**StellarMap**



Figure 1 : Système solaire (freepik, -)

# Table des matières

[1 Table des matières 2](#_Toc133831538)

[2 Analyse préliminaire 3](#_Toc133831539)

[2.1 Introduction 3](#_Toc133831540)

[2.2 Objectifs 3](#_Toc133831541)

[2.3 Planification initiale 3](#_Toc133831542)

[3 Analyse / Conception 4](#_Toc133831543)

[3.1 Concept 4](#_Toc133831544)

[3.2 Stratégie de test 4](#_Toc133831545)

[3.3 Risques techniques 4](#_Toc133831546)

[3.4 Planification 4](#_Toc133831547)

[3.5 Dossier de conception 4](#_Toc133831548)

[4 Réalisation 5](#_Toc133831549)

[4.1 Dossier de réalisation 5](#_Toc133831550)

[4.2 Description des tests effectués 5](#_Toc133831551)

[4.3 Erreurs restantes 5](#_Toc133831552)

[4.4 Liste des documents fournis 5](#_Toc133831553)

[5 Conclusions 6](#_Toc133831554)

[6 Bibliographie 7](#_Toc133831555)

[7 Table des illustrations 7](#_Toc133831556)

[8 Lexique 7](#_Toc133831557)

[9 Annexes 8](#_Toc133831558)

[9.1 Résumé du rapport du TPI / version succincte de la documentation 8](#_Toc133831559)

[9.2 Journal de travail 8](#_Toc133831560)

[9.3 Archives du projet 8](#_Toc133831561)

# Analyse préliminaire

## Introduction

Ce projet est réalisé dans le cadre du au travail pratique individuel (TPI) qui s’effectue lors de la dernière année de CFC en informatique.

Ce travail s’effectue sur une période de réalisation de 90 heures, entre le 2 mai de 8h50 au 2 juin à 15h20.

Le sujet est une carte 3D interactive du système solaire, il a été choisi à la suite de la proposition de ce sujet par le candidat.

## Objectifs

L’objectif de ce projet est de crée une carte interactive du système solaire sur le quel il sera possible de voir les 8 planètes et leurs lunes ainsi que le soleil. Il sera possible de tourner autour du soleil et d’observer les planètes sous un autre angle. Une description des planètes devra s’afficher lorsqu’un utilisateur clique sur celui-ci, de plus il sera possible d’accélérer la vitesse de déplacement des planètes.

Sept objectifs spécifiques sont à atteindre :

1. La carte s’affiche avec toutes les huit planètes.
2. L’utilisateur peut naviguer dans le système solaire.
3. Ergonomie et facilité d’utilisation du produit (Bastien et Scapin).
4. Les informations des différentes planètes s’affichent quand on clique dessus.
5. Le site est « responsive » et peut être utilisé depuis un smartphone ou une tablette.
6. L’utilisateur peut modifier la vitesse de déplacement des planètes.
7. Les angles de vue du système peuvent être déterminé par l’utilisateur.

Tout au long de mon travail je vais me conformer aux critères d’évaluation établis par le canton de Vaud (Schwab, 2018).

## Cahier des charges

## Planification initiale

La planification initiale se découpe en cinq sprints découpés sur cinq semaines.

Une image contenant texte, capture d’écran, nombre, Police

Description générée automatiquement

# Analyse / Conception

## Concept

Ce site est conçu pour afficher un système solaire en 3D, le visiteur pourra tourner autour de la carte et ainsi voir les planètes sous d’autres angles, il aura aussi la possibilité d’accélérer le temps afin de voir le déplacement des planètes à des vitesses différentes.

Ce système se verra affiché les 8 planètes du système solaire, leurs lunes ainsi que les astéroïdes à proximité de la planète terre.

Le projet sera hébergé sur swisscenter, les liens des accès au projet sont les suivantes :

Code source : <https://github.com/Juillet-Mikael/TPI>

Planification du projet : <https://icescrum.cpnv.ch/p/TPIJUILLET/#/project>

Documentation :

<https://github.com/Juillet-Mikael/TPI/blob/main/documents/documentation.docx>

Journal de travail :

<https://github.com/Juillet-Mikael/TPI/blob/main/documents/journaux.xlsm>

#### Requêtes

Il est prévu d’utiliser deux API de la Nasa, l’api « [horizon view](https://ssd.jpl.nasa.gov/horizons/app.html#/)» permet de récupérer des informations précises sur les objets spatiaux dans notre système solaire. Horizon sera utilisé pour récupérer toutes les informations nécessaires au placement, et à la définition des planètes comme le volume, la densité, la position précise actuel, la température etc.

La deuxième API est « Near Earth Object » qui permet de récupérer la liste des objets proche de la terre à un temps donnée, elle sera utilisée pour placer approximativement les astéroïdes sur la carte car aucune données de placement précise ne peut être récupérer via cette api.

#### Maquettes

## Stratégie de test

Je vais uniquement effectuer des tests manuels, j’ai choisi de faire cela car je ne dispose que de peu de temps pour crée mon projet de plus j’ai une très faible connaissance de la création de tests en Javascript.

J’effectuerai à chaque fin de sprint une série de tests en dès l’implémentation de l’api en semaine numéro 2.

Les tests en rapport à l’api seront des tests unitaires sur des requêtes précises.

Les tests en rapport à la vue comme pour le déplacement des planètes seront des tests fonctionnels. Ces tests on pour objectif de s’affurer du bon déroulement d’une tâche.

## Risques techniques

Les risques techniques sont mon manque de connaissance à l’utilisation de Three.js (three.js, 2023) et Ajax (W3schools, -), malgré de solides base acquises grâce a la préparation au TPI, j’ai tout encore besoin de beaucoup me référer aux documentations.

## Planification

## Dossier de conception

### Logiciels / Framework utilisé :

|  |  |  |
| --- | --- | --- |
| **Nom** | **Version** | **Utilisation** |
| Visual Studio Code (visualstudio, 2023) | 1.74.3 | Editeur de code |
| Balsamiq Wireframe (balsamiq, 2023) | 4.6.5 | Wireframe |
| Figma (figma, 2023) | - | Mockup |
| HTML, CSS | Html 5, CSS 3 | Mise en page |
| Vite.js (vitejs, 2023) | 4.1.1 | Frontend Tooling |
| Three.js (WebGL) (three.js, 2023) | 0.149.0 | Rendu 3D |
| Ajax (W3schools, -) |  | Requêtes |
| moment (momentjs, 2023) | 2.29.4 | Gestion des dates |

# Réalisation

## Dossier de réalisation

### Classes

### Requête API

Les deux requêtes sont placées dans le dossier model dans un fichier appelé requests.js.

Une image contenant texte, capture d’écran, logiciel, affichage

Description générée automatiquement

#### Variables d’environnement

Un fichier .env a été crée pour stocker les données sensibles tel que la clé api.

Une image contenant texte, capture d’écran, Police, ligne

Description générée automatiquement

#### Horizon system

Url de requête : <https://ssd.jpl.nasa.gov/api/horizons.api>

Certains paramètres sont absolument obligatoires en voici la liste :

* **Format**, sera retourné en Json.
* **Command**, représente l’id de l’objet ciblé (ex. terre = 399).
* **Objet\_data**, représente les données de l’objet tel que la période de rotation.
* **Make\_Ephem**, représente les données de placement des planètes.
* **Start\_time**, la date de départ des données en format années-mois-jour.
* **End\_time**, la date de fin des données en format années-mois-jour.
* **Step\_size**, la durée séparant les informations de placement de l’objet ciblé.

Il est important d’ajouter le mode de requête en format « no-cors », sinon l’erreur suivante survient.

|  |
| --- |
| No 'Access-Control-Allow-Origin' header is present on the requested resource. If an opaque response serves your needs, set the request's mode to 'no-cors' to fetch the resource with CORS disabled. |

#### Near Earth Objects

Url de requête : <https://api.nasa.gov/neo/rest/v1/feed>

## Répertoires

Code source : <https://github.com/Juillet-Mikael/TPI>

Planification du projet : <https://icescrum.cpnv.ch/p/TPIJUILLET/#/project>

Documentation se situe aussi dans un dossier nommé doc au sein du projet Git.

**Architecture des documents :**

* TPI
  + documents
    - journals
    - documentation
    - planification initiale
    - diagrams
      * diagrame de classe
      * diagrame de scéquence
    - instruction
  + src
    - model
    - view
    - controller

## Description des tests effectués

## Erreurs restantes

## Liste des documents fournis

# Conclusions

# Bibliographie

freepik. (-, - -). Vecteur gratuit système de système solaire classique avec deisgn plat. *Vecteur gratuit système de système solaire classique avec deisgn plat*. -, -, -. Récupéré sur https://fr.freepik.com/

# Table des illustrations

[Figure 1 : Système solaire (freepik, -) 1](file:///C:\Users\Mikael.JUILLET\Desktop\TPI-\StellarMap.docx#_Toc133831564)

# Lexique

**Aucune entrée d'index n'a été trouvée.**

# Annexes

## Une image contenant texte, capture d’écran, nombre, Police Description générée automatiquementUne image contenant texte, logiciel, nombre, Page web Description générée automatiquementPlanification initiale

Une image contenant texte, capture d’écran, Police, nombre

Description générée automatiquementUne image contenant texte, capture d’écran, nombre, Police

Description générée automatiquement

## 

Une image contenant texte, diagramme, Caractère coloré, ligne

Description générée automatiquement

## Résumé du rapport du TPI / version succincte de la documentation

## Journal de travail

## Archives du projet

*Media, … dans une fourre en plastique*