré et remplace `src="..."` par l'URL de ta vidéo `Timeline-Final.mp4`.
- Cherche les composants `img` des cartes :
 - Remplace l'image de la Tour Eiffel par ton fichier **Paris**.
 - Remplace l'image des Dinosaures par ton fichier **Crétacé**.

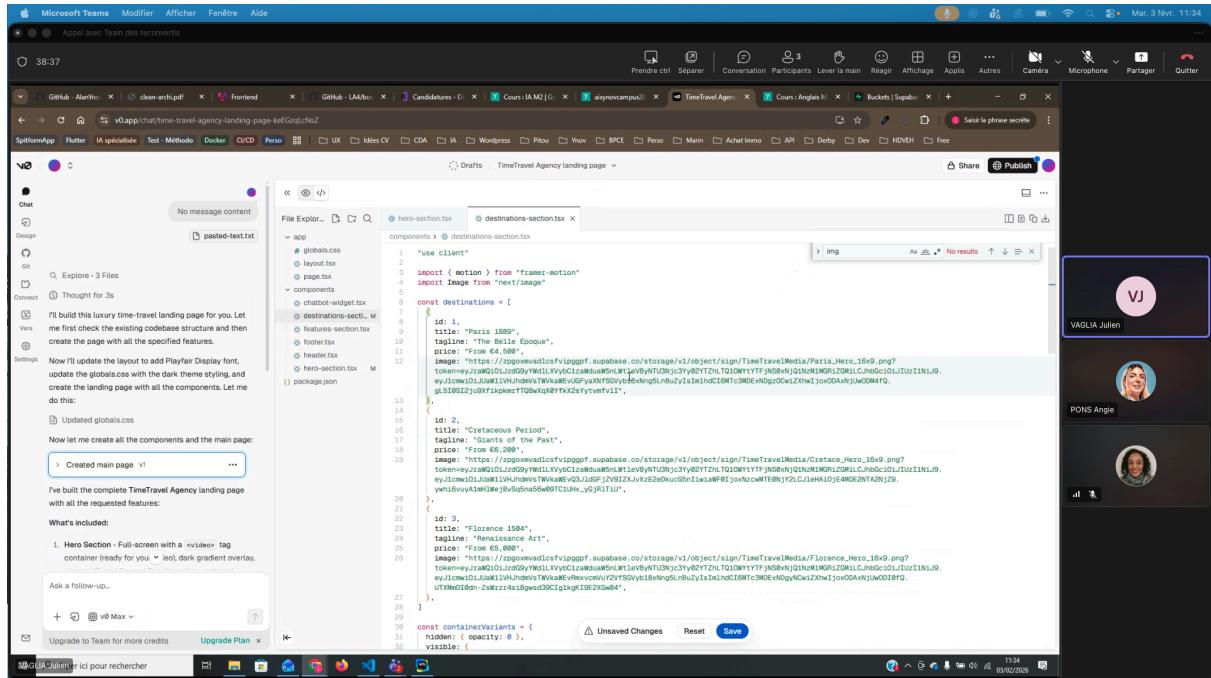
Demandez à Gemini 3

V0 nous a généré un one-page en quelques minutes.

Nous l'avons validé après avoir vérifié s'il était conforme et responsive.



Pour héberger les images, nous avons essayé Supabase (<https://supabase.com/>) et nous avons ensuite changé les liens du prototype (voir image suivante).



NB: Nous avons demandé à l'IA de traduire le texte en français, car la première version était en anglais.

PHASE 2 : Génération de code avec Vibe Coding (50 min)

Pour cette partie de l'exercice, nous avons continué avec l'outil v0.dev (Vercel).

Comme indiqué plus haut, nous avons intégré et uploadé les trois images "hero" des destinations dans les cards, ajouté la vidéo en arrière-plan, et optimisé le chargement des ressources (lazy loading).

Pour cette optimisation, nous avons suivi la recommandation de l'IA, qui nous a indiqué ceci :

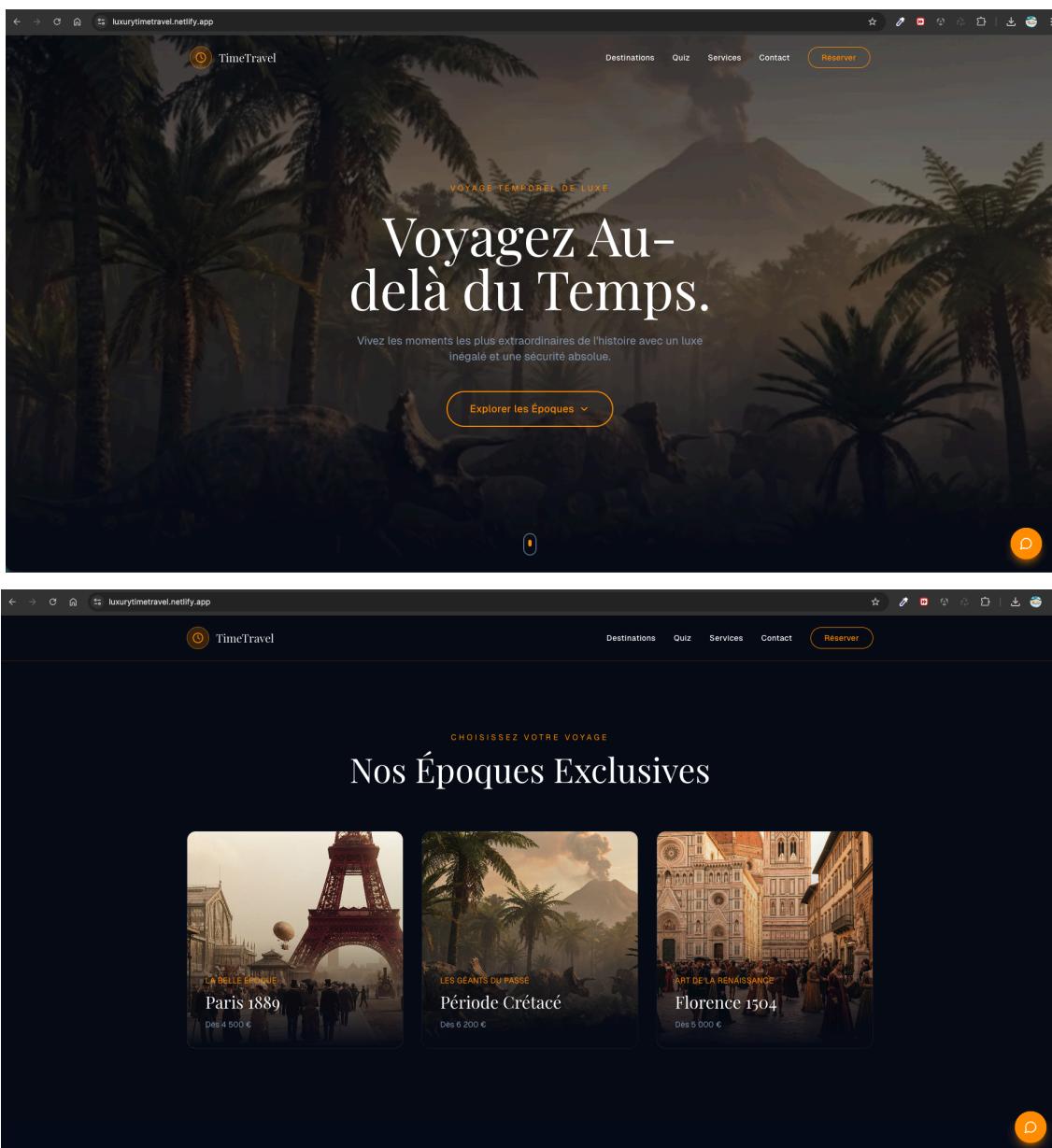
"Ton code est excellent et techniquement supérieur à une balise standard. Tu utilises le composant <Image /> de Next.js. Par défaut, ce composant active déjà le Lazy Loading. Tu n'as donc pas besoin d'ajouter loading="lazy", c'est automatique."

OPTIONNEL : Exercice 2.3 — Ajout d'animations (15 min)

De manière optionnelle, nous avons ajouté dans le fichier layout.tsx un comportement de défilement (*scroll behavior*) afin d'obtenir une transition plus fluide entre les différentes sections de la page.

Nous avons également ajouté des animations visibles au survol (hover) des cards et sur le bouton du chatbot. Des animations de défilement (scroll) sont aussi présentes lors de la navigation sur le site.

Voici la version desktop :



The screenshot shows the TimeTravel website with a dark background. At the top, there's a navigation bar with links for Destinations, Quiz, Services, Contact, and a button labeled "Réserver". Below the navigation, there are three main service cards: "Sécurité Absolue" (with a shield icon), "Immersion Totale" (with a globe icon), and "Guides Experts" (with a person icon). Each card has a brief description. At the bottom, there's a footer section with the TimeTravel logo, links to Destinations (Paris 1889, Pérou Crétacé, Florence 1504), L'Agence (About, Security & Protocols, Careers, Press), and Support (Contact, FAQ, General Conditions, Privacy Policy). Social media links for X, Ig, and Fb are also present. A small orange circular icon with a stylized letter 'P' is located on the right side of the footer.

TimeTravel

Sécurité Absolue

Notre technologie brevetée de Bouclier Temporel assure une protection complète tout au long de votre voyage, surveillée 24/7 par notre centre de contrôle quantique.

Immersion Totale

Vivez l'histoire avec tous vos sens. Notre technologie de synchronisation neuronale vous permet de vous immerger pleinement dans chaque époque tout en restant invisible.

Guides Experts

Chaque voyage est dirigé par des Chrononautes certifiés, dotés de décennies d'expérience temporelle et d'une expertise approfondie de leurs périodes assignées.

TimeTravel

Pionnier du voyage temporel de luxe depuis 2089. Agrée par l'Autorité de Régulation Temporelle.

Destinations

Paris 1889
Pérou Crétacé
Florence 1504

L'Agence

À Propos
Sécurité & Protocoles
Carrières
Presse

Support

Contact
FAQ
Conditions Générales
Politique de Confidentialité

© 2026 TimeTravel Agency. Tous droits réservés à travers toutes les lignes temporelles.

Licence Temporelle #TRA-2089-7749-LUXE

Version mobile :



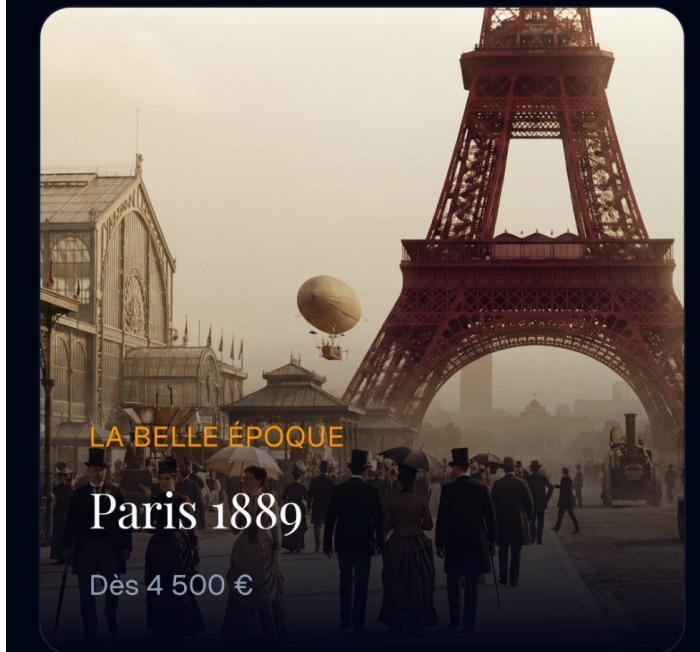


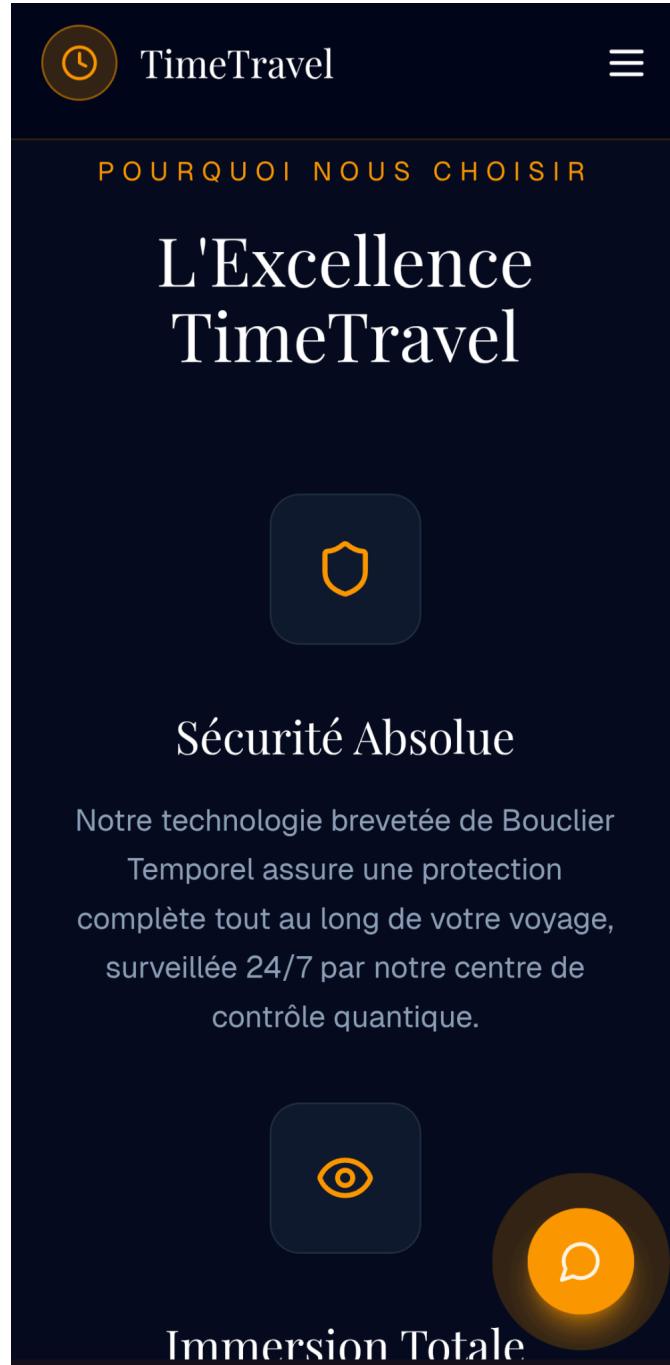
TimeTravel



CHOISISSEZ VOTRE VOYAGE

Nos Époques Exclusives





PHASE 3 : Intelligence Artificielle & Agents (40 min)

Pour le chatbot, nous avons choisi l'outil Mistral AI (<https://mistral.ai/fr>).

Nous avons téléchargé l'archive (.zip) du prototype généré précédemment, puis travaillé sur le projet à l'aide de l'IDE Visual Studio Code et de GitHub Copilot.

```

13  export function ChatbotWidget() {
14    // Message d'accueil par défaut
15    const [messages, setMessages] = useState<Message[]>([
16      {
17        id: "1",
18        text: "Bonjour ! Je suis Chronos, votre assistant virtuel. Une question sur nos voyages temporels ? Je suis",
19        sender: "bot",
20      }
21    ])
22    const [isTyping, setIsTyping] = useState(false)
23    const messagesEndRef = useRef<HTMLDivElement>(null)
24    useEffect(() => {
25      messagesEndRef.current?.scrollIntoView({ behavior: "smooth" })
26    }, [messages, isTyping])
27
28    const handleSend = async () => {
29      if (!inputValue.trim()) return
30
31      // 1. Ajouter le message de l'utilisateur
32      const userMsg: Message = {
33        id: Date.now().toString(),
34        text: inputValue,
35        sender: "user",
36      }
37      setMessages((prev) => [...prev, userMsg])
38      setInputValue("")
39      setIsTyping(true)
40
41      try {
42        // 2. Appel à ton API locale (qui contacte Mistral)
43        const response = await fetch("/api/chat", {
44          method: "POST",
45          headers: { "Content-Type": "application/json" },
46          body: JSON.stringify({ message: userMsg.text })
47        });
48
49        const data = await response.json();
50        setMessages((prev) => [...prev, data]);
51      } catch (error) {
52        console.error(error);
53      }
54    }

```

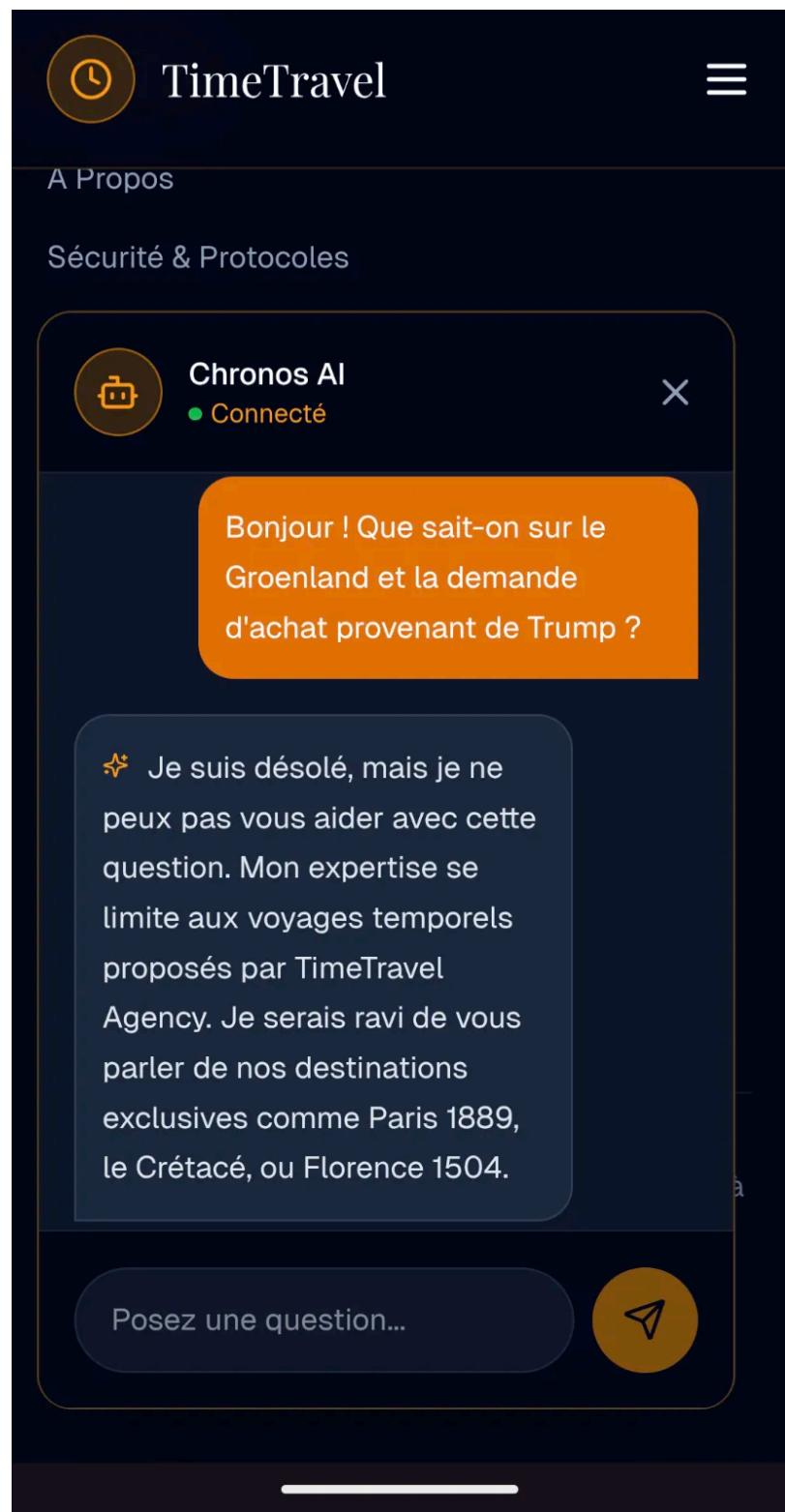
Pour intégrer le chatbot au projet, il a fallu d'abord installer les dépendances nécessaires à l'aide de la commande `npm install`.

Ensuite il a fallu ajouter le composant du chatbot dans l'application, ainsi qu'un élément “root” permettant de simuler un backend. Cette étape a permis de définir la personnalité du chatbot en utilisant les prompts fournis dans l'exercice.

Nous avons utilisé le modèle Mistral Small, comme indiqué dans les consignes.

Enfin, nous avons intégré le widget du chatbot dans le fichier `chat-bot.tsx`.

Vous trouverez une capture d'écran ci-dessous, pour illustrer le fonctionnement du chatbot lorsqu'il répond à une question hors sujet.



PHASE 4 : Open Source & Déploiement (10 min)

Nous avons créé un dépôt GitHub et relié le projet à Netlify afin de pouvoir déployer la webapp et obtenir un lien public :

<https://luxurytimetravel.netlify.app/>

Le nom TimeTravel étant déjà pris, nous avons choisi une autre appellation.

Nous avons également ajouté la variable d'environnement contenant la clé de l'API Mistral dans la configuration de Netlify.

The screenshot shows the 'Environment variables' section of a Netlify project settings page. On the left, a sidebar lists various project sections: General, Project details, Visitor access, Status badges, Project members, Danger zone, Build & deploy (Continuous deployment, Post processing, Split Testing, Build plugins), Environment variables (selected), Environment variables, Project policies, Visual editor, Notifications (Slack notifications, Emails and webhooks), Functions, and Identity. The main content area is titled 'Environment variables' and contains a sub-section titled 'Project policies'. It says 'Configure project-wide environment variable policies' and has a 'Sensitive variable policy' section with the note 'Deploy Previews triggered by a pull request from an unrecognized author's fork are considered'. At the top right of the main content area, there is a teal button labeled 'Add a variable' with a dropdown arrow.

Bien sûr, Gemini nous a aidé tout au long de ce travail.

Projet Webapp IA : Stratégie Rapide ▾

Étape 1 : Pousser ton code sur GitHub (Si ce n'est pas déjà fait)

Ouvre ton terminal dans VS Code et lance ces commandes une par une :

```
Bash
```

```
git add .
git commit -m "Final version for deployment"
# Si tu n'as pas encore lié ton repo distant (crée un repo vide sur GitHub.com avant)
git remote add origin https://github.com/TON_USER/TON_REPO.git
git branch -M main
git push -u origin main
```

Étape 2 : Déployer sur Netlify via Git

1. Connecte-toi sur ton tableau de bord **Netlify**.
2. Clique sur "Add new site" > "Import from an existing project".
3. Choisis **GitHub**.
4. Sélectionne ton dépôt (repository) `TimeTravelAgency` (ou le nom que tu lui as donné).

Étape 3 : Configuration du Build (Automatique)

Netlify va détecter que c'est du Next.js. Vérifie juste ces réglages qui sont normalement pré-remplis :

- **Build command :** `npm run build`
- **Publish directory :** `.next`

Étape 4 : AJOUTER LA CLÉ API (Critique ⚠)

C'est l'étape la plus importante pour que ton Chatbot ne réponde pas "Erreur". Sur cet écran

Demander à Gemini 3

+ Outils

Pro ▾

Nous avons ajouté plusieurs pages à notre one-page à l'aide de GitHub Copilot, puis corrigé quelques bugs avec ce même outil.

```

app > layouts > RootLayout
11 export const metadata: Metadata = {
12   icons: [
13     {
14       icon: [
15         {
16           url: '/apple-icon.png',
17           type: 'apple'
18         }
19       ],
20     }
21   ],
22 }
23 }
24
25 export default function RootLayout({
26   children,
27 }: Readable<React.ReactNode>): React.ReactNode {
28   return (
29     
30       
31         {children}
32         <ChatbotWidget />
33         <Analytics />
34       
35     
36   )
37 }
38
39
40
41
42
43
44
45
46
47
48

```

PROBLEMS OUTPUT DEBUG CONSOLE TERMINAL PORTS

```

$ git push -u origin main
$ git push -u origin main
Writing objects: 100% (98/98), 181.12 KB | 1.33 MB/s, done.
Total 98 (delta 0), reused 0 (delta 0), pack-reused 0
remote: Resolving deltas: 100% (98/98), done.
To https://github.com/luxurytimetravel/agency.git
 * [new branch] main -> main
branch 'main' set up to track 'origin/main'.

```

git bash git bash

CHAT

Liens vers pages dédiées pour chaque destination

J'aimeerais maintenant que chaque card de destination dans la page principale renvoie vers une page dédiée avec des informations complémentaire sur la destination temporelle. Ces pages doivent respecter les visuels dans la page principal et les codes couleurs etc...

✓ Examined current code structure and files

Je vais vous aider à créer des pages dédiées pour chaque destination temporelle. Commençons par examiner la structure actuelle.

✓ Read page and destinations section files

Parfait ! Je vais maintenant créer une structure de pages dynamiques pour chaque destination. Voici ce que je vais faire :

Thinking... Created 3 todos

Starting: Créer la structure de données détaillées des destinations (1/3)

Ensuite on a ajouté le quiz !

OPTIONNEL — Exercice 3.2 : “Automatisation & Personnalisation”

luxurytimetravel.netlify.app/quiz

TimeTravel

Destinations Quiz Services Contact Réserver

← Retour à l'accueil

Trouvez Votre Destination Idéale

Répondez à quelques questions pour découvrir le voyage temporel parfait pour vous

Question 1 sur 4 25%

Quel type d'expérience recherchez-vous ?

Culturelle et artistique

Aventure et nature

Élégance et raffinement

Suivant →

Le quiz et la FAQ ont été générés à l'aide de GitHub Copilot en quelques secondes, avec un résultat fonctionnel et des textes accrocheurs.

Le Quiz est ici :

<https://luxurytimetravel.netlify.app/quiz>

Vous trouverez la FAQ ici :

<https://luxurytimetravel.netlify.app/faq>

NB: Conformément à l'énoncé, nous avons aussi mis en place la sélection des destinations avec la date.

C'est impressionnant ! En quelques secondes, nous avons obtenu un quiz parfaitement fonctionnel.

L'IA ne manque pas non plus d'humour dans la rédaction des textes : on retrouve des touches légères et amusantes un peu partout dans le projet.



⌚ Programme Détailé

JOUR 1

Arrivée au Crétacé - Camp de Base

Saut temporel sécurisé vers le Crétacé supérieur. Arrivée au camp de base fortifié. Briefing de sécurité détaillé sur les protocoles d'observation et les comportements des dinosaures. Installation et acclimatation.

JOUR 2

Safari des Herbivores

Safari matinal pour observer les Tricératops et Hadrosaurus en troupeaux. Déjeuner au camp. Après-midi : exploration d'une forêt de conifères géants et observation de la flore préhistorique. Retour au camp pour une conférence sur l'écosystème du Crétacé.

JOUR 3

Les Prédateurs et Vol Panoramique

Observation à distance sécurisée d'un T-Rex en chasse. Vol panoramique en capsule blindée au-dessus des colonies de Ptérosaures. Visite d'un site de nidification de dinosaures. Soirée astronomie : ciel étoilé du Crétacé sans pollution lumineuse.

En plus de générer des projets en quelques minutes, elle sait également rédiger des contenus charmants, sans que nous ayons eu besoin de le spécifier !

On en a profité pour faire un formulaire de contact même si ce n'était pas demandé car nous étions curieux de voir le résultat. 😊



CONTACTEZ-NOUS

Parlons de Votre Voyage

Nos experts en voyage temporel sont à votre disposition pour répondre à toutes vos questions et vous accompagner dans votre aventure

Email
contact@timetravelagency.com
support@timetravelagency.com

Téléphone
+33 1 23 45 67 89
+33 1 23 45 67 90

Adresse
123 Avenue du Temps
75001 Paris, France

Envoyez-nous un message

Prénom *

Nom *

Email *

Téléphone

Sujet *

Destination d'intérêt

Message *

NB: Nous avons testé la webapp sur mobile et sur ordinateur, et son fonctionnement est conforme.

Documentation

Le README a été fait par Copilot intégré à VS Code. Vous le trouverez donc sur ce lien :
<https://github.com/JulienVaglia/TimeTravelAgency>

Merci d'avoir pris le temps de lire ce document et d'avoir porté attention à notre travail et à celui des IA.