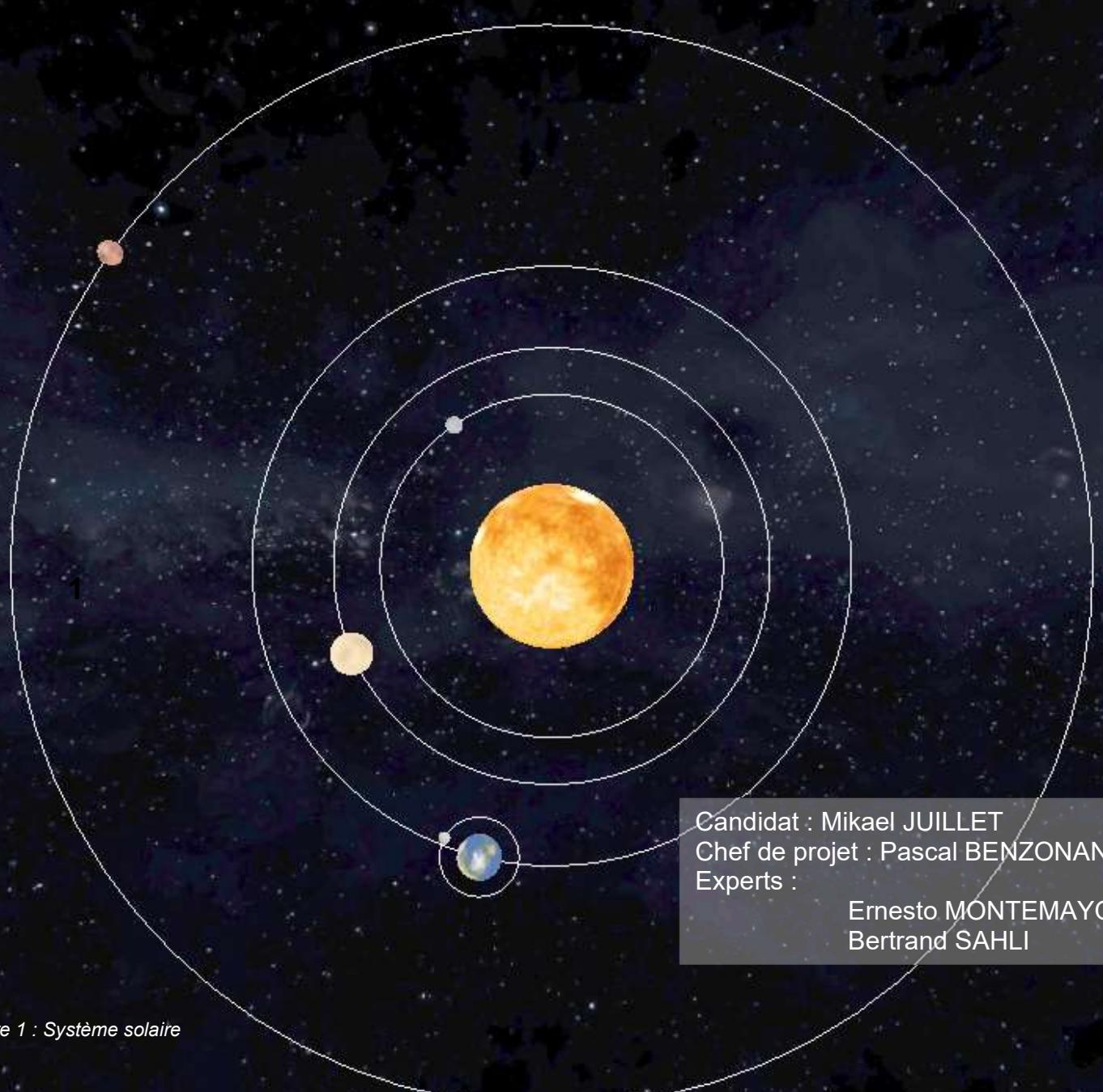


StellarMap

Mikael Juillet



Candidat : Mikael JUILLET
Chef de projet : Pascal BENZONANA
Experts :

Ernesto MONTEMAYOR
Bertrand SAHLI

Figure 1 : Système solaire

Table des matières

1	Table des matières	1
2	Analyse préliminaire	4
2.1	Introduction	4
2.2	Objectifs.....	4
2.3	Planification initiale	5
3	Analyse / Conception.....	6
3.1	Concept	6
3.1.1	Requêtes	6
3.1.2	Wireframe	7
3.1.3	Mockups	8
3.1.4	Diagramme de classe	10
3.1.5	Diagramme de séquences.....	10
3.2	Stratégie de test.....	11
3.3	Risques techniques	11
3.4	Planification	11
3.5	Dossier de conception	13
3.5.1	Logiciels / Framework utilisé	13
4	Réalisation.....	14
4.1	Requêtes API	14
4.1.1	Horizon system	15
4.1.2	Near Earth Objects	21
4.2	3D Affichages et animations	22
4.2.1	Classe Renderer.....	23
4.2.2	Classe PlanetaryCelestialBody.....	25
4.2.3	Classe Planet.....	26
4.2.4	Classe Moon.....	26
4.2.5	Classe Asteroid.....	26
4.2.6	Responsive	28
4.2.7	Mouvement utilisateur.....	28
4.3	Accélération.....	28
4.4	Description.....	29
4.5	Favicon	29
4.6	Déploiement	30
4.7	Répertoires	30
4.8	Description des tests effectués	31
4.9	Erreurs restantes	37
4.10	Liste des documents fournis	37
5	Conclusion.....	38
6	Bilan personnel.....	39
7	Résumé	40

8	Bibliographie	41
9	Table des illustrations	43
10	Lexique	44
11	Annexes	46
11.1	Planification initiale	46
11.2	Journal de travail	50
11.3	Journal de bord	54
11.4	Cahier des charges	55
11.5	Archives du projet	59

2 Analyse préliminaire

2.1 Introduction

Ce projet est réalisé dans le cadre du 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 92 heures, entre le 2 mai de 8h50 au 2 juin à 16h20. Il sera suivi d'une semaine de préparation à la présentation dans lequel le candidat se prépare à présenter son projet devant deux experts et le chef de projet. Cette présentation s'effectuera le 12 juin à 13h50 en salle C315 au centre professionnel de Sainte-Croix.

Lors de ce projet deux experts Monsieur Montemayor et Monsieur Sahli suivront les avancements du projet.

Le sujet sélectionné est la réalisation d'une carte 3D interactive du système solaire, il a été choisi à la suite de la proposition de ce sujet par le candidat et validation du chef de projet et des experts.

2.2 Objectifs

L'objectif de ce projet est de créer une carte interactive du système solaire sur laquelle les utilisateurs pourront sans autre naviguer dans le système et découvrir les différentes planètes et leurs satellites. 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 déterminés 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) ainsi que le cahier des charges disponible à la [section 11.4](#).

2.3 Planification initiale

La planification initiale se découpe en cinq sprints découpés sur cinq semaines. Le détail de la planification est disponible à [la section 11.1](#).

La planification initiale sous forme de méthodologie Waterfall (Wikipedia, 22), j'ai choisi ce format car c'est cette forme convient le mieux pour la planification initiale de projet, étant donné que ce projet doit respecter des dates de rendu précise il est possible de définir à l'avance le travail à effectuer. Cependant le travail se déroulera en format Agile via l'utilisation de sprint de 7 jours, le format Agile permet de modifier si nécessaire en cours de projet la feature demandée, le logiciel Icescrum est utilisé pour cette planification.

Sur l'image ci-dessous, le projet est prés découpés en semaine et les tâches générales du projet sont près définis. En annexe se trouvent au [point 11.1](#) les détails de cette planification initiale avec les tâches et sous tâches détaillées.

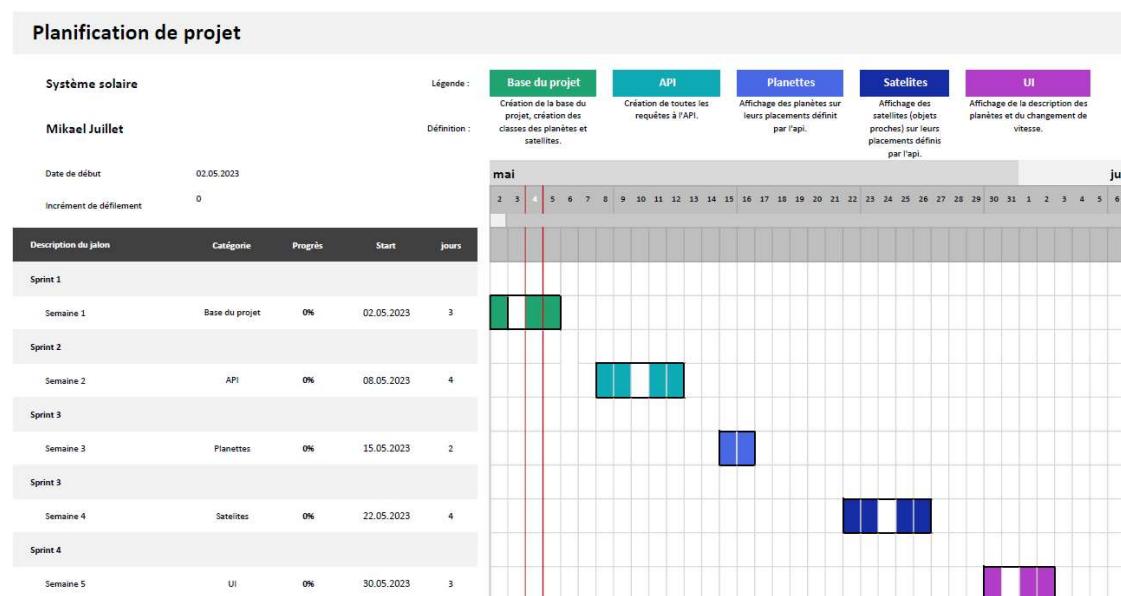


Figure 2 : Planification initiale du projet.

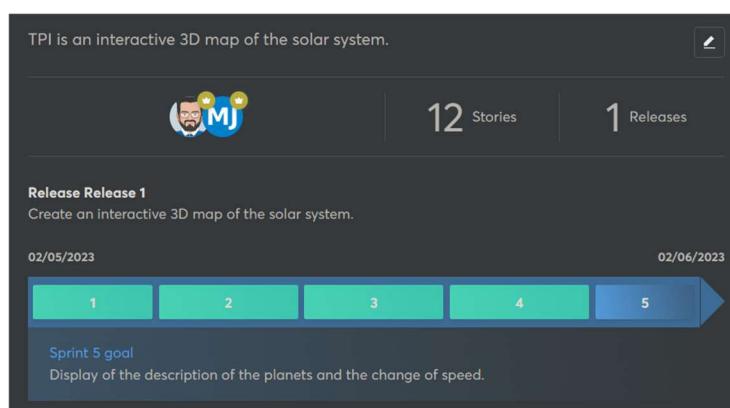


Figure 3 : Format Agile sur IceScrum du projet.

3 Analyse / Conception

3.1 Concept

Ce site est conçu pour afficher un système solaire en 3D, le visiteur peut tourner autour de la carte et ainsi voir les planètes sous d'autres angles, il a 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 ainsi que les astéroïdes à proximité de la planète terre.

Le projet sera hébergé sur la plateforme Swisscenter, les liens relatifs au projet sont les suivantes :

Site web: <https://stellarmap.mycpnv.ch/>

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.xls>

3.1.1 Requêtes

Il est prévu d'utiliser deux API de la Nasa, l'API « [horizon view](#) » 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 actuelle, 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 donnée de placement précise ne peut être récupéré via cette API.

3.1.2 Wireframe

Les Wireframe ont été créés via le logiciel Balsamiq, les maquettes disposent de trois boutons (actuel, 7j/s, 30j/s) ainsi qu'une zone de description des planètes. Le format vertical mobile affichera un message demandant aux utilisateurs de tourner leurs téléphones afin de pouvoir utiliser le site et finalement un écran de chargement s'affichera lors de l'ouverture de l'app en attendant le chargement des différents composants.

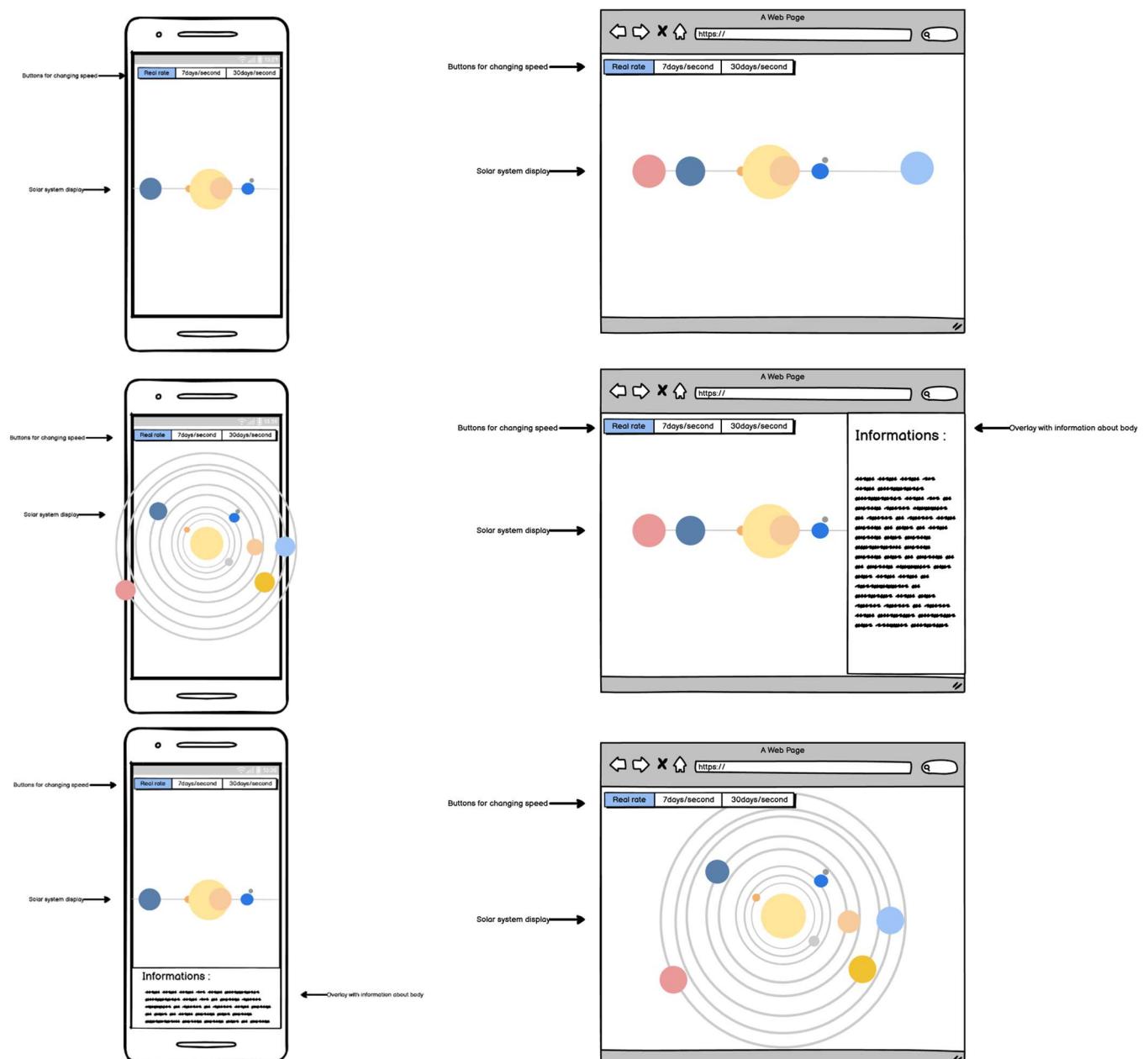


Figure 5 : Wireframe format mobile vertical.

Figure 4 : Wireframe format desktop.

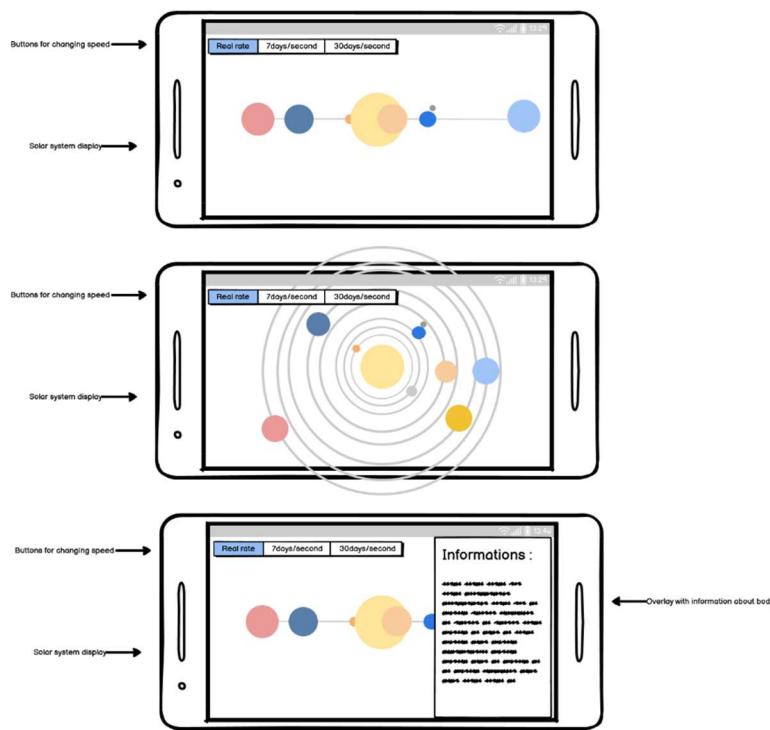


Figure 6 : Wireframe format mobile horizontal.

3.1.3 Mockups

Les Mockups ont été créées à la suite des wireframes depuis le logiciel Figma.

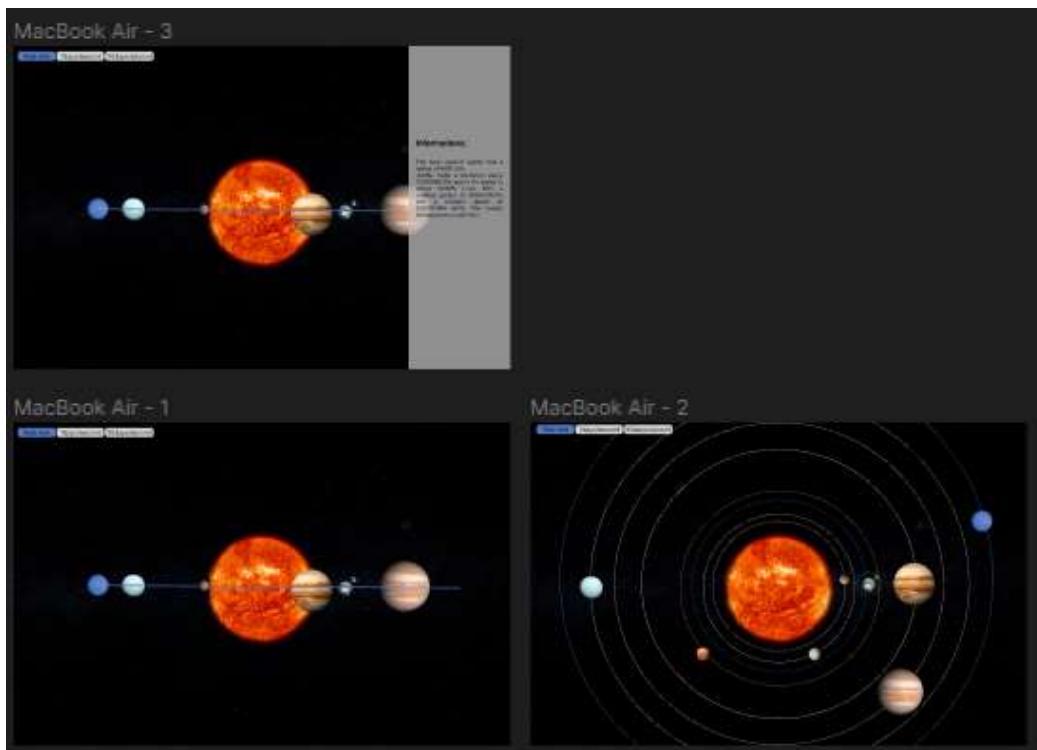


Figure 7 : Mockups format desktop

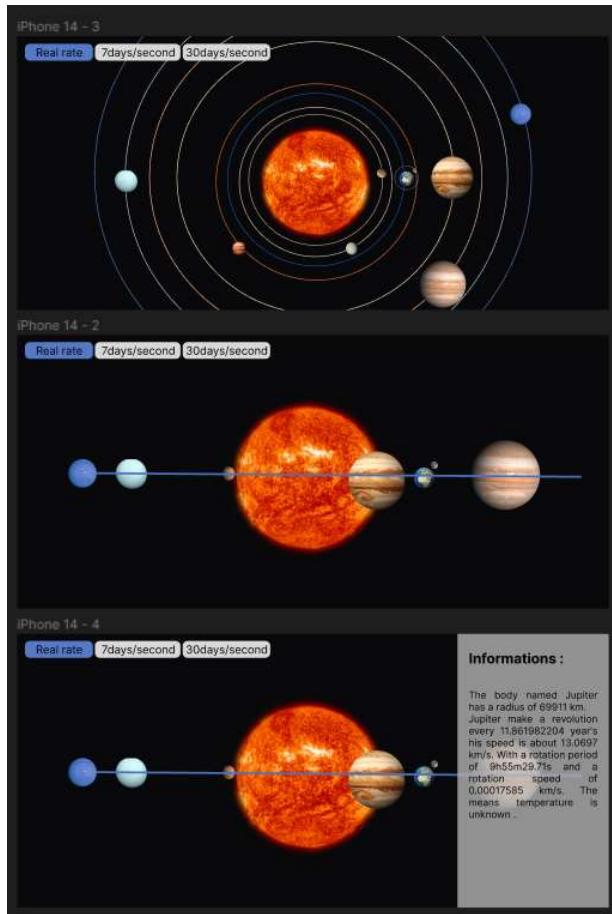


Figure 8 : Mockups format mobile horizontal.

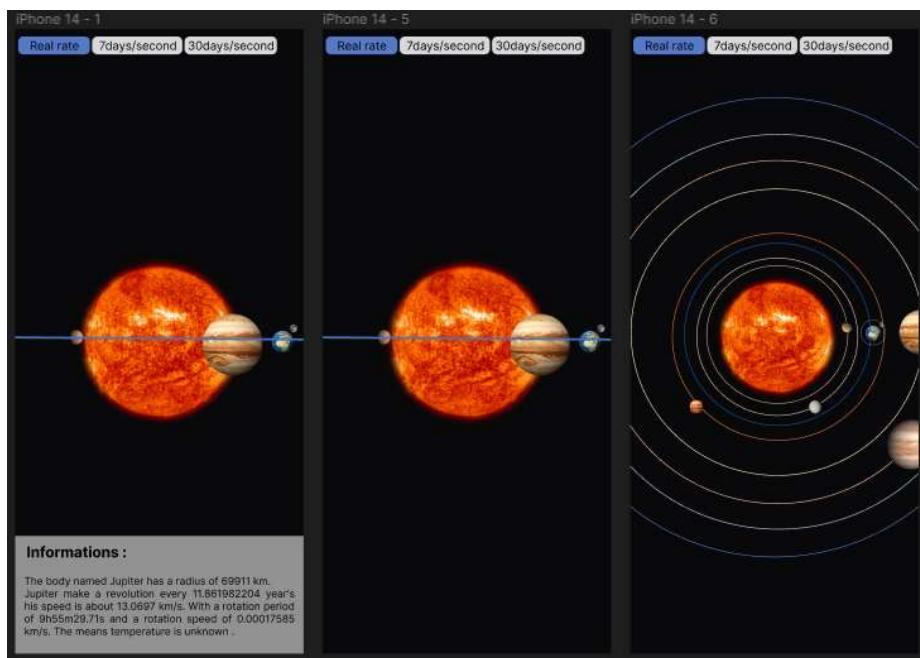


Figure 9 : Mockups format mobile vertical.

3.1.4 Diagramme de classe

Le diagramme de classe représente le contenu ainsi que les interactions entre les différentes classes du projet.

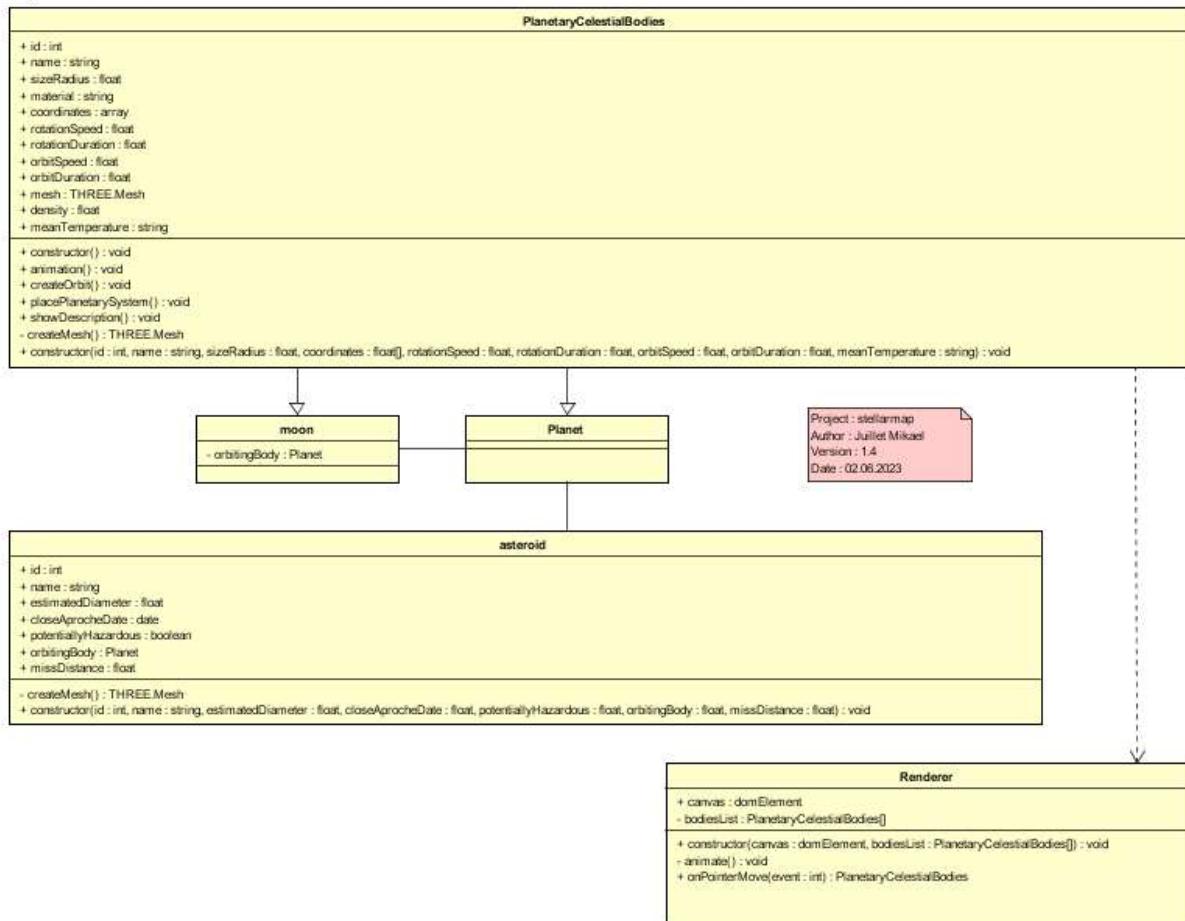


Figure 10 : Diagramme de classe

3.1.5 Diagramme de séquences

Le diagramme de séquence représente un cas nominal du fonctionnement complet de l'app depuis l'arrivée sur site jusqu'à l'affichage de la scène.

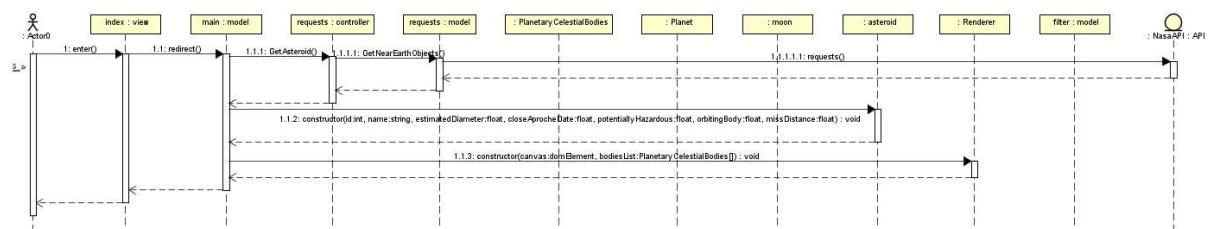


Figure 11 : Diagramme de séquence pour les astéroïdes

3.2 Stratégie de test

Deux types de tests vont être effectués, des tests unitaires sur le filtrage des données et la réception des données et des tests fonctionnels pour la partie visuelle, j'ai choisi de faire cela car je ne dispose que de peu de temps pour créer mon projet de plus j'ai une très faible connaissance de la création de tests en Javascript.

Des tests seront effectués à chaque fin de sprint afin de valider le fonctionnement du site et plus spécifiquement du sprint effectué.

3.3 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 bases acquises grâce à la préparation au TPI, j'ai tout encore besoin de beaucoup me référer aux documentations.

3.4 Planification

Date de début : 2 mai 2023 à 8h50

Date de fin : 2 juin 2023 à 16h20

A la demande des experts le rendu de documentation et du journal de travail est le mardi et vendredi.

La planification finale qui se situe ci-dessous indique les changements de durées en gris en fonction de la planification initiale (heures prévu) et ce qui a été inscrit dans le journal de travail. Les seuls changements se sont effectués sur la durée de certaines tâches, il n'y a pas eu de grand chamboulement dans le projet.

Description	Catégorie	Progrès	Début	Heures effectuée	Heures prévu	Sprint 2					
Sprint 1											
Diagramme de classes	Base du projet	100%	03.05.2023	1.50	0.75	Création d'un contrôleur	API	100%	08.05.2023	0.25	0.25
Diagramme de séquence	Base du projet	100%	03.05.2023	0.75	0.75	Création d'un modèle	API	100%	08.05.2023	0.25	0.25
Création de la classe PlanetaryCelestialBody	Base du projet	100%	03.05.2023	0.75	1.00	Ajout d'un fichier .env	API	100%	08.05.2023	0.25	0.25
Création de la classe satellite	Base du projet	100%	03.05.2023	0.75	1.00	Récupération de la clef API	API	100%	08.05.2023	0.25	0.25
Ajout des opérations dans les classes	Base du projet	100%	05.05.2023	1.50	2.25	Requêtes de récupération des planètes	API	100%	08.05.2023	5.00	3.00
Ajout de vites	Base du projet	100%	05.05.2023	0.25	0.25	Requêtes de récupération des objets proches	API	100%	09.05.2023	3.00	3.00
Création du fichier détenant les codes de planètes	Base du projet	100%	05.05.2023	0.50	0.50	Requêtes de récupération des images	API	0%	11.05.2023	0.00	2.25
Sprint 2											
Sprint 3											
Ajout du déplacement utilisateur	Satellites	100%	22.05.2023	0.50	2.25	Création de maquettes	Planettes	100%	14.05.2023	2.25	0.75
Création des Satellites	Satellites	100%	23.05.2023	1.50	1.50	Ajout de three.js	Planettes	100%	14.05.2023	0.25	0.25
Ajout des lunes	Satellites	100%	25.05.2023	0.75	1.50	Création des planètes	Planettes	100%	14.05.2023	2.75	0.75
Placement sur la carte	Satellites	100%	25.05.2023	1.50	1.50	Placement des planètes	Planettes	100%	15.05.2023	0.75	0.50
Ajout de l'orbite	Satellites	100%	26.05.2023	0.00	1.50	Rotation	Planettes	100%	15.05.2023	0.50	0.50
Résolution de problèmes	Planettes	100%	26.05.2023	2.00	1.50	Orbite autour du soleil	Planettes	100%	15.05.2023	2.00	0.75
Orbite sidérale	Satellites	100%	26.05.2023	0.00	1.50						

Figure 13 : Planification finale sprint1

Sprint 4						
Ajout du déplacement utilisateur	Satellites	100%	22.05.2023	0.50	2.25	
Création des Satellites	Satellites	100%	23.05.2023	1.50	1.50	
Ajout des lunes	Satellites	100%	25.05.2023	0.75	1.50	
Placement sur la carte	Satellites	100%	25.05.2023	1.50	1.50	
Ajout de l'orbite	Satellites	100%	26.05.2023	0.00	1.50	
Résolution de problèmes	Planettes	100%	26.05.2023	2.00	1.50	
Orbite sidérale	Satellites	100%	26.05.2023	0.00	1.50	
Sprint 5						
Création de maquettes	UI	100%	30.05.2023	0.75	0.75	
Placement du canvas en arrière-plan	UI	100%	30.05.2023	0.50	0.75	
Ajout de la description des planètes	UI	100%	01.06.2023	2.00	2.25	
Ecoute d'un clique sur planètes	UI	100%	02.06.2023	0.75	0.75	
Ajout de changement de vitesse	UI	100%	02.06.2023	1.50	1.50	

Figure 14: Planification finale sprint4-5

Figure 12 : Planification finale sprint2-3

3.5 Dossier de conception

3.5.1 Logiciels / Framework utilisé :

Nom	Version	Utilisation
Ajax (W3schools, -)	-	Requêtes
Balsamiq Wireframe (balsamiq, 2023)	4.6.5	Wireframe
Figma (figma, 2023)	-	Mockup
FileZila	3.64.0	Déploiement
HTML, CSS	Html 5, CSS 3	Mise en page
Moment (momentjs, 2023)	2.29.4	Gestion des dates
Postman (postman, 2023)	v10.14	Requêtes
Three.js (WebGL) (three.js, 2023)	0.152.2	Rendu 3D
Visual Studio Code (visualstudio, 2023)	1.74.3	Editeur de code
Vite.js (vitejs, 2023)	4.1.1	Frontend Tooling
Vitest (vitest, 2021)	0.31.0	Tests
Merci app (merci-app, 2023)	-	Correcteur d'orthographe en français
Grammarly (grammarly, 2023)	-	Correcteur d'orthographe en anglais

4 Réalisation

4.1 Requêtes API

Pour ce projet, il est nécessaire de mettre en place deux API de la Nasa, la première s'appelle Horizon et sert à la récupération de données précises sur les objets célestes de notre système solaire. La deuxième API est Near Earth Objets et répertorie tous astéroïdes proches de la terre.

L'ensemble des requêtes s'effectuent sous forme de Get et en Https, il est important d'utiliser le https car elle garantit l'intégrité des données envoyées en chiffrant la requête, garantit l'identité du serveur et renforce la confiance des utilisateurs sur le site.

Il est possible de retrouver les liens des documentations des différentes API ci-dessous :

- **Horizon** : <https://ssd-api.jpl.nasa.gov/doc/horizons.html>
- **Near Earth Objects** (section : Asteroids - NeoWs) : <https://api.nasa.gov/>

L'api Near Earth Objet a besoin d'une clef API, c'est pourquoi j'ai choisi de créer un fichier .env qui contient ma clef API ainsi que les URL respectives des API.

```
VITE_API_URL_HORIZON=https://ssd.jpl.nasa.gov/api/horizons.api
VITE_API_URL_NEO=https://api.nasa.gov/neo/rest/v1/feed
VITE_API_KEY=APIKEY
```

Figure 15 : Contenu du fichier .env

Pour l'ensemble des API je forme de l'url via une fonction formatURL qui va permettre d'assembler mon URL avec les paramètres demandés par l'API.

```
function formatURL(url, parameters) {  
  
    var fullURL = url + "?";  
  
    Object.keys(parameters).forEach( (key, index) => {  
        fullURL = fullURL + key + "=" + parameters[key]  
  
        if (index !== Object.keys(parameters).length - 1) {  
            fullURL = fullURL + "&";  
        }  
  
    });  
  
    return fullURL;  
}
```

Figure 16 : Fonction de formating des URL de requêtes.

4.1.1 Horizon system

Url de requête : https://ssd-api.jpl.nasa.gov/doc/horizons_file.html

Certains paramètres sont absolument obligatoires en voici la liste (Nasa, 2022) :

- **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.

La requête est effectuée via un fetch qui va formater l'url et récupérer soit une réponse soit une erreur.

```

export function GetHorizonSpecificBody(bodyId) {
    return new Promise(resolve => {
        const parameters = {};
        parameters.format = 'json';
        parameters.COMMAND = bodyId;
        parameters.OBJ_DATA = 'YES';
        parameters.MAKE_EPHEM = 'YES';
        parameters.EPHEM_TYPE = 'VECTORS';
        parameters.START_TIME = moment().subtract(1, 'day').format('YYYY-MM-DD');
        parameters.STOP_TIME = moment().format('YYYY-MM-DD');
        parameters.STEP_SIZE = '1d';

        fetch(formatURL(urlHorizon, parameters), {
            headers: {
                'Origin' : 'http://localhost:5173'
            }
        })
        .then(response => {
            resolve(response.json());
        })
        .catch(error => {
            console.error('Error:', error);
            resolve(error);
        });
    });
}

```

Figure 17 : Requête Horizon

Cependant lors ce que la requête est effectuée depuis le navigateur elle engendre une erreur corse. Pour résoudre ce problème, j'ai ajouté dans le header un Origin mais l'erreur ne se résout pas. (Braiam, 2017) (Simplified, 2021)
Je ne parviens pas à récupérer les données pourtant le lien est valide et retourne des données.

Erreur Corse :

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.

Il est possible de récupérer les données via Postman.

Figure 18: Données retournées via Postman.

Mais malgré un retour de requête avec un code d'état de 200 je n'avais toujours aucune donnée.

▼ Général

URL de requête: https://ssd.jpl.nasa.gov/api/horizons.api?format=json&COMMAND=undefined&OBJ_DATA=YES&MAKE_EPHEM=YES&EPHEM_TYPE=IZE=1d
 Mode de requête: GET
 Code d'état: 200 OK
 Règlement sur les URL de provenance: strict-origin-when-cross-origin

▼ En-têtes de réponse

Connection: keep-alive
 Content-Type: application/json
 Date: Fri, 12 May 2023 09:48:57 GMT
 Server: nginx
 Transfer-Encoding: chunked

▼ En-têtes de requête

Afficher la source
 Accept: */*
 Accept-Encoding: gzip, deflate, br
 Accept-Language: en-GB,en;q=0.9,fr-CH;q=0.8,fr;q=0.7,de-DE;q=0.6,de;q=0.5,en-US;q=0.4
 Connection: keep-alive
 Host: ssd.jpl.nasa.gov
 Origin: http://localhost:5173
 Referer: http://localhost:5173/
 sec-ch-ua: "Chromium";v="112", "Google Chrome";v="112", "Not:A-Brand";v="99"
 sec-ch-ua-mobile: ?0
 sec-ch-ua-platform: "Windows"
 Sec-Fetch-Dest: empty
 Sec-Fetch-Mode: cors
 Sec-Fetch-Site: cross-site
 User-Agent: Mozilla/5.0 (Windows NT 10.0; Win64; x64) AppleWebKit/537.36 (KHTML, like Gecko) Chrome/112.0.0.0 Safari/537.36

Figure 19 : Etat de la requête Horizon API sans données.

Échec du chargement des données de la réponse: No data found for resource with given identifier

Figure 20 : Erreur d'affichage des données reçues.

Après en avoir discuté avec mon chef de projet Monsieur Benzonana, nous avons décidé de l'envoi d'un e-mail au support de la Nasa et de l'exportation les données dans un fichier JSON. Ainsi le moins de temps possible n'est perdu. J'ai créé une [issue](#) sur GitHub afin de maintenir le projet à jour.

La réponse de la Nasa a été expéditive (traduction du document original) :

« Les API de la NASA ne peuvent être intégrées à aucun site Web non-NASA. Donc, je crains que vous n'ayez besoin de trouver une autre méthode. C'est la politique de la NASA que nous sommes tenus de suivre. Vous pourrez peut-être appeler l'API à partir d'un script en dehors de votre serveur Web, puis déterminer comment connecter votre application Web à ce script. »

Suite à ce message nous avons décidé de garder le document JSON pour la réalisation de ce projet.

4.1.1.1 Récupération des données

Les données reçues sont sous format texte, je dois récupérer des valeurs précises dans ces tableaux, c'est pourquoi j'utilise des expressions régulières (regex). Celles-ci vont me permettre de récupérer des valeurs en fonction d'un texte dans mon fichier. (mozilla, 2023)

Voici un exemple des données reçues après suppression des espaces, les espaces ont été enlevés car ceux-ci posent des problèmes de correspondance des regex. Pour être plus précis les regex se fient à une expression régulière et prend donc en compte les espaces pour éviter des problèmes de compatibilité entre deux valeurs, il est important d'enlever le maximum de valeurs pouvant interférer.

```
*****Revised:April12,2021Mercury199 dataFilter.js?t=1683886315807:19
PHYSICALDATA(updated2021-Apr-
12):Vol_MeanRadius(km)=2440+1Density(gcm^-3)=5.427Massx10^23(kg)=3.302Volume(x10^10km^3)=6.085Siderealrot.period=58.6463dSid.rot.rate(rad/s)=0
.0000124001Meansolarday=175.9421dCoreradius(km)=~1600GeometricAlbedo=0.106Surfaceemissivity=0.77+-0.06GM(km^3/s^2)=22031.86855Equatorialradius
,Re=2440kmGM1-sigma(km^3/s^2)=Massratio(Sun/plnt)=6023682Mmom.ofInertia=0.33Equ.gravity=s^2=3.701Atmos.pressure(bar)=
<5x10^-15Max.angulardiam.=11.0°MeanTemperature(K)=440Visualmag.(V,1,0)=0.420Obliquitytoorbit[1]=2.11'+'-0.1°Hill'ssphererad.Rp=94.4Siderealorb.p
er.=0.24084671MeanOrbitvel.km/s=47.3625dSiderealorb.per.=87.969257dEscapevel.km/s=4.435PerihelionAphelionMeanSolarConstant(W/m^2)1446262789126Max
imunPlanetaryIR(W/m^2)12700550080000minimumPlanetaryIR(W/m^2)666*****
*****Ephemeris/API_USER!omMay807:4
@:322023Pasadena,USA/Horizons*****Targetbodyname:Mercury(199)
{source:DE441}Centerbodyname:Earth(399){source:DE441}Center-
sitename:BODYCENTER*****Starttime:A.D.2023-May-0500:00.0000TDBStep-
size:1440minutes*****Centergeodetic:0.0,0.0,0.0{E-
lon(deg),Lat(deg),Alt(km)}Centercylindric:0.0,0.0,0.0{E-
lon(deg),Dx(km),Dz(km)}Centerradii:6378.137,6378.137,6356.752km{Equator_a,b,pole_c}Outputunits:KM-
SCalendarmode:MixedJulian/GregorianOutputtype:GEOMETRICCartesianOutputformat:3(position,velocity,LT,range,range-
rate)EOPfile:eop_230504.p230727EOPcoverage:DATA-BASED1962-JAN-20T2023-MAY-04,PREDICTS->2023-JUL-
26Referenceframe:EclipticofJ2000.0*****JDTDBXYZVXYYVZLTRGR*****$5$OE2460068.500000000=A.D.2023-May-
0400:00.00.0000TDBX=6.439857249172552E+07Y=5.346674700502940E+07Z=1.766200542283878E+05VX=5.932433425858534E+00VY=-9.068016481428440E+00VZ=-4.8
94541217796761E+00LT=2.791973394962744E+02RG=8.370125667464858E+07RR=-1.238470929543310E-002460069.50000000=A.D.2023-May-
0500:00.00.0000TDBX=6.496800946265010E+07Y=5.274906761607163E+07Z=-2.463364871165343E+05VX=7.230568569470446E+00VY=-7.535644675529745E+00VZ=-4.
893499593699156E+00LT=2.791468567349237E+02RG=8.368612232353663E+07RR=8.778380713596032E-
01$SEOE*****TMEBarycentricDynamicalTime("TDB"orT_eph)outputwasreques-
ted.Thiscontinuouscoordinatetimeisequivalenttotherelativisticpropertyofaclockatrestinareferenceframeco-
movingwiththesolarsystemcenterbutoutsidethesystem'sgravitywell.Itistheindependentvariableinthesolarsystemrelativisticequationsofmotion.TDBr
unsaturniformrateofone51secpersecondindependentoffirregularitiesinEarth'srotation.CALENDARSYSTEMMixedcalendarmodemwasactivesuchthatcalend-
ardatesafterAD1582-Oct-15(ifany)areinthe modernGregorian system.Datesprior to1582-Oct-
5(ifany)areinthe Juliancalendar system,whichisautomaticallyextendedfordatesprior toitsadoptionon5-Jan-
1BC.The Juliancalendarisusefulfor matchinghistoricaldates.TheGregoriancalendarmoreaccuratelycorrespondstotheEarth'sorbital motion and seasons.A"Greg-
orian-
only"calendarmodeisavailableifsuchphysicaleventsaretheprimaryinterest.REFERENCEFRAMEANDCOORDINATESEclipticatthestandardreferenceepochReference
poch:J2000.0X-Yplane:adoptedEarth orbitalplaneattheresearcherepochNote:IAU76obliquityof84381.448arcseconds wrtICRFX-YplaneX-axis:ICRFZ-
axis:perpendiculartotheX-
Yplaneinthe direction(+or-)senseofEarth'snorthpoleattheresearcherepoch.Symbolmeaning:JDTDBJulianDayNumber,BarycentricDynamicaltimeXX-
componentofpositionvector(km)YY-componentofpositionvector(km)ZZ-componentofpositionvector(km)VXX-componentofvelocityvector(km/sec)VYY-
componentofvelocityvector(km/sec)VZZ-componentofvelocityvector(km/sec)LTOne-waydown-legNewtonianlight-
time(sec)RGRangetime(sec)distancefromcoordinatcenter(km)RRrange-
rate,radialvelocitywrtcoord.center(km/sec)ABERRATIONSANDCORRECTIONSGeometricstatevectorshaveno corrections or aberrations applied.Computationsby...
SolarSystemDynamicsGroup,Horizons-OnlineEphemerisSystem48000akGroveDrive,JetPropulsionLaboratoryPasadena,CA91109USAGeneral site:https://ssd.jpl.nasa.gov/Mailinglist*****$5$OE2460068.500000000=A.D.2023-May-
0400:00.00.0000TDBX=6.439857249172552E+07Y=5.346674700502940E+07Z=1.766200542283878E+05VX=5.932433425858534E+00VY=-9.068016481428440E+00VZ=-4.8
94541217796761E+00LT=2.791973394962744E+02RG=8.370125667464858E+07RR=-1.238470929543310E-002460069.50000000=A.D.2023-May-
0500:00.00.0000TDBX=6.496800946265010E+07Y=5.274906761607163E+07Z=-2.463364871165343E+05VX=7.230568569470446E+00VY=-7.535644675529745E+00VZ=-4.
893499593699156E+00LT=2.791468567349237E+02RG=8.368612232353663E+07RR=8.778380713596032E-
01$SEOE*****TMEBarycentricDynamicalTime("TDB"orT_eph)outputwasreques-
ted.Thiscontinuouscoordinatetimeisequivalenttotherelativisticpropertyofaclockatrestinareferenceframeco-
movingwiththesolarsystemcenterbutoutsidethesystem'sgravitywell.Itistheindependentvariableinthesolarsystemrelativisticequationsofmotion.TDBr
unsaturniformrateofone51secpersecondindependentoffirregularitiesinEarth'srotation.CALENDARSYSTEMMixedcalendarmodemwasactivesuchthatcalend-
ardatesafterAD1582-Oct-15(ifany)areinthe modernGregorian system.Datesprior to1582-Oct-
5(ifany)areinthe Juliancalendar system,whichisautomaticallyextendedfordatesprior toitsadoptionon5-Jan-
1BC.The Juliancalendarisusefulfor matchinghistoricaldates.TheGregoriancalendarmoreaccuratelycorrespondstotheEarth'sorbital motion and seasons.A"Greg-
orian-
only"calendarmodeisavailableifsuchphysicaleventsaretheprimaryinterest.REFERENCEFRAMEANDCOORDINATESEclipticatthestandardreferenceepochReference
poch:J2000.0X-Yplane:adoptedEarth orbitalplaneattheresearcherepochNote:IAU76obliquityof84381.448arcseconds wrtICRFX-YplaneX-axis:ICRFZ-
axis:perpendiculartotheX-
Yplaneinthe direction(+or-)senseofEarth'snorthpoleattheresearcherepoch.Symbolmeaning:JDTDBJulianDayNumber,BarycentricDynamicaltimeXX-
componentofpositionvector(km)YY-componentofpositionvector(km)ZZ-componentofpositionvector(km)VXX-componentofvelocityvector(km/sec)VYY-
componentofvelocityvector(km/sec)VZZ-componentofvelocityvector(km/sec)LTOne-waydown-legNewtonianlight-
time(sec)RGRangetime(sec)distancefromcoordinatcenter(km)RRrange-
rate,radialvelocitywrtcoord.center(km/sec)ABERRATIONSANDCORRECTIONSGeometricstatevectorshaveno corrections or aberrations applied.Computationsby...
SolarSystemDynamicsGroup,Horizons-OnlineEphemerisSystem48000akGroveDrive,JetPropulsionLaboratoryPasadena,CA91109USAGeneral site:https://ssd.jpl.nasa.gov/Mailinglist*****$5$OE2460068.500000000=A.D.2023-May-
0400:00.00.0000TDBX=6.439857249172552E+07Y=5.346674700502940E+07Z=1.766200542283878E+05VX=5.932433425858534E+00VY=-9.068016481428440E+00VZ=-4.8
94541217796761E+00LT=2.791973394962744E+02RG=8.370125667464858E+07RR=-1.238470929543310E-002460069.50000000=A.D.2023-May-
0500:00.00.0000TDBX=6.496800946265010E+07Y=5.274906761607163E+07Z=-2.463364871165343E+05VX=7.230568569470446E+00VY=-7.535644675529745E+00VZ=-4.
893499593699156E+00LT=2.791468567349237E+02RG=8.368612232353663E+07RR=8.778380713596032E-
01$SEOE*****TMEBarycentricDynamicalTime("TDB"orT_eph)outputwasreques-
ted.Thiscontinuouscoordinatetimeisequivalenttotherelativisticpropertyofaclockatrestinareferenceframeco-
movingwiththesolarsystemcenterbutoutsidethesystem'sgravitywell.Itistheindependentvariableinthesolarsystemrelativisticequationsofmotion.TDBr
unsaturniformrateofone51secpersecondindependentoffirregularitiesinEarth'srotation.CALENDARSYSTEMMixedcalendarmodemwasactivesuchthatcalend-
ardatesafterAD1582-Oct-15(ifany)areinthe modernGregorian system.Datesprior to1582-Oct-
5(ifany)areinthe Juliancalendar system,whichisautomaticallyextendedfordatesprior toitsadoptionon5-Jan-
1BC.The Juliancalendarisusefulfor matchinghistoricaldates.TheGregoriancalendarmoreaccuratelycorrespondstotheEarth'sorbital motion and seasons.A"Greg-
orian-
only"calendarmodeisavailableifsuchphysicaleventsaretheprimaryinterest.REFERENCEFRAMEANDCOORDINATESEclipticatthestandardreferenceepochReference
poch:J2000.0X-Yplane:adoptedEarth orbitalplaneattheresearcherepochNote:IAU76obliquityof84381.448arcseconds wrtICRFX-YplaneX-axis:ICRFZ-
axis:perpendiculartotheX-
Yplaneinthe direction(+or-)senseofEarth'snorthpoleattheresearcherepoch.Symbolmeaning:JDTDBJulianDayNumber,BarycentricDynamicaltimeXX-
componentofpositionvector(km)YY-componentofpositionvector(km)ZZ-componentofpositionvector(km)VXX-componentofvelocityvector(km/sec)VYY-
componentofvelocityvector(km/sec)VZZ-componentofvelocityvector(km/sec)LTOne-waydown-legNewtonianlight-
time(sec)RGRangetime(sec)distancefromcoordinatcenter(km)RRrange-
rate,radialvelocitywrtcoord.center(km/sec)ABERRATIONSANDCORRECTIONSGeometricstatevectorshaveno corrections or aberrations applied.Computationsby...
SolarSystemDynamicsGroup,Horizons-OnlineEphemerisSystem48000akGroveDrive,JetPropulsionLaboratoryPasadena,CA91109USAGeneral site:https://ssd.jpl.nasa.gov/Mailinglist*****$5$OE2460068.500000000=A.D.2023-May-
0400:00.00.0000TDBX=6.439857249172552E+07Y=5.346674700502940E+07Z=1.766200542283878E+05VX=5.932433425858534E+00VY=-9.068016481428440E+00VZ=-4.8
94541217796761E+00LT=2.791973394962744E+02RG=8.370125667464858E+07RR=-1.238470929543310E-002460069.50000000=A.D.2023-May-
0500:00.00.0000TDBX=6.496800946265010E+07Y=5.274906761607163E+07Z=-2.463364871165343E+05VX=7.230568569470446E+00VY=-7.535644675529745E+00VZ=-4.
893499593699156E+00LT=2.791468567349237E+02RG=8.368612232353663E+07RR=8.778380713596032E-
01$SEOE*****TMEBarycentricDynamicalTime("TDB"orT_eph)outputwasreques-
ted.Thiscontinuouscoordinatetimeisequivalenttotherelativisticpropertyofaclockatrestinareferenceframeco-
movingwiththesolarsystemcenterbutoutsidethesystem'sgravitywell.Itistheindependentvariableinthesolarsystemrelativisticequationsofmotion.TDBr
unsaturniformrateofone51secpersecondindependentoffirregularitiesinEarth'srotation.CALENDARSYSTEMMixedcalendarmodemwasactivesuchthatcalend-
ardatesafterAD1582-Oct-15(ifany)areinthe modernGregorian system.Datesprior to1582-Oct-
5(ifany)areinthe Juliancalendar system,whichisautomaticallyextendedfordatesprior toitsadoptionon5-Jan-
1BC.The Juliancalendarisusefulfor matchinghistoricaldates.TheGregoriancalendarmoreaccuratelycorrespondstotheEarth'sorbital motion and seasons.A"Greg-
orian-
only"calendarmodeisavailableifsuchphysicaleventsaretheprimaryinterest.REFERENCEFRAMEANDCOORDINATESEclipticatthestandardreferenceepochReference
poch:J2000.0X-Yplane:adoptedEarth orbitalplaneattheresearcherepochNote:IAU76obliquityof84381.448arcseconds wrtICRFX-YplaneX-axis:ICRFZ-
axis:perpendiculartotheX-
Yplaneinthe direction(+or-)senseofEarth'snorthpoleattheresearcherepoch.Symbolmeaning:JDTDBJulianDayNumber,BarycentricDynamicaltimeXX-
componentofpositionvector(km)YY-componentofpositionvector(km)ZZ-componentofpositionvector(km)VXX-componentofvelocityvector(km/sec)VYY-
componentofvelocityvector(km/sec)VZZ-componentofvelocityvector(km/sec)LTOne-waydown-legNewtonianlight-
time(sec)RGRangetime(sec)distancefromcoordinatcenter(km)RRrange-
rate,radialvelocitywrtcoord.center(km/sec)ABERRATIONSANDCORRECTIONSGeometricstatevectorshaveno corrections or aberrations applied.Computationsby...
SolarSystemDynamicsGroup,Horizons-OnlineEphemerisSystem48000akGroveDrive,JetPropulsionLaboratoryPasadena,CA91109USAGeneral site:https://ssd.jpl.nasa.gov/Mailinglist*****$5$OE2460068.500000000=A.D.2023-May-
0400:00.00.0000TDBX=6.439857249172552E+07Y=5.346674700502940E+07Z=1.766200542283878E+05VX=5.932433425858534E+00VY=-9.068016481428440E+00VZ=-4.8
94541217796761E+00LT=2.791973394962744E+02RG=8.370125667464858E+07RR=-1.238470929543310E-002460069.50000000=A.D.2023-May-
0500:00.00.0000TDBX=6.496800946265010E+07Y=5.274906761607163E+07Z=-2.463364871165343E+05VX=7.230568569470446E+00VY=-7.535644675529745E+00VZ=-4.
893499593699156E+00LT=2.791468567349237E+02RG=8.368612232353663E+07RR=8.778380713596032E-
01$SEOE*****TMEBarycentricDynamicalTime("TDB"orT_eph)outputwasreques-
ted.Thiscontinuouscoordinatetimeisequivalenttotherelativisticpropertyofaclockatrestinareferenceframeco-
movingwiththesolarsystemcenterbutoutsidethesystem'sgravitywell.Itistheindependentvariableinthesolarsystemrelativisticequationsofmotion.TDBr
unsaturniformrateofone51secpersecondindependentoffirregularitiesinEarth'srotation.CALENDARSYSTEMMixedcalendarmodemwasactivesuchthatcalend-
ardatesafterAD1582-Oct-15(ifany)areinthe modernGregorian system.Datesprior to1582-Oct-
5(ifany)areinthe Juliancalendar system,whichisautomaticallyextendedfordatesprior toitsadoptionon5-Jan-
1BC.The Juliancalendarisusefulfor matchinghistoricaldates.TheGregoriancalendarmoreaccuratelycorrespondstotheEarth'sorbital motion and seasons.A"Greg-
orian-
only"calendarmodeisavailableifsuchphysicaleventsaretheprimaryinterest.REFERENCEFRAMEANDCOORDINATESEclipticatthestandardreferenceepochReference
poch:J2000.0X-Yplane:adoptedEarth orbitalplaneattheresearcherepochNote:IAU76obliquityof84381.448arcseconds wrtICRFX-YplaneX-axis:ICRFZ-
axis:perpendiculartotheX-
Yplaneinthe direction(+or-)senseofEarth'snorthpoleattheresearcherepoch.Symbolmeaning:JDTDBJulianDayNumber,BarycentricDynamicaltimeXX-
componentofpositionvector(km)YY-componentofpositionvector(km)ZZ-componentofpositionvector(km)VXX-componentofvelocityvector(km/sec)VYY-
componentofvelocityvector(km/sec)VZZ-componentofvelocityvector(km/sec)LTOne-waydown-legNewtonianlight-
time(sec)RGRangetime(sec)distancefromcoordinatcenter(km)RRrange-
rate,radialvelocitywrtcoord.center(km/sec)ABERRATIONSANDCORRECTIONSGeometricstatevectorshaveno corrections or aberrations applied.Computationsby...
SolarSystemDynamicsGroup,Horizons-OnlineEphemerisSystem48000akGroveDrive,JetPropulsionLaboratoryPasadena,CA91109USAGeneral site:https://ssd.jpl.nasa.gov/Mailinglist*****$5$OE2460068.500000000=A.D.2023-May-
0400:00.00.0000TDBX=6.439857249172552E+07Y=5.346674700502940E+07Z=1.766200542283878E+05VX=5.932433425858534E+00VY=-9.068016481428440E+00VZ=-4.8
94541217796761E+00LT=2.791973394962744E+02RG=8.370125667464858E+07RR=-1.238470929543310E-002460069.50000000=A.D.2023-May-
0500:00.00.0000TDBX=6.496800946265010E+07Y=5.274906761607163E+07Z=-2.463364871165343E+05VX=7.230568569470446E+00VY=-7.535644675529745E+00VZ=-4.
893499593699156E+00LT=2.791468567349237E+02RG=8.368612232353663E+07RR=8.778380713596032E-
01$SEOE*****TMEBarycentricDynamicalTime("TDB"orT_eph)outputwasreques-
ted.Thiscontinuouscoordinatetimeisequivalenttotherelativisticpropertyofaclockatrestinareferenceframeco-
movingwiththesolarsystemcenterbutoutsidethesystem'sgravitywell.Itistheindependentvariableinthesolarsystemrelativisticequationsofmotion.TDBr
unsaturniformrateofone51secpersecondindependentoffirregularitiesinEarth'srotation.CALENDARSYSTEMMixedcalendarmodemwasactivesuchthatcalend-
ardatesafterAD1582-Oct-15(ifany)areinthe modernGregorian system.Datesprior to1582-Oct-
5(ifany)areinthe Juliancalendar system,whichisautomaticallyextendedfordatesprior toitsadoptionon5-Jan-
1BC.The Juliancalendarisusefulfor matchinghistoricaldates.TheGregoriancalendarmoreaccuratelycorrespondstotheEarth'sorbital motion and seasons.A"Greg-
orian-
only"calendarmodeisavailableifsuchphysicaleventsaretheprimaryinterest.REFERENCEFRAMEANDCOORDINATESEclipticatthestandardreferenceepochReference
poch:J2000.0X-Yplane:adoptedEarth orbitalplaneattheresearcherepochNote:IAU76obliquityof84381.448arcseconds wrtICRFX-YplaneX-axis:ICRFZ-
axis:perpendiculartotheX-
Yplaneinthe direction(+or-)senseofEarth'snorthpoleattheresearcherepoch.Symbolmeaning:JDTDBJulianDayNumber,BarycentricDynamicaltimeXX-
componentofpositionvector(km)YY-componentofpositionvector(km)ZZ-componentofpositionvector(km)VXX-componentofvelocityvector(km/sec)VYY-
componentofvelocityvector(km/sec)VZZ-componentofvelocityvector(km/sec)LTOne-waydown-legNewtonianlight-
time(sec)RGRangetime(sec)distancefromcoordinatcenter(km)RRrange-
rate,radialvelocitywrtcoord.center(km/sec)ABERRATIONSANDCORRECTIONSGeometricstatevectorshaveno corrections or aberrations applied.Computationsby...
SolarSystemDynamicsGroup,Horizons-OnlineEphemerisSystem48000akGroveDrive,JetPropulsionLaboratoryPasadena,CA91109USAGeneral site:https://ssd.jpl.nasa.gov/Mailinglist*****$5$OE2460068.500000000=A.D.2023-May-
0400:00.00.0000TDBX=6.439857249172552E+07Y=5.346674700502940E+07Z=1.766200542283878E+05VX=5.932433425858534E+00VY=-9.068016481428440E+00VZ=-4.8
94541217796761E+00LT=2.791973394962744E+02RG=8.370125667464858E+07RR=-1.238470929543310E-002460069.50000000=A.D.2023-May-
0500:00.00.0000TDBX=6.496800946265010E+07Y=5.274906761607163E+07Z=-2.463364871165343E+05VX=7.230568569470446E+00VY=-7.535644675529745E+00VZ=-4.
893499593699156E+00LT=2.791468567349237E+02RG=8.368612232353663E+07RR=8.778380713596032E-
01$SEOE*****TMEBarycentricDynamicalTime("TDB"orT_eph)outputwasreques-
ted.Thiscontinuouscoordinatetimeisequivalenttotherelativisticpropertyofaclockatrestinareferenceframeco-
movingwiththesolarsystemcenterbutoutsidethesystem'sgravitywell.Itistheindependentvariableinthesolarsystemrelativisticequationsofmotion.TDBr
unsaturniformrateofone51secpersecondindependentoffirregularitiesinEarth'srotation.CALENDARSYSTEMMixedcalendarmodemwasactivesuchthatcalend-
ardatesafterAD1582-Oct-15(ifany)areinthe modernGregorian system.Datesprior to1582-Oct-
5(ifany)areinthe Juliancalendar system,whichisautomaticallyextendedfordatesprior toitsadoptionon5-Jan-
1BC.The Juliancalendarisusefulfor matchinghistoricaldates.TheGregoriancalendarmoreaccuratelycorrespondstotheEarth'sorbital motion and seasons.A"Greg-
orian-
only"calendarmodeisavailableifsuchphysicaleventsaretheprimaryinterest.REFERENCEFRAMEANDCOORDINATESEclipticatthestandardreferenceepochReference
poch:J2000.0X-Yplane:adoptedEarth orbitalplaneattheresearcherepochNote:IAU76obliquityof84381.448arcseconds wrtICRFX-YplaneX-axis:ICRFZ-
axis:perpendiculartotheX-
Yplaneinthe direction(+or-)senseofEarth'snorthpoleattheresearcherepoch.Symbolmeaning:JDTDBJulianDayNumber,BarycentricDynamicaltimeXX-
componentofpositionvector(km)YY-componentofpositionvector(km)ZZ-componentofpositionvector(km)VXX-componentofvelocityvector(km/sec)VYY-
componentofvelocityvector(km/sec)VZZ-componentofvelocityvector(km/sec)LTOne-waydown-legNewtonianlight-
time(sec)RGRangetime(sec)distancefromcoordinatcenter(km)RRrange-
rate,radialvelocitywrtcoord.center(km/sec)ABERRATIONSANDCORRECTIONSGeometricstatevectorshaveno corrections or aberrations applied.Computationsby...
SolarSystemDynamicsGroup,Horizons-OnlineEphemerisSystem48000akGroveDrive,JetPropulsionLaboratoryPasadena,CA91109USAGeneral site:https://ssd.jpl.nasa.gov/Mailinglist*****$5$OE2460068.500000000=A.D.2023-May-
0400:00.00.0000TDBX=6.439857249172552E+07Y=5.346674700502940E+07Z=1.766200542283878E+05VX=5.932433425858534E+00VY=-9.068016481428440E+00VZ=-4.8
94541217796761E+00LT=2.791973394962744E+02RG=8.370125667464858E+07RR=-1.238470929543310E-002460069.50000000=A.D.2023-May-
0500:00.00.0000TDBX=6.496800946265010E+07Y=5.274906761607163E+07Z=-2.463364871165343E+05VX=7.230568569470446E+00VY=-7.535644675529745E+00VZ=-4.
893499593699156E+00LT=2.791468567349237E+02RG=8.368612232353663E+07RR=8.778380713596032E-
01$SEOE*****TMEBarycentricDynamicalTime("TDB"orT_eph)outputwasreques-
ted.Thiscontinuouscoordinatetimeisequivalenttotherelativisticpropertyofaclockatrestinareferenceframeco-
movingwiththesolarsystemcenterbutoutsidethesystem'sgravitywell.Itistheindependentvariableinthesolarsystemrelativisticequationsofmotion.TDBr
unsaturniformrateofone51secpersecondindependentoffirregularitiesinEarth'srotation.CALENDARSYSTEMMixedcalendarmodemwasactivesuchthatcalend-
ardatesafterAD1582-Oct-15(ifany)areinthe modernGregorian system.Datesprior to1582-Oct-
5(ifany)areinthe Juliancalendar system,whichisautomaticallyextendedfordatesprior toitsadoptionon5-Jan-
1BC.The Juliancalendarisusefulfor matchinghistoricaldates.TheGregoriancalendarmoreaccuratelycorrespondstotheEarth'sorbital motion and seasons.A"Greg-
orian-
only"calendarmodeisavailableifsuchphysicaleventsaretheprimaryinterest.REFERENCEFRAMEANDCOORDINATESEclipticatthestandardreferenceepochReference
poch:J2000.0X-Yplane:adoptedEarth orbitalplaneattheresearcherepochNote:IAU76obliquityof84381.448arcseconds wrtICRFX-YplaneX-axis:ICRFZ-
axis:perpendiculartotheX-
Yplaneinthe direction(+or-)senseofEarth'snorthpoleattheresearcherepoch.Symbolmeaning:JDTDBJulianDayNumber,BarycentricDynamicaltimeXX-
componentofpositionvector(km)YY-componentofpositionvector(km)ZZ-componentofpositionvector(km)VXX-componentofvelocityvector(km/sec)VYY-
componentofvelocityvector(km/sec)VZZ-componentofvelocityvector(km/sec)LTOne-waydown-legNewtonianlight-
time(sec)RGRangetime(sec)distancefromcoordinatcenter(km)RRrange-
rate,radialvelocitywrtcoord.center(km/sec)ABERRATIONSANDCORRECTIONSGeometricstatevectorshaveno corrections or aberrations applied.Computationsby...
SolarSystemDynamicsGroup,Horizons-OnlineEphemerisSystem48000akGroveDrive,JetPropulsionLaboratoryPasadena,CA91109USAGeneral site:https://ssd.jpl.nasa.gov/Mailinglist*****$5$OE2460068.500000000=A.D.2023-May-
0400:00.00.0000TDBX=6.439857249172552E+07Y=5.346674700502940E+07Z=1.766200542283878E+05VX=5.932433425858534E+00VY=-9.068016481428440E+00VZ=-4.8
94541217796761E+00LT=2.791973394962744E+02RG=8.370125667464858E+07RR=-1.238470929543310E-002460069.50000000=A.D.2023-May-
0500:00.00.0000TDBX=6.496800946265010E+07Y=5.274906761607163E+07Z=-2.463364871165343E+05VX=7.230568569470446E+00VY=-7.535644675529745E+00VZ=-4.
893499593699156E+00LT=2.791468567349237E+02RG=8.368612232353663E+07RR=8.778380713596032E-
01$SEOE*****TMEBarycentricDynamicalTime("TDB"orT_eph)outputwasreques-
ted.Thiscontinuouscoordinatetimeisequivalenttotherelativisticpropertyofaclockatrestinareferenceframeco-
movingwiththesolarsystemcenterbutoutsidethesystem'sgravitywell.Itistheindependentvariableinthesolarsystemrelativisticequationsofmotion.TDBr
unsaturniformrateofone51secpersecondindependentoffirregularitiesinEarth'srotation.CALENDARSYSTEMMixedcalendarmodemwasactivesuchthatcalend-
ardatesafterAD1582-Oct-15(ifany)areinthe modernGregorian system.Datesprior to1582-Oct-
5(ifany)areinthe Juliancalendar system,whichisautomaticallyextendedfordatesprior toitsadoptionon5-Jan-
1BC.The Juliancalendarisusefulfor matchinghistoricaldates.TheGregoriancalendarmoreaccuratelycorrespondstotheEarth'sorbital motion and seasons.A"Greg-
orian-
only"calendarmodeisavailableifsuchphysicaleventsaretheprimaryinterest.REFERENCEFRAMEANDCOORDINATESEclipticatthestandardreferenceepochReference
poch:J2000.0X-Yplane:adoptedEarth orbitalplaneattheresearcherepochNote:IAU76obliquityof84381.448arcseconds wrtICRFX-YplaneX-axis:ICRFZ-
axis:perpendiculartotheX-
Yplaneinthe direction(+or-)senseofEarth'snorthpoleattheresearcherepoch.Symbolmeaning:JDTDBJulianDayNumber,BarycentricDynamicaltimeXX-
componentofpositionvector(km)YY-componentofpositionvector(km)ZZ-componentofpositionvector(km)VXX-componentofvelocityvector(km/sec)VYY-
componentofvelocityvector(km/sec)VZZ-componentofvelocityvector(km/sec)LTOne-waydown-legNewtonianlight-
time(sec)RGRangetime(sec)distancefromcoordinatcenter(km)RRrange-
rate,radialvelocitywrtcoord.center(km/sec)ABERRATIONSANDCORRECTIONSGeometricstatevectorshaveno corrections or aberrations applied.Computationsby...
SolarSystemDynamicsGroup,Horizons-OnlineEphemerisSystem48000akGroveDrive,JetPropulsionLaboratoryPasadena,CA91109USAGeneral site:https://ssd.jpl.nasa.gov/Mailinglist*****$5$OE2460068.500000000=A.D.2023-May-
0400:00.00.0000TDBX=6.439857249172552E+07Y=5.346674700502940E+07Z=1.766200542283878E+05VX=5.932433425858534E+00VY=-9.068016481428440E+00VZ=-4.8
94541217796761E+00LT=2.791973394962744E+02RG=8.370125667464858E+07RR=-1.238470929543310E-002460069.50000000=A.D.2023-May-
0500:00.00.0000TDBX=6.496800946265010E+07Y=5.274906761607163E+07Z=-2.463364871165343E+05VX=7.230568569470446E+00VY=-7.53564467
```

Dans le regex nous avons plusieurs modifications possibles du texte :

1. [Mm] cherche la lettre M en majuscule ou minuscule.
2. [Rr] cherche la lettre R en majuscule ou minuscule.

Il existe une multitude d'autres modifications possibles qui permettent de rechercher des données plus ou moins complexes.

```
const regexMeanRadius = /Vol\.[Mm]ean[Rr]adius\((km\)=([\d.]+)/
const matchMeanRadius = dataWithoutSpace.match(regexMeanRadius);

// Return data in object form
return {
  id : matchNameID[2],
  name : matchNameID[1],
  sizeRadius : matchMeanRadius[1],
  coordinate : {
    x : matchPlacement[1],
    y : matchPlacement[2],
    z : matchPlacement[3]
  },
  rotationSpeed : matchRotationRate[1],
  rotationDuration : matchRotationDays[2],
  orbitSpeed : matchOrbitalSpeed[2],
  orbitDuration : matchOrbitPeriod[2],
  obliquity : matchObliquity[2],
  density : matchDensity[1],
  meanTemperature : matchMeanTemp ? matchMeanTemp[2] : "Unknown"
};
```

Figure 22 : Exemple de l'utilisation de regex

4.1.2 Near Earth Objects

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

La requête Near Earth Objects demande peu de paramètres, il suffit de passer dans l'url une date de début, une date de fin et une clé api donnée par la Nasa.

La fonction de requête ci-dessous montre qu'il est effectué un fetch de mon url formatée en passant url et les paramètres, puis la réponse est récupérée et retournée sous format json. Si une erreur apparaît elle sera récupérée via le catch. (developer.mozilla.org, 2022)

```
export function GetNearEarthObjects(apiKey, startDate, endDate) {
  return new Promise(resolve => {
    const parameters = {};
    parameters.start_date = startDate;
    parameters.end_date = endDate;
    parameters.api_key = apiKey;

    fetch(formatURL(urlNeo, parameters))
      .then(response => {
        resolve(response.json())
      })
      .catch(function (err) {
        resolve("Something went wrong!", err);
      });
  });
}
```

Figure 23 : Fonction de création fetch des objets proches.

Voici les données récupérées :

```
▼ {links: {...}, element_count: 25, near_earth_objects: {...}} ⓘ
  element_count: 25
  ▶ links: {next: 'https://api.nasa.gov/neo/rest/v1/feed?start_date=20...&api_key=LTDByOvAiwdRywXsbe4dMf1eJrok44Ip5aZVe0F', previous: 'ht
  ▶ near_earth_objects:
    ▶ 2023-05-11: Array(15)
      ▶ 0:
        absolute_magnitude_h: 22.64
        ▶ close_approach_data: [{}]
        ▶ estimated_diameter: {kilometers: {}, meters: {}, miles: {}, feet: {}}
        id: "2293726"
        is_potentially_hazardous_asteroid: false
        is_sentry_object: false
        ▶ links: {self: 'https://api.nasa.gov/neo/rest/v1/neo/2293726?api_key=LTDByOvAiwdRywXsbe4dMf1eJrok44Ip5aZVe0F'}
        name: "2293726 (2007 RQ17)"
        nasa_jpl_url: "http://ssd.jpl.nasa.gov/sbdb.cgi?str=2293726"
        neo_reference_id: "2293726"
        ▶ [[Prototype]]: Object
      ▶ 1: {links: {}, id: '2467460', neo_reference_id: '2467460', name: '(2006 JF42)', nasa_jpl_url: 'http://ssd.jpl.nasa.gov/
      ▶ 2: {links: {}, id: '3092313', neo_reference_id: '3092313', name: '(2001 QN142)', nasa_jpl_url: 'http://ssd.jpl.nasa.gov/sbdb.c
      ▶ 3: {links: {}, id: '3564040', neo_reference_id: '3564040', name: '(2011 H05)', nasa_jpl_url: 'http://ssd.jpl.nasa.gov/sbdb.cgi
      ▶ 4: {links: {}, id: '3670297', neo_reference_id: '3670297', name: '(2014 JR2)', nasa_jpl_url: 'http://ssd.jpl.nasa.gov/sbdb.cgi
      ▶ 5: {links: {}, id: '3696301', neo_reference_id: '3696301', name: '(2014 W4)', nasa_jpl_url: 'http://ssd.jpl.nasa.gov/sbdb.cgi
      ▶ 6: {links: {}, id: '3742055', neo_reference_id: '3742055', name: '(2016 CB31)', nasa_jpl_url: 'http://ssd.jpl.nasa.gov/sbdb.cgi
      ▶ 7: {links: {}, id: '3841669', neo_reference_id: '3841669', name: '(2019 HF4)', nasa_jpl_url: 'http://ssd.jpl.nasa.gov/sbdb.cgi
      ▶ 8: {links: {}, id: '3878610', neo_reference_id: '3878610', name: '(2019 TQ4)', nasa_jpl_url: 'http://ssd.jpl.nasa.gov/sbdb.cgi
      ▶ 9: {links: {}, id: '3989253', neo_reference_id: '3989253', name: '(2020 BD11)', nasa_jpl_url: 'http://ssd.jpl.nasa.gov/sbdb.cgi
      ▶ 10: {links: {}, id: '54051138', neo_reference_id: '54051138', name: '(2020 QM)', nasa_jpl_url: 'http://ssd.jpl.nasa.gov/sbdb.cgi
      ▶ 11: {links: {}, id: '54087420', neo_reference_id: '54087420', name: '(2020 VL)', nasa_jpl_url: 'http://ssd.jpl.nasa.gov/sbdb.cgi
      ▶ 12: {links: {}, id: '54135432', neo_reference_id: '54135432', name: '(2021 GU3)', nasa_jpl_url: 'http://ssd.jpl.nasa.gov/sbdb.cgi
      ▶ 13: {links: {}, id: '54184284', neo_reference_id: '54184284', name: '(2021 PF7)', nasa_jpl_url: 'http://ssd.jpl.nasa.gov/sbdb.cgi
      ▶ 14: {links: {}, id: '54350904', neo_reference_id: '54350904', name: '(2023 FK2)', nasa_jpl_url: 'http://ssd.jpl.nasa.gov/sbdb.cgi
      length: 15
      ▶ [[Prototype]]: Array(0)
    ▶ 2023-05-12: (10) [{}]
      ▶ [[Prototype]]: Object
      ▶ [[Prototype]]: Object
  ▶ [[Prototype]]: Object
```

Figure 24 : Données récupérées via a l'api NEO.

4.2 3D Affichages et animations

Ce projet utilise la librairie Three.js qui permet d'afficher sur des pages web du contenu en 3D.

Three.js est utilisé car il contient une énorme communauté due à sa grande popularité, il est facile à apprendre, il est compatible avec une large gamme de navigateurs et c'est une librairie que j'ai déjà expérimentée.

Threejs a besoin de 3 choses minimum :

- Une scène
- Une caméra
- Un moteur de rendu

La scène est une sorte de conteneur, elle permet de définir les éléments qui doivent être affichés ainsi que leurs placements. La caméra c'est elle qui va « filmer » ce que qu'il y a dans la scène et le moteur de rendu a pour objectif d'afficher la scène dans le navigateur via WebGL. (Bradley, 2022)

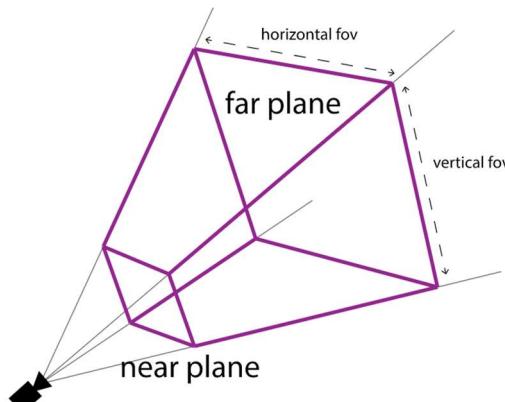


Figure 25 : Exemple de ce qu'est une scène. (Punkasem, 2017)

Il faut aussi ajouter une fonction d'animation, cette fonction crée une boucle qui fera afficher la scène via le moteur de recherche à chaque fois que l'écran est actualisé.

Quand un écran fait 60 images par seconde (frames per second ou FPS) c'est-à-dire qu'il y a 60 images qui sont affichées en une seconde.

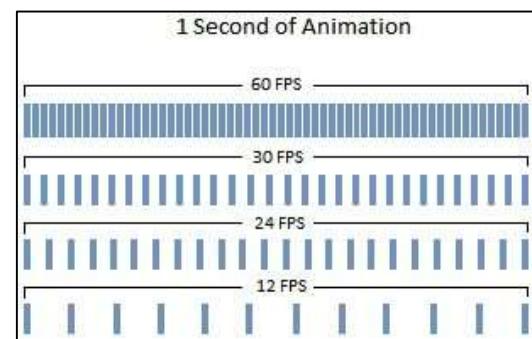


Figure 26 : Schéma d'animation par seconde.

Ce projet utilise programmation orientée objet (POO) qui est répartit en plusieurs classes les éléments de mon projet.

- Renderer
 - Il est utilisé pour initialiser le rendu et géré l'animation.
- PlanetaryCelestialBody
 - Utiliser pour gérer les objets célestes planétaires.
- Planet
 - Hérite de PlanetaryCelestialBody.
- Moon
 - Hérite de PlanetaryCelestialBody.
 - Ajoute un paramètre orbitingBody qui définit planète autour de la quel elle orbite.
- Asteroid
 - Utilisé pour gérer les astéroïdes.

4.2.1 Classe Renderer

La classe Renderer est utilisé pour créer le rendu, il est défini dans celui-ci lors de la construction de la classe qu'il construit la scène, la caméra, la lumière, le renderer, la Skybox, le soleil et les planètes. Il prend comme paramètre un canvas ainsi qu'une liste d'objets à ajouter dans la scène.

Création de la skybox lors de la création de la géométrie, il est défini une valeur de rayon de -20000 afin que les faces de la sphère soient retournée vers l'intérieur. Ce procédé est inspiré de ce site : <https://codinhood.com/post/create-skybox-with-threejs>

```
// Skybox creation
var skyboxGeometry = new THREE.SphereGeometry(-20000, 64, 16); // -20000 for facing interior
const skyboxTexture = new THREE.TextureLoader().load("./src/assets/images/Skybox.jpg");
const skyboxMaterial = new THREE.MeshBasicMaterial({ map: skyboxTexture });
const skybox = new THREE.Mesh( skyboxGeometry, skyboxMaterial );
this.#scene.add(skybox);
```

Figure 27 : Création de la skybox.

La création du soleil est classique, il est important de préciser que la taille du soleil est arbitraire sinon les planètes seront trop petites pour être visualisé c'est ce pourquoi la valeur de sa taille est de 8.

```
// Sun creation
var sunGeometry = new THREE.SphereGeometry(8, 64, 16);
const sunTexture = new THREE.TextureLoader().load("./src/assets/images/Sun.jpg");
const sunMaterial = new THREE.MeshBasicMaterial({ map: sunTexture });
this.#sun = new THREE.Mesh( sunGeometry, sunMaterial );
this.#sun.position.set(0,0,0);
this.#sun.rotation.x = 360;
this.#scene.add(this.#sun);
```

Figure 28 : Création du soleil.

La création des bodies (corps célestes) sont créés via un paramètre définit lors de la création de la classe qui vas boucler autour d'une liste de corps et récupérer le système de chaque corps.

```
// Create bodies
this.#bodiesList.forEach(bodies => {
  const bodiesSystem = bodies.planetarySystem;
  this.#scene.add(bodiesSystem);
  bodies.createOrbit();
});
```

Figure 29 : Création des planètes.

Il dispose d'une méthode animate qui se charge d'ajouter l'animation des objets, la rotation du soleil et de dessiner le rendu.

```
animate() {
  this.#bodiesList.forEach(bodies => {
    bodies.animation();
  });

  this.#sun.rotation.y += 0.00007292115;

  this.#renderer.render( this.#scene, this.#camera );
}
```

Figure 30 : Fonction d'animation du renderer

4.2.2 Classe PlanetaryCelestialBody

La classe PlanetaryCelestialBody est utilisée pour créer les objets célestes planétaires. Elle contient les méthodes de création de mesh, d'animation, de création d'orbite et de placement du système.

Elle prend pour paramètre :

- Un id
- Un nom
- Le radius de la taille
- Un fichier de texture
- Des coordonnées
- Une vitesse de rotation
- Une durée de rotation
- Une vitesse orbitale
- Une durée orbitale
- Une température moyenne

Les corps célestes planétaires définissent tout corps en orbite autour du soleil. Une mesh est objets basés sur un maillage polygonal triangulaire.

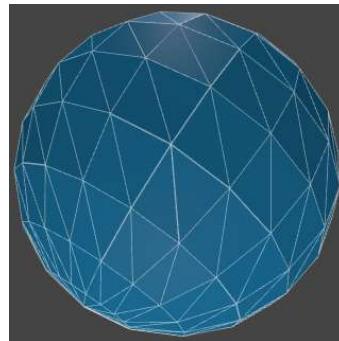


Figure 31 : Exemple de mesh sur une sphère

Cette méthode sert à créer une mesh en fonction de la taille et de la position.

```
#createMesh() {
  const geometry = new THREE.SphereGeometry(this.sizeRadius / 1000, 64, 16 );
  const texture = new THREE.TextureLoader().load("./src/assets/images/" + this.textureFile);
  const material = new THREE.MeshBasicMaterial({ map: texture });
  const mesh = new THREE.Mesh(geometry, material);
  mesh.position.set(parseFloat(this.coordinates.x) / 5000000, parseFloat(this.coordinates.y) / 5000000, 0);
  mesh.rotation.set(3, 3, 0)
  return mesh;
}
```

Figure 32 : Création physique des planètes

La méthode d'animation définit le déplacement du corps sur elle-même ainsi qu'autour de son centre comme le soleil.

Figure 33 : animation des planètes

4.2.3 Classe Planet

La classe `Planet` hérite uniquement de la classe `PlanetaryCelestialBody`.

4.2.4 Classe Moon

La classe `Moon` hérite elle aussi de la classe `PlanetaryCelestialBody` mais elle contient en plus une propriété nommée `orbitingBody` qui permet de définir la planète autour de laquelle elle orbite.

4.2.5 Classe Asteroid

Les astéroïdes n'héritent d'aucunes classes car elle particulière et n'a pas le même comportement que les lunes ou planètes.

En effet la classe Asteroid ne contient pas les mêmes paramètres et n'a pas d'animation dédiée.

Elle prend comme paramètre :

- Un nom
 - Un id
 - Une taille estimée
 - Une date d'approche
 - Si l'astéroïde est dangereux
 - La planète d'orbite
 - La distance manquée

```
constructor(id, name, estimatedDiameter, closeAprocheDate, potentiallyHazardous, orbitingBody, missDistance) {
    this.id = id;
    this.name = name;
    this.estimatedDiameter = estimatedDiameter;
    this.closeAprocheDate = closeAprocheDate;
    this.potentiallyHazardous = potentiallyHazardous;
    this.orbitingBody = orbitingBody;
    this.missDistance = missDistance;
}
```

Figure 34 : Constructeur de la classe Ateroid

Elle contient une seule méthode qui est la création de la mesh, elle définit la taille via le paramètre reçu dans le constructeur et définit la position en calculant la distance avec la position de la planète autour de laquelle orbite.

```
#createMesh() {
    const geometry = new THREE.SphereGeometry(this.estimatedDiameter);
    const texture = new THREE.TextureLoader().load("/src/assets/images/Asteroid.jpg");
    const material = new THREE.MeshBasicMaterial({ map: texture });
    const mesh = new THREE.Mesh(geometry, material);
    mesh.position.set(
        this.orbitingBody.coordinates.x / 5000000 - this.missDistance / 5000000,
        this.orbitingBody.coordinates.y / 5000000 - this.missDistance / 5000000,
        0
    );
    return mesh;
}
```

Figure 35 : Création de la mesh de la classe Asteroid.

4.2.6 Responsive

Le responsive est défini lors de l'utilisation de l'app avec la fonction ci-dessous permet de changer dynamiquement la taille du renderer ce qui évitera d'avoir des bordures blanches non voulues.

```
// Resize render when listen to resize
window.addEventListener('resize', () => {
  this.#camera.aspect = window.innerWidth / window.innerHeight;
  this.#camera.updateProjectionMatrix();
  this.#renderer.setSize(window.innerWidth, window.innerHeight);
  this.animate();
});
```

Figure 36 : Code permettant au redimensionnement du renderer

4.2.7 Mouvement utilisateur

Pour le mouvement de l'utilisateur, j'utilise une fonction de threejs qui permet de définir le déplacement de la caméra.

Documentation threejs : <https://threejs.org/docs/#examples/en/controls/OrbitControls>

```
// Add users controls
const controls = new OrbitControls( this.#camera, this.#renderer.domElement );
controls.enablePan = false
controls.maxDistance = 10000
controls.update();
```

Figure 37 : Code permettant le mouvement de l'utilisateur

4.3 Accélération

L'accélération s'effectue en fonction de trois possibilités, aucune accélération elle est à la vitesse actuelle des planètes, accélération de 7 jours par seconde ce qui indique que 7 jours se passent en une seconde puis 30 jours par seconde.

Pour capter l'accélération, une écoute est faite sur des boutons avec des id spécifiques, une fois cliqué ils modifient une valeur de vitesse qui va multiplier la vitesse actuelle par le nombre correspondant au bouton.

4.4 Description

La description est une zone de texte sur la droite de l'écran qui affiche une petite description de l'objet cliqué.

Pour afficher la description il faut premièrement crée un événement au click de la souri qui créera un raycast et qui permet de récupérer une liste d'objet, que je filtre et recherche dans la liste des objets pour ensuite utiliser une fonction qui va afficher le texte correspondant a l'objet

Le raycast est un lancer de rayon permettant de déterminer quels éléments croise la route de ce rayon.

La technique de raycast est créée en suivant la documentation officielle :

<https://threejs.org/docs/#api/en/core/Raycaster>

```
showDescription(element) {
  var texte =
    "The body named " + this.name + " has a radius of " + this.sizeRadius + " km. <br>" +
    + this.name + " make a revolution every " + this.orbitDuration + " year's " +
    " his speed is about " + this.orbitSpeed + " km/s. With a rotation period of " + this.rotationDuration +
    " and a rotation speed of " + this.rotationSpeed + " km/s. " + " The means temperature is " +
    [this.meanTemperature !== "Unknown" ? (this.meanTemperature - 273.15).toFixed(2) + " °C." : " unknown ."];
  element.innerHTML = texte;
}
```

Figure 38 : Création de la description

4.5 Favicon

J'ai choisi comme favicon une image de soleil du site png art.
(pngarts, -).



Figure 39 : Favicon

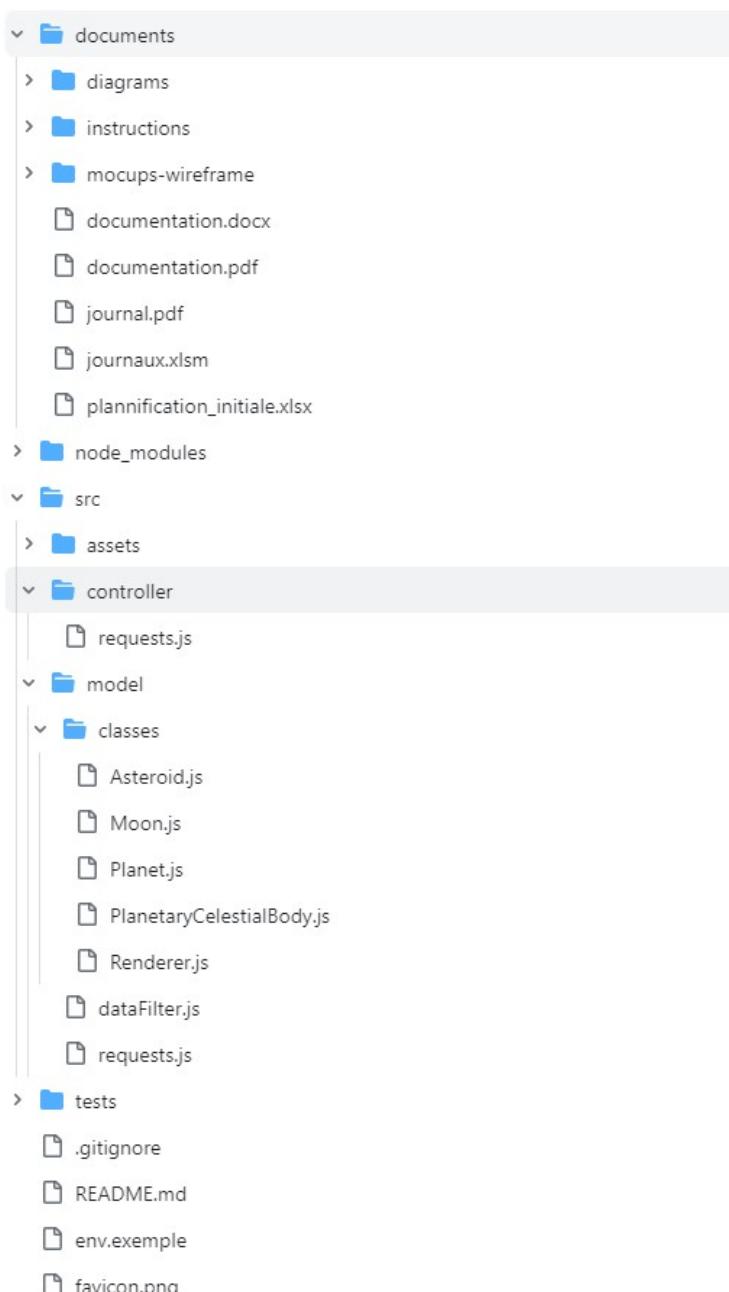
4.6 Déploiement

Le déploiement s'effectue sur Swisscenter, afin de parvenir à la connexion au serveur j'ai créé une clé ssh qui me permet de me connecter de façon sécuriser au serveur via FileZila.

Site web: <https://stellarmap.mycpnv.ch/>

4.7 Répertoires

GitHub contenant les fichiers : <https://github.com/Juillet-Mikael/TPI>



L'architecture des documents est séparé en plusieurs catégories, les documents qui regroupent les diagrammes, les instructions et les documents (journaux, documentations etc.) et la partie code qui contient les documents sources, les tests et autres.

Figure 40 : Architecture des documents sur GitHub.

4.8 Description des tests effectués

Tests effectués sur Chrome et Firefox.

Type	Prés-requis	Action	Attendu	Résultat
API - Horizon	Données format json.	Filtrer les données de Mercure.	<pre>id : "199", name : "Mercury", sizeRadius : "2440", coordinate : { x : "6.439857249172552E+07", y : "5.346674700502940E+07", z : "1.766200542283878E+05" }, rotationSpeed : "0.00000124001", rotationDuration : "58.6463", orbitSpeed : "47.362", orbitDuration : "0.2408467", obliquity : "2.11", density : "5.427", meanTemperature : "440"</pre>	Réussi
API - Horizon	Données format json.	Filtrer les données de Venus.	<pre>id : "299", name : "Venus", sizeRadius : "6051.84", material : "Venus.png", coordinate : { x : "1.073618289428096E+07", y : "1.428930836016723E+08", z : "6.291878178878881E+06" }, rotationSpeed : "-0.00000029924", rotationDuration : "243.018484", orbitSpeed : "35.021", orbitDuration : "0.61519726", obliquity : "177.3", density : "5.204", meanTemperature : "735"</pre>	Réussi

API - Horizon	Données format json.	Filtrer les données de la terre.	<pre>id : "399", name : "Earth", sizeRadius : "6371.01", coordinate : { x : "-1.116613234083490E+08", y : "-9.446394564291658E+07", z : "-4.091379678603614E+07" }, rotationSpeed : "0.00007292115", rotationDuration : "24h", orbitSpeed : "29.79", orbitDuration : "1.0000174", obliquity : "23.4392911", density : "5.51", meanTemperature : "287.6"</pre>	Réussi
API - Horizon	Données format json.	Filtrer les données de Mars.	<pre>id : "499", name : "Mars", sizeRadius : "3389.92", coordinate : { x : "-9.172658128265008E+07", y : "2.475395789325977E+08", z : "7.984203671250358E+06" }, rotationSpeed : "0.0000708822", rotationDuration : "24.622962h", orbitSpeed : "24.13", orbitDuration : "1.88081578", obliquity : "25.19", density : "3.933", meanTemperature : "210"</pre>	Réussi
API - Horizon	Données format json.	Filtrer les données de Jupiter.	<pre>id : "599", name : "Jupiter", sizeRadius : "69911", coordinate : { x : "7.900239544305509E+08", y : "3.977974646088730E+08", z : "-1.643804799922679E+07" },</pre>	Réussi

			rotationSpeed : "0.00017585", rotationDuration : "9h55m29.71s", orbitSpeed : "13.0697", orbitDuration : "11.861982204", obliquity : "3.13", density : "1.3262", meanTemperature : "Unknown"	
API - Horizon	Données format json.	Filtrer les données de Saturne.	<pre>id : "699", name : "Saturn", sizeRadius : "58232", coordinate : { x : "1.377989499063393E+09", y : "-6.338851815853779E+08", z : "-3.764483124414477E+07" }, rotationSpeed : "0.000163785", rotationDuration : "10h39m22.4s", orbitSpeed : "9.68", orbitDuration : "29.447498", obliquity : "26.73", density : "0.687", meanTemperature : "Unknown"</pre>	Réussi
API - Horizon	Données format json.	Filtrer les données de Uranus.	<pre>id : "799", name : "Uranus", sizeRadius : "25362", coordinate : { x : "2.056177409121107E+09", y : "2.306377240773124E+09", z : "-1.704586625654197E+07" }, rotationSpeed : "-0.000101237", rotationDuration : "17.24", orbitSpeed : "6.8", orbitDuration : "84.0120465", obliquity : "97.77", density : "1.271", meanTemperature : "Unknown"</pre>	Réussi

API - Horizon	Données format json.	Filtrer les données de Neptune.	<pre>id : "899", name : "Neptune", sizeRadius : "24624", coordinate : { x : "4.567398114374511E+09", y : "-2.794219386842926E+08", z : "-9.485098077934098E+07" }, rotationSpeed : "0.000108338", rotationDuration : "16.11", orbitSpeed : "5.43", orbitDuration : "164.788501027", obliquity : "28.32", density : "1.638", meanTemperature : "Unknown"</pre>	Réussi
API - Neo	startDate : 2023-05-04 endDate : 2023-05-05 apiKey : Clé api valide	Envois de la requête : https://api.nasa.gov/neo/rest/v1/feed?start_date=2023-05-04&end_date=2023-05-05&api_key=APIKEY	Reçois 20 éléments de la requête.	Réussi
API - Neo	startDate : 2023-05-04 endDate : 2023-05-05 apiKey : Clé api non valide	Envois de la requête : https://api.nasa.gov/neo/rest/v1/feed?start_date=2023-05-04&end_date=2023-05-05&api_key=APIKEY	Un message d'erreur.	Réussi
Planet	Liste de planète créées via la classe Planet.	Affichage des 8 planètes.	Les planètes s'affichent aux bonnes coordonnées	Réussi
Planet	Liste de planète créées via la classe Planet.	Affichage des 8 planètes et mouvement de rotation.	Les planètes tournent sur elle-même à la vitesse calculée.	Réussi
Planet	Liste de planète créées via la classe Planet.	Affichage des 8 planètes et déplacement de l'orbite.	Les planètes tournent autour du soleil à la vitesse calculée.	Réussi
Planet	Liste de planète créées via la classe Planet.	Affichage des 8 planètes et l'orbite.	Les orbites s'affichent et les planètes se déplacent et suivent la courbe de l'orbite dessiné.	Réussi

Moon	Lune crée via la classe Moon.	Affichage de la lune.	La lune s'affiche autour de son corps (terre) et à la coordonnée récupérée.	Réussi
Moon	Lune crée via la classe Moon.	Affichage de la lune et déplacement de l'orbite.	La lune tourne autour du corp (terre) à la vitesse calculée.	Réussi
Moon	Lune crée via la classe Moon.	Affichage de la lune et de l'orbite.	L'orbite s'affiche et la lune se déplace et suit la courbe de l'orbite dessiné.	Réussi
Asteroid	Liste d'astéroïde crée via la classe Asteroid.	Affichage des astéroïdes.	Les astéroïdes s'affichent et sont positionné à la distance calculée.	Réussi
Description	Liste de planète crées via la classe Planet.	Click sur une planète.	Zone de description s'affiche en cliquant sur une planète.	Réussi
Description	Liste de planète crées via la classe Planet.	Click sur une planète.	Afficher la zone de description avec le nom de la planète Terre et les informations que la concerne. <i>Ex : The body named Venus has a radius of 6051.84 km. Venus make a revolution every 0.61519726 year's his speed is about 35.021 km/s. With a rotation period of 243.018484 and a rotation speed of -0.00000029924 km/s. The means temperature is 461.85 °C.</i>	Réussi
Vitesse	Les planètes et lune s'affichent et bougent.	Click sur le bouton « Real rate ».	Les planètes bougent lentement à la vitesse actuelle.	Réussi
Vitesse	Les planètes s'affichent et bougent.	Click sur le bouton « 7 days/second».	Les planètes bougent plus vite.	Réussi
Vitesse	Les planètes s'affichent et bougent.	Click sur le bouton «30 days/second».	Les planètes bougent très vite.	Réussi

Pour effectuer les tests sur les requêtes j'ai utilisé le Framework vitetest qui me permet de créer des tests rapidement et facilement.

Tous les tests pour l'api Horizon sont semblables seules les données reçues sont différentes, cela me permet de tester que le filtrage des données est réussi.

```
test('horizonAPIFilter() should return Jupiter data', () => {
  //Given
  const jupiter = Allplanets.planets[4].result;
  const jupiterExpected = [
    {
      id : "599",
      name : "Jupiter",
      sizeRadius : "69911",
      material : "Jupiter.png",
      coordinate : {
        x : "7.900239544305509E+08",
        y : "3.977974646088730E+08",
        z : "-1.643804799922679E+07"
      },
      rotationSpeed : "0.00017585",
      rotationDuration : "9h55m29.71s",
      orbitSpeed : "13.0697",
      orbitDuration : "11.861982204",
      obliquity : "3.13",
      density : "1.3262",
      meanTemperature : "Unknown"
    };
  //When
  const result = horizonAPIFilter(jupiter);
  //Then
  expect(result).toEqual(jupiterExpected);
})
```

Figure 42 : Exemple de tests sur l'api Horizon.

DEV v0.31.0 C:/Users/Mikael.JUILLET/Desktop/TPI

✓ tests/nearEarthObjectsRequests.test.js (2) 1343ms
 ✓ tests/dataFilter.test.js (9)

Test Files 2 passed (2)
 Tests 11 passed (11)
 Start at 14:23:34
 Duration 2.17s (transform 349ms, setup 0ms, collect 716ms, tests 1.35s, environment 0ms, prepare 275ms)

PASS Waiting for file changes...
 press h to show help, press q to quit

Figure 41: Vision des tests passé.

4.9 Erreurs restantes

Je n'ai pu recenser aucune erreur dans la console, en faisant les tests, cependant j'ai l'impression que mon calcul de vitesse n'est pas à 100% précis, en effet à la suite d'une discussion avec mon professeur de mathématique, Monsieur Delaporte Stéphane, qui m'a expliqué qu'étant donné que la trajectoire de mes planètes est sous forme de cercle et non d'ellipse il y a un décalage de quelques jours des rotations, ce serait quelque chose à approfondir et modifier pour les prochaines features.

4.10 Liste des documents fournis

Les documents fournis sont :

- Documentation du projet daté au 02.06.2023
- Journal de travail daté au 02.06.2023
- Journal de bord daté au 02.06.2023
- Cahier des charges daté au 02.05.2023

5 Conclusion

Tous les objectifs mentionnés au [point 2.2](#) qui sont les suivants ont été atteints :

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 déterminés par l'utilisateur.

Les points positifs du projet sont la qualité de code qui suit les directives POO, la qualité de la documentation ainsi que le respect du temp imparfait pour le projet, la communication avec les différents membres du projet a été constante et sans accros. De plus un écran de chargement a été ajouté afin de palier au confort de l'utilisateur et ne pas être confronté à un écran vide.

Les points négatifs sont le manque de l'accès à une api pour les données dynamiques des planètes, un système de récupération via un script sera effectué pour la suite du projet ainsi que le déplacement des vitesses qui est à vérifier et corriger.

Les difficultés que j'ai rencontrées ont été principalement sur l'api horizon et le filtrage de ses données. Ainsi que sur la vitesse de déplacement, sur lesquels il m'a fallu plusieurs essais pour avoir le bon calcul.

Les améliorations seraient la justesse de la vitesse, l'accès à l'api et l'ajout de toutes les lunes.

Les évolutions possibles que je vois pour ce projet sont l'ajout du déplacement des astéroïdes en affichant leurs trajectoires, comme mentionné dans mon travail il serait intéressant d'étendre le système pour avoir les exoplanètes, ce qui nécessite une adaptation de la taille des planètes pour que l'ensemble des planètes puissent être vus. Le site <https://eyes.nasa.gov/apps/solar-system/#/home> est une bonne source d'inspiration pour la suite du projet.

6 Bilan personnel

Ce projet m'a appris énormément sur la gestion de projet, j'ai pu ajuster et mettre en pratique ce que je n'avais pas fait lors de la préparation au TPI, ce qui m'a permis de pratiquer les points sur lesquels j'ai eu de la peine.

J'ai aussi amélioré ma gestion du temps et du stress, ayant bien étudié le projet au début, j'ai pu créer une planification qui me convient et au cours de celui-ci j'ai remarqué que mes estimations de temps sur le sprint étaient très réalistes.

J'ai aussi découvert que j'avance rapidement si aucun problème ne survient mais que dès qu'une erreur survient j'ai beaucoup de difficulté à trouver la source et résoudre le problème. Ce sera un point à améliorer pour le futur.

J'ai découvert ma capacité à mener un projet de A à Z dans un temps restreint, je suis fière du travail accompli et je suis fière de voir ma progression au cours des quatre dernières années au CPNV.

7 Résumé

Le projet StellarMap a été créé pour l'évaluation de fin d'année au CFC d'informatique au sein d'un travail pratique individuel (TPI) imposé par le canton de Vaud. Le sujet sélectionné a été proposé par le candidat et validé par le chef de projet ainsi que les deux experts. Le sujet est la réalisation d'une carte 3D interactive du système solaire contenant le soleil, les 8 planètes une lune, les astéroïdes et le tracé des orbites. Elle a pour particularité de proposer aux utilisateurs de tourner autour du soleil afin de voir depuis d'autres angles les planètes et de modifier le temps qui secoue en partant de la vitesse actuelle à 7 jours par secondes puis 30 jours par secondes, ce qui permet de visualiser les orbites en fonction du temps des planètes.

Le projet est réalisé sous format web avec les technologies, HTML, CSS, JavaScript, vite et Threejs pour la réalisation 3D. Les conventions d'orientées objet (POO) et MVC ont été utilisé pour la réalisation de ce projet. Le site utilise les données fournies via l'API de la NASA afin de récupérer les données spécifiques des planètes, lunes et astéroïdes. Les tests unitaires réalisés pour les requêtes API sont effectués via Vitetest et des tests manuels sont effectués sur la partie graphique du projet. Le projet est réalisé sous format agile.

Le projet a été publié sur le site <https://stellarmap.mycpnv.ch/> via Swisscenter, le résultat obtenu est conforme aux demandes du client, en plus du système solaire un système d'écran de chargement est mis en place pour que l'utilisateur attende lors du chargement du site. Le site est disponible sur chrome et Firefox. Le site contient les planètes une lune, les astéroïdes et un soleil qui bouge en orbite et en rotation il est aussi affiché le tracé de l'orbite de chaque planète.

8 Bibliographie

- balsamiq. (2023, - -). *balsamiq*. Récupéré sur balsamiq: <https://balsamiq.com/>
- Bradley, S. (2022). *Scene, Camera and Renderer*. Récupéré sur sbcode.net: <https://sbcode.net/threejs/scene-camera-renderer/#:~:text=The%20Renderer%20displays%20the%20scene,2D%20image%20for%20the%20Canvas>.
- Braiam. (2017, Mai 09). *No 'Access-Control-Allow-Origin' header is present on the requested resource—when trying to get data from a REST API*. Récupéré sur stackoverflow: <https://stackoverflow.com/questions/43871637/no-access-control-allow-origin-header-is-present-on-the-requested-resource-whe>
- claude aubry. (21, mai 2018). <https://claudeaubry.fr/post/2018/extraits-du-livre-scrum/>. Paris, Paris, France.
- cpnv.ch. (-, - -). *Icescrum*. Récupéré sur Icescrum.cpnv.ch: <https://icescrum.cpnv.ch/#/>
- day.js. (2023, - -). *day.js*. Récupéré sur day.js: <https://day.js.org/>
- developer.mozilla.org. (2022, Décembre 23). *Utiliser Fetch*. Récupéré sur developer.mozilla.org: https://developer.mozilla.org/fr/docs/Web/API/Fetch_API/Using_Fetch
- figma. (2023, - -). *figma*. Récupéré sur figma: <https://www.figma.com/>
- 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/>
- grammarly. (2023). *grammarly*. Récupéré sur grammarly: <https://app.grammarly.com/>
- kpargeter. (s.d.). Free photo starry night sky. *Free photo starry night sky*. feepick, -. Récupéré sur https://www.freepik.com/free-photo/starry-night-sky_7061153.htm#query=stars&position=2&from_view=search&track=sph
- merci-app. (2023). *merci-app*. Récupéré sur merci-app.com: <https://www.merci-app.com/>
- momentjs. (2023, - -). *momentjs*. Récupéré sur momentjs: <https://momentjs.com/>
- mozilla. (2023, Mai 05). *Regular expressions*. Récupéré sur developer.mozilla.org: https://developer.mozilla.org/en-US/docs/Web/JavaScript/Guide/Regular_expressions
- Nasa. (2022, Septembre 1). *Horizons API*. Récupéré sur ssd-api: <https://ssd-api.jpl.nasa.gov/doc/horizons.html>
- OpenAI. (2023, - -). *Introducing GPT-4, OpenAI's most advanced system*. Récupéré sur Introducing GPT-4, OpenAI's most advanced system: <https://openai.com/pinia>
- pinia. (2023, - -). *pinia*. Récupéré sur pinia: <https://pinia.vuejs.org/>
- pngarts. (-, - -). *Images Transparentes de soleil*. Récupéré sur pngarts: <https://www.pngarts.com/fr/explore/123391>
- postman. (2023). *What is Postman?* Récupéré sur postman: <https://www.postman.com/>
- Punkasem. (2017). *Learning Three.js*. Récupéré sur junethanaon.com: <https://junethanaon.com/blog/blog-threejs.html>
- Simplified, W. D. (2021, Mai 22). *Apprenez CORS en 6 minutes*. Récupéré sur Youtube: <https://www.youtube.com/watch?v=PNtFSVU-YTI>
- three.js. (2023, - -). *three.js*. Récupéré sur three.js: <https://threejs.org/>

visualstudio. (2023, --). *visualstudio*. Récupéré sur visualstudio:
<https://code.visualstudio.com/>

vitejs. (2023, --). *vitejs*. Récupéré sur vite: <https://vitejs.dev/>

vitest. (2021, --). *vitest*. Récupéré sur vitest: <https://vitest.dev/>

vuejs. (2023, --). *vuejs*. Récupéré sur vuejs:
<https://vuejs.org/guide/components/provide-inject.html#prop-drilling>

W3schools. (-, --). *AJAX Introduction*. Récupéré sur W3schools:
https://www.w3schools.com/js/js_ajax_intro.asp

Wikipedia. (22, Février 2023).
https://fr.wikipedia.org/wiki/Mod%C3%A8le_en_cascade. -, -, -.

wrike. (-, --). *wrike*. Récupéré sur wrike.com: <https://www.wrike.com/main/>

9 Table des illustrations

FIGURE 1 : SYSTEME SOLAIRE	1
FIGURE 2 : PLANIFICATION INITIALE DU PROJET.	5
FIGURE 3 : FORMAT AGILE SUR ICESCRUM DU PROJET.	5
FIGURE 5 : WIREFRAME FORMAT MOBILE VERTICAL.	7
FIGURE 4 : WIREFRAME FORMAT DESKTOP.	7
FIGURE 6 : WIREFRAME FORMAT MOBILE HORIZONTAL.	8
FIGURE 7 : MOCKUPS FORMAT DESKTOP	8
FIGURE 8 : MOCKUPS FORMAT MOBILE HORIZONTAL.	9
FIGURE 9 : MOCKUPS FORMAT MOBILE VERTICAL.	9
FIGURE 10 : DIAGRAMME DE CLASSE	10
FIGURE 11 : DIAGRAMME DE SEQUENCE POUR LES ASTEROÏDES	10
FIGURE 12 : PLANIFICATION FINALE SPRINT2-3	12
FIGURE 13 : PLANIFICATION FINALE SPRINT1	12
FIGURE 14: PLANIFICATION FINALE SPRINT4-5	12
FIGURE 15 : CONTENU DU FICHIER .ENV	14
FIGURE 16 : FONCTION DE FORMATING DES URL DE REQUETES.	15
FIGURE 17 : REQUETE HORIZON	16
FIGURE 18: DONNEES RETOURNEES VIA POSTMAN.	17
FIGURE 19 : ETAT DE LA REQUETE HORIZON API SANS DONNEES.	18
FIGURE 20 : ERREUR D'AFFICHAGE DES DONNEES RECUES.	18
FIGURE 21 : DONNEES RECUPEREE APRES SUPPRESSION DES ESPACES.	19
FIGURE 22 : EXEMPLE DE L'UTILISATION DE REGEX	20
FIGURE 23 : FONCTION DE CREATION FETCH DES OBJETS PROCHES.	21
FIGURE 24 : DONNEES RECUPEREES VIA A L'API NEO.	21
FIGURE 25 : EXEMPLE DE CE QU'EST UNE SCENE. (PUNKASEM, 2017)	22
FIGURE 26 : SCHEMA D'ANIMATION PAR SECONDE.	23
FIGURE 27 : CREATION DE LA SKYBOX.	23
FIGURE 28 : CREATION DU SOLEIL.	24
FIGURE 29 : CREATION DES PLANETES.	24
FIGURE 30 : FONCTION D'ANIMATION DU RENDERER	24
FIGURE 31 : EXEMPLE DE MESH SUR UNE SPHERE	25
FIGURE 32 : CREATION PHYSIQUE DES PLANETES	25
FIGURE 33 : ANIMATION DES PLANETES	26
FIGURE 34 : CONSTRUCTEUR DE LA CLASSE ATEROID	27
FIGURE 35 : CREATION DE LA MESH DE LA CLASSE ASTEROID.	27
FIGURE 36 : CODE PERMETTANT AU REDIMENSIONNEMENT DU RENDERER	28
FIGURE 37 : CODE PERMETTANT LE MOUVEMENT DE L'UTILISATEUR	28
FIGURE 38 : CREATION DE LA DESCRIPTION	29
FIGURE 39 : FAVICON	29
FIGURE 40 : ARCHITECTURE DES DOCUMENTS SUR GITHUB.	30
FIGURE 41: VISION DES TESTS PASSE.	36
FIGURE 42 : EXEMPLE DE TESTS SUR L'API HORIZON.	36
FIGURE 43 : JOURNAL DE TRAVAIL 1/4	50
FIGURE 44 : JOURNAL DE TRAVAIL 2/4	51
FIGURE 45 : JOURNAL DE TRAVAIL 3/4	52
FIGURE 46 : JOURNAL DE TRAVAIL CALCULS 4/4	53
FIGURE 47 : JOURNAL DE BORD	54
FIGURE 48 : CAHIER DES CHARGES 1/4	55
FIGURE 49 : CAHIER DES CHARGES 2/4	56
FIGURE 50 : CAHIER DES CHARGES 3/4	57
FIGURE 51 : CAHIER DES CHARGES 4/4	58

10 Lexique

A

API

Permet d'accéder aux fonctions ou aux données d'une application à distance. · 6, 14, 18

C

CFC

Certificat Fédérale de Capacité · 4

E

erreur corse

Produisent lorsqu'un serveur ne renvoie pas les en-têtes HTTP requis par la norme CORS. · 16

exoplanètes

Les exoplanètes sont des planètes situées en dehors du système solaire. · 38

G

Get

Moyen pour vous de récupérer des données à partir d'une source de données à l'aide d'Internet. · 14

H

Https

Protocole de transfert hypertexte est un protocole de communication client-serveur. · 14

I

Illustration en deux dimensions de l'interface d'une page, se concentre spécifiquement sur l'allocation d'espace. · 7 **issue**

Problème, référence les problèmes relatif au projet. · 18

J

JSON

JavaScript Objet Notation est un langage d'échange de données textuelles. · 18

M

mesh

Objets basés sur un maillage polygonal triangulaire. · 25

Mockups

Un modèle de ce à quoi ressemblera votre produit final. · 8

P

POO

Programmation orientée objet · 23

R

regex

Regex ou expressions rationnelles sont des motifs de combinaisons de caractères au sein de chaînes d'un texte. · 19

responsive

Adaptation aux différentes tailles d'écrans. · 28

S

scène

Définit les éléments qui doivent être affiché ainsi que leurs placements. · 22

Skybox

Est un décor projeté sur les parois intérieures d'un cube ou une sphère dont le centre est une caméra. · 23

ssh

Secure Shell est un protocole de communication sécurisé. · 30

T

TPI

Les TPI ou Travail Pratique Individuel est une épreuve exécutée par un candidat lors de la fin de son CFC dans le domaine de l'informatique. · 4, 6, 11, 30, 61

U

Un moteur de rendu

Responsable de l'affichage graphique des objets 3D dans une scène. · 22

Url

Adresse d'un site ou d'une page hypertexte sur Internet. · 14

W

WebGL

Bibliothèque de graphismes pour le Web est une API JavaScript permettant le rendu de graphismes en 2D ou 3D avec de hautes performances, sans avoir à utiliser de plugin. · 22

11 Annexes

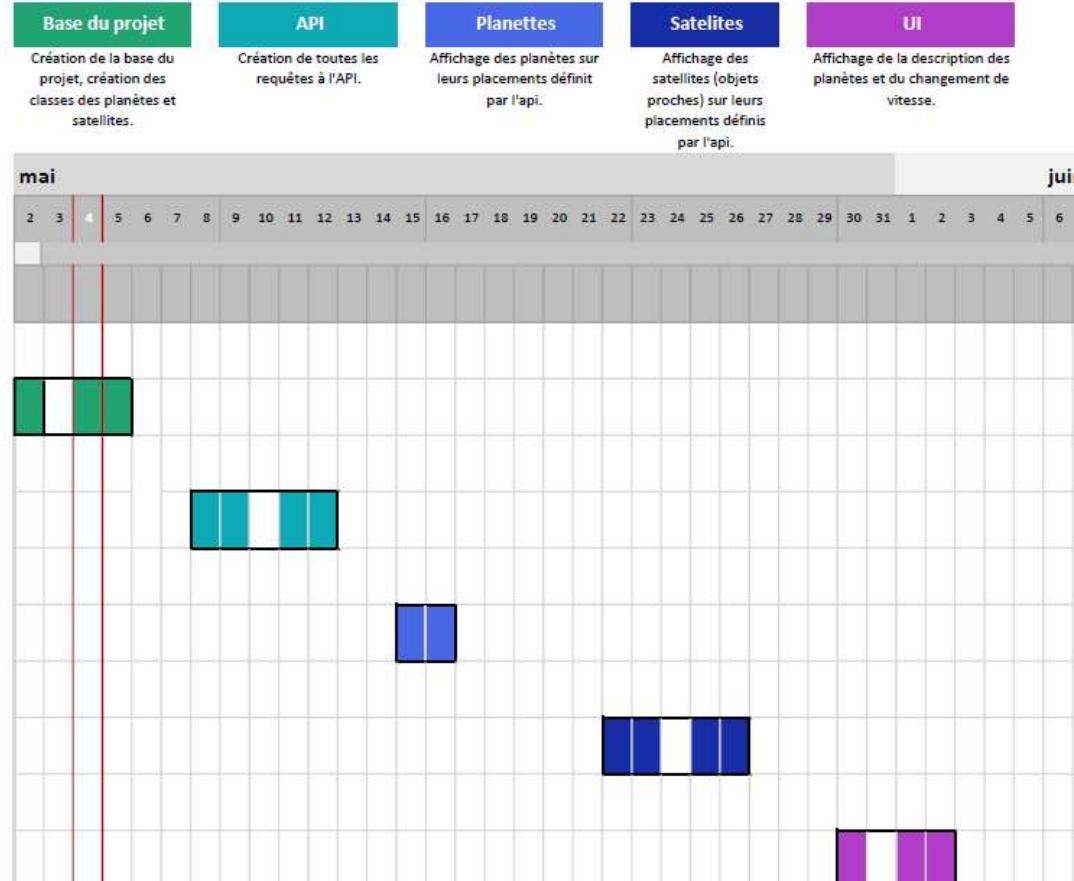
11.1 Planification initiale

Description	Catégorie	Progrès	Début	Heures prévu	Sprint 2				
Sprint 1					Création d'un contrôleur	API	0%	08.05.2023	0.25
Diagramme de classes	Base du projet	0%	03.05.2023	0.75	Création d'un modèle	API	0%	08.05.2023	0.25
Diagramme de séquence	Base du projet	0%	03.05.2023	0.75	Ajout d'un fichier .env	API	0%	08.05.2023	0.25
Création de la classe planète	Base du projet	0%	03.05.2023	1.00	Récupération de la clef API	API	0%	08.05.2023	0.25
Création de la classe satellite	Base du projet	0%	03.05.2023	1.00	Requêtes de récupération des planètes	API	0%	08.05.2023	0.25
Ajout des opérations dans les classes	Base du projet	0%	05.05.2023	2.25	Requêtes de récupération des objets proches	API	0%	08.05.2023	3.00
Ajout de vitesjs	Base du projet	0%	05.05.2023	0.25	Requêtes de récupération des images	API	0%	09.05.2023	3.00
Création du fichier détenant les codes de planètes	Base du projet	0%	05.05.2023	0.50	Récupération des erreurs dans le contrôleur	API	0%	11.05.2023	2.25
					Lien entre la récupération des données et les classes	API	0%	12.05.2023	0.75

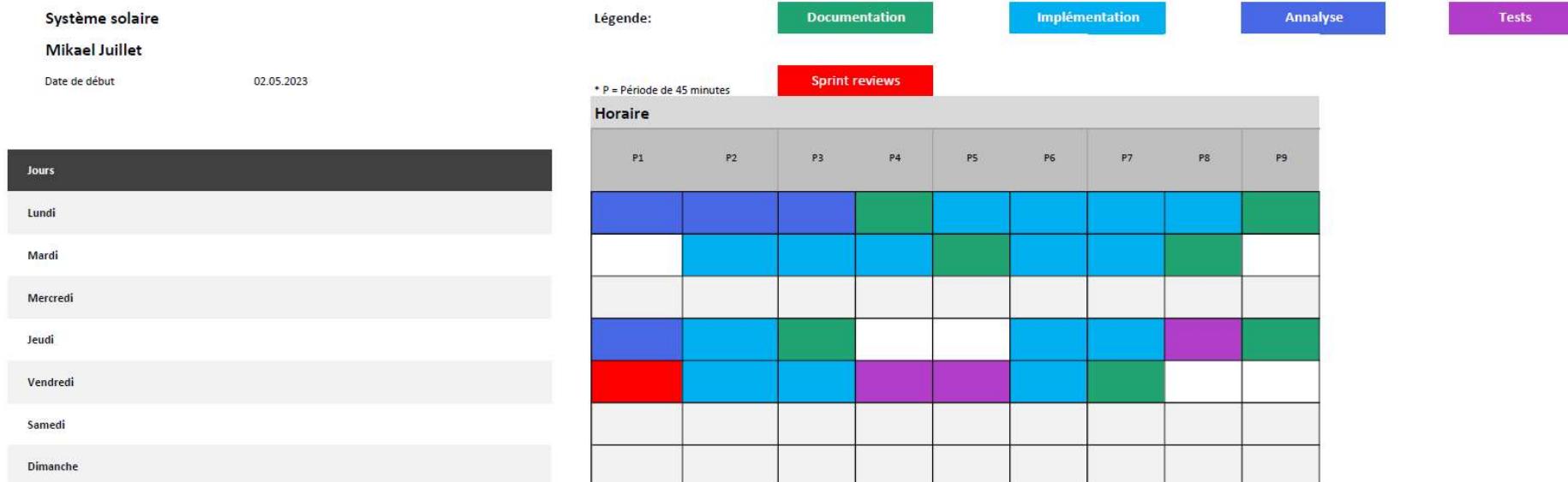
Sprint 3					Sprint 5				
Création de maquettes	Planettes	0%	14.05.2023	0.75	Création de maquettes	UI	0%	30.05.2023	0.75
Ajout de three.js	Planettes	0%	14.05.2023	0.25	Placement du canvas en arrière-plan	UI	0%	30.05.2023	0.75
Création des planètes	Planettes	0%	14.05.2023	0.75	Ajout de la description des planètes	UI	0%	01.06.2023	2.25
Placement des planètes	Planettes	0%	15.05.2023	0.50	Ecoute d'un clique sur planètes	UI	0%	02.06.2023	0.75
Orbite sidérale	Planettes	0%	15.05.2023	0.50	Ajout de changement de vitesse	UI	0%	02.06.2023	1.50
Orbite autour du soleil	Planettes	0%	15.05.2023	0.75					
Sprint 4									
Ajout du déplacement utilisateur	Satellites	0%	22.05.2023	2.25					
Création des Satellites	Satellites	0%	23.05.2023	1.50					
Ajout des lunes	Satellites	0%	25.05.2023	1.50					
Placement sur la carte	Satellites	0%	25.05.2023	1.50					
Ajout de l'orbite	Satellites	0%	26.05.2023	1.50					
Orbite sidérale	Planettes	0%	26.05.2023	1.50					

Planification de projet

Système solaire					Définition :
Mikael Juillet					
Date de début	02.05.2023				
Incrémentation de défilement	0				
Description du jalon	Catégorie	Progrès	Start	jours	
Sprint 1					
Semaine 1	Base du projet	0%	02.05.2023	3	
Sprint 2					
Semaine 2	API	0%	08.05.2023	4	
Sprint 3					
Semaine 3	Planètes	0%	15.05.2023	2	
Sprint 3					
Semaine 4	Satellites	0%	22.05.2023	4	
Sprint 4					
Semaine 5	UI	0%	30.05.2023	3	



Exemple d'un sprint d'une semaine



Note :

Il est important de prendre en compte que la disposition changera car il y a par exemple des semaines avec seulement deux jours mais dans ces deux jours il y aura de l'analyse et des tests même s'ils ne sont pas prévus. C'est un schéma approximatif.

11.2 Journal de travail

Journal de travail

Projet : Travail pratique individuel
 Candidat : Juillet Mikael
 Chef de projet : Benzonana Pascal
 Experts : Sahli Bertrand
 Montemayor Ernesto

Jour	Semaine	Temps [h]	Type	Description	Remarques/problèmes
01.juin	22	0.00	Analyse		
Ressentis du lundi 2 juin				Mon travail s'est bien fini, je suis content du travail accompli et je me réjouis de présenter mon projet.	
01.juin	22	2.75	Implémentation	Finalisation du projet, peaufinage des détails du projet, modification des commentaires et finalisation du CSS.	
02.juin	22	0.75	Meeting	Rendez-vous avec Monsieur Delaporte, pour correction de la vitesse des planètes.	
02.juin	22	3.00	Documentation	Correction de la documentation, ajout des dernières versions des différents diagrammes. Impression des différents documents et préparation du rendu.	
Ressentis du lundi 1 juin				Très bonne journée dans le calme, j'ai bien avancé sur la documentation, il ne reste qu'à ajouter le journal de travail et à corriger l'orthographe et la documentation sera finie.	
01.juin	22	3.00	Documentation	Finalisation de la documentation finalisation des chapitres, ajout des résumés du bilan final et autres points nécessaires.	
Ressentis du lundi 30 mai				Très bonne journée, le travail est bientôt fini, j'ai pu faire mes features, il ne me reste que le polish à effectuer. Je suis content de moi et de mon efficacité du jour.	
30.mai	22	3.00	Documentation	Mise à jour de la documentation, j'ai changé les maquettes et ai relu l'ensemble de la documentation et ai effectué les modifications nécessaires pour préparer le rendu du projet vendredi.	
30.mai	22	0.50	Analyse	Modification des maquettes et analyse de la construction des features.	
30.mai	22	2.00	Implémentation	Implémentation des features de changement de vitesse et de description des planètes.	
Ressentis du lundi 26 mai				Je suis content la journée, a été très productive, j'ai bien avancé sur la documentation et les problèmes de la veille sont résolus et le site prend forme. Je suis très fière de mon travail.	
26.mai	21	1.00	Tests	Test du site afin de s'assurer que tout marche correctement.	
26.mai	21	1.50	Implémentation	Ajout d'une skybox finalisation du sprint et correction des erreurs.	Chaque fois j'ai aidé à trouver l'erreur de placement de ma lune qui avait été initialisé deux fois.
26.mai	21	2.00	Documentation	Ajout de détails sur la documentation sur les classes.	
26.mai	21	1.00	Documentation	Revue de la documentation sur la partie de l'api.	
26.mai	21	1.00	Meeting	Sprint review avec Monsieur Benzonana et validation du sprint 4.	
Ressentis du lundi 25 mai				La journée a été bonne, j'ai bien avancé le projet, il faudra que je rattrape les heures de documentation le lendemain car du au bugs, je n'ai pas eu le temps de faire de la documentation.	
25.mai	21	2.25	Implémentation	Implémentation de l'animation de la lune, j'ai eu quelques problèmes avec le positionnement de la lune.	
25.mai	21	0.75	Tests	J'ai fait un test pour le rendage de la lune, il a fallu que j'effectue quelques modifications pour que tout fonctionne correctement.	
25.mai	21	1.50	Implémentation	J'ai ajouté la lune et l'orbite de la lune.	
25.mai	21	0.92	Implémentation	J'ai ajouté les astéroïdes et je les ai placés afin qu'ils se déplacent avec la planète.	
Ressentis du lundi 23 mai				J'ai bien avancé, j'ai mis à jour le site, je suis satisfait de mon travail du jour.	
23.mai	21	1.00	Implémentation	Ajout des astéroïdes, j'ai commencé à créer des astéroïdes en fonction des données reçues.	
23.mai	21	0.50	Tests	Tests des points nécessaires au bon affichage de la planète et des animations.	
23.mai	21	0.75	Analyse	J'ai analysé et fait quelques schémas sur une feuille du fonctionnement de la mise en place des satellites.	
23.mai	21	1.25	Documentation	Documentation de la partie animation, changement des captures en blanc.	
23.mai	21	0.75	Implémentation	Ajout de commentaires dans le code des classes.	

Figure 43 : Journal de travail 1/4

23.mai	21	2.00	Implémentation	J'ai continué à essayé de résoudre le problème de vitesse orbital en changeant les calculs.	Le problème est moins grand mais la vitesse n'est toujours pas exacte, les planètes du fond sont bien plus lentes que celles de devant, j'ai envoyé un e-mail à mon professeur de math pour savoir s'il sait ou est le problème dans mon calcul.
Ressentis du lundi 22 mai				Les vitesses d'orbites m'ont pris la tête, et m'ont fait perdre beaucoup de temps, ce qui ma contrarié une bonne partie de la journée.	
22.mai	21	4.00	Implémentation	En montrant mon travail à mes camarades ils m'ont fait remarquer que mes planètes avaient beau avoir la bonne vitesse quand on compte le nombre de tour que fait la terre pour q'Uranus en face un seul, les calculs ne sont pas justes. J'ai par conséquent passé plusieurs heures à essayé de résoudre ce problème en cherchant différents calculs	Les planètes les plus en arrières vont plus vite que celles de devant ce qui devrait être le contraire.
22.mai	21	3.00	Implémentation	Je me suis occupé d'ajouter le visuel des orbites autour du soleil, c'est-à-dire le cercle qui correspond à la trajectoire de la planète.	
Ressentis du lundi 16 mai				C'est une bonne journée, j'ai beaucoup peiné sur la partie orbite. Je reste motivé et je suis content d'avoir atteint mon objectif de la semaine.	
16.mai	20	2.25	Implémentation	Finalisation de l'ajout des planètes à leurs tailles ainsi qu'à leurs placements reçus par l'api. Ajout de l'orbite de la planète.	
16.mai	20	0.50	Documentation	Modification de la documentation et du journal de travail. J'ai notamment ajouté les informations relatives aux classes et au rendu.	
16.mai	20	1.75	Implémentation	Implémentation de la nouvelle classe renderer et j'ai modifié le main et la classe celestial bodies pour que tous s'emboîtent parfaitement.	Elliott Jaquier, m'a aidé pour la partie création des classes, il m'a aiguillé sur les bonnes pratiques et le fonctionnement des classes.
16.mai	20	0.25	Analyse	Modification du diagramme de classe pour ajouter une classe renderer.	Elliott Jaquier, m'a aidé pour la partie création de la classe renderer, il m'a confirmé que le diagramme de classe pour la classe renderer était bien effectué.
16.mai	20	0.75	Configuration	Tentative d'accès en ssh au serveur de production. J'ai créé la clé ssh et ai tenté d'accéder au serveur mais swisscenter est down. https://stellarmap.mycpnv.ch/	
Ressentis du lundi 15 mai				Une bonne journée, sans trop de stress je suis parfaitement dans le planning, la visite de Monsieur Sahli m'a motivé encore plus sur ce projet. Je suis content de cette journée.	
15.mai	20	0.25	Documentation	Ajout de détails sur la vue dans la documentation.	
15.mai	20	2.75	Implémentation	Ajout des planètes sur la vue, j'ai ajouté threejs et un renderer puis j'ai fait quelques modifications dans les classes pour que cela marche, quelques problèmes de relation avec threejs que j'ai réglé en vérifiant le code.	
15.mai	20	1.00	Implémentation	Début de l'implémentation de la vue avec three.js.	
15.mai	20	2.25	Analyse	Création de maquettes et wireframe. J'ai dû rechercher des images pour les mettre dans les maquettes, c'est ce qui m'a pris plus de temps. J'ai recensé plusieurs sites pour lesquels j'ai récupéré les images: https://www.stickpng.com , www.pngarts.com et iconscout.com . Les liens de chaque image à été sauvegardé et a été ajoutée dans la documentation.	
15.mai	20	1.00	Documentation	Modification de la documentation ajout notamment des liens vers les images ainsi que l'ajout des maquettes.	
15.mai	20	0.50	Meeting	Rendez-vous avec Monsieur Sahli pour la visite de stage, nous avons discuté du travail en cours des problèmes et des solutions, ainsi que les rendus de documents.	
Ressentis semaine n°2				J'ai eu quelques difficultés sur les requêtes avec l'API Horizon de la Nasa, je n'arrivais pas à faire mes requêtes et à cause d'une erreur cors j'ai dû exporter les données ce qui m'a fait perdre du temps.	
12.mai	19	0.75	Meeting	Sprint review avec Monsieur Benzomana et validation du sprint 3.	
12.mai	19	2.50	Documentation	Modification de la documentation, mise à jour des images, ajout de détails sur les api et ajout du favicon.	
12.mai	19	2.25	Tests	Finalisation des tests pour horizon et création de tests pour la requête de récupération des objets proches.	
11.mai	19	3.00	Implémentation	Création du filtrage des données, mise à jour des regex, j'ai eu des complications avec les regex en créant les regex j'ai découvert à ma grande surprise que les données étaient toutes différentes. J'ai perdu pas mal de temps sur l'implémentation des regex afin qu'ils marchent pour toutes les planètes.	
11.mai	19	3.00	Tests	Implémentation de vites, mise en place des tests création des premiers tests pour horizon.	
09.mai	19	0.75	Implémentation	Continuation de la création des regex.	
09.mai	19	1.00	Analyse	Ajout de sources dans la documentation et écriture de mail pour la Nasa et pour les experts afin de rendre mon travail.	
09.mai	19	0.75	Documentation	Documentation du code, plus particulièrement du problème avec l'api et de la filtration des requêtes.	
09.mai	19	1.75	Implémentation	J'ai revu mon code et j'ai commencé à ajouter le code de filtrage des données avec les expressions régulières. Je me suis aidé du site suivant : https://developer.mozilla.org/en-US/docs/Web/JavaScript/Guide/Regular_expressions	

Figure 44 : Journal de travail 2/4

09.mai	19	1.00	Implémentation	J'ai récupéré toutes les données des planètes et lunes que j'ai mises dans un fichier afin de contourner le problème de l'API horizon pour l'instant.	
08.mai	19	1.30	Analyse	J'ai ajouté la requête pour la récupération des objets proches.	
08.mai	19	4.25	Analyse	J'ai ajouté la requête api pour récupérer toutes les données de la Nasa sur chaque planète, hors j'ai eu une erreur cors policy que j'ai corrigée, cependant ma requête n'a rien retourné. C'est-à-dire que je ne peux pas récupérer les données. J'ai demandé de l'aide à Monsieur Benzonana, qui m'a conseillé d'exporter les données en attendant d'avoir l'avis du support informatique de la Nasa.	Je suis très frustré d'avoir perdu du temps et j'espère trouver la solution rapidement.
08.mai	19	0.45	Analyse	Mise à jour de la documentation, ajout des requêtes api.	
Ressentis semaine n°1					
05.mai	18	1.29	Documentation	Modification de la documentation, ajout des l'analyse et des tests.	
05.mai	18	1.75	Analyse	Création des sprints et des tests dans icescrum, la description de chaque partie et la réalisation de tests sont détaillés.	
05.mai	18	0.75	Meeting	Review de Sprint avec Monsieur Benzonana, nous avons discuté du sprint actuel ainsi que du prochain et des difficultés de création du diagramme de classe.	
05.mai	18	1.50	Implémentation	Modification des Classes ajout des calculs de positions et de rotations.	Quelques problèmes au niveau de la création des classes. J'ai quelques difficultés à savoir quels sont les éléments nécessaires et comment les implémenter avec les flèches. Le site suivant m'a aidé : https://developer.ibm.com/articles/the-class-diagram/
04.mai	18	0.75	Analyse	Création de la documentation, ajout de l'introduction.	
04.mai	18	1.31	Implémentation	Ajout de vite et création des classes en suivant le diagramme de classe créé précédemment.	
04.mai	18	0.50	Documentation	Création du diagramme de séquence initiale.	
04.mai	18	1.50	Documentation	Création du diagramme de classe, ajout de la classe PlanetaryCelestialBody, planet, moon et asteroid.	
04.mai	18	0.50	Analyse	Creation du sprint 1, ajout des stories, création des tâches et des tests.	
04.mai	18	0.40	Analyse	Emails sur la planification initiale ainsi que sur mon absence de 12 mai.	
04.mai	18	0.25	Analyse	Modification des erreurs de dates de la planification initiale, suite à l'e-mail de Monsieur Sahli.	Cette tâche as été effectuée en dehors du temps de travail, sous nécessité d'un envoi de document important.
04.mai	18	0.50	Analyse	Création du github et de l'icescrum et des dossiers.	
02.mai	18	1.50	Analyse	finalisation de la planification initiale, création des email d'envoie sur la planification initiale et création d'un e-mail à Monsieur Benzonana questionnant différents points du projet.	
02.mai	18	2.50	Analyse	Création de la planification initiale, je suis allé dans les détails de la planification en expliquant le détails d'une semaine type en plus d'un diagramme de gantt et d'une liste des points.	
02.mai	18	0.50	Meeting	Rendez-vous avec Monsieur Montemayor Ernesto afin de discuter du cahier des charges et de certains points sur ce dernier. Nous avons notamment abordé les points relatifs aux rendus et de présentation.	
02.mai	18	0.33	Analyse	Récupération du cahier des charges, lecture et annotation de cette dernière.	

Figure 45 : Journal de travail 3/4

Calculs des données

Projet	Travail pratique individuel
Candidat	Juillet Mikael
Chef de projet	Benzonana Pascal
Experts	Sahli Bertrand Montemayor Ernesto
Heures attendues	92
Heures totales	92.00
Pourcentage effectué	100.00

Type de tâche	Heures	Pourcentage	Attendu
Test	7.5	8.15	10%
Annalyse	19.233333	20.91	20%
Meeting	4.25	4.62	-
implementation	38.73	42.10	45%
Documentation	21.54	23.41	25%
Configuration	0.75	0.82	-
Total	100.00		

Figure 46 : Journal de travail calculs 4/4

11.3 Journal de bord

Journal de bord

Projet Travail pratique individuel
Candidat Juillet Mikael
Chef de projet Benzonana Pascal
Experts Sahli Bertrand
 Montemayor Ernesto



Jour	Semaine	Temps [h]	Type	Ajouter une ligne	Description	Remarques/problèmes
02.mai	18	0.00	Analyse
15.mai	20	0.50	Meeting		Rendez-vous avec Monsieur Sahli pour la visite de stage, nous avons discuté du travail en cours des problèmes et des solutions, ainsi que les rendus de documents.	
08.mai	19	0.50	Meeting		J'ai regardé avec Monsieur Benzonana, le problème de requête API avec Horizon, nous avons essayé de débloquer la situation, finalement nous avons convenu d'écrire un e-mail à la Nasa puis d'exporter toutes les données afin de ne pas perdre du temps sur le projet.	
02.mai	18	0.50	Meeting		Rendez-vous avec Monsieur Montemayor Ernesto afin de discuter du cahier des charges et de certains points sur ce dernier. Nous avons notamment abordé les points relatifs aux rendus et de présentation.	

Figure 47 : Journal de bord

11.4 Cahier des charges

ANNEXE 1	Cahier des charges		TPI
1 INFORMATIONS GENERALES			
Candidat:	Nom: JUILLET	Prénom: Mikael	
	✉: mikael.jUILLET@cpnv.ch		
Lieu de travail:	CPNV, Sainte-Croix		
Orientation :	<input type="checkbox"/> 88601 Développement d'applications <input checked="" type="checkbox"/> 88602 Informatique d'entreprise <input type="checkbox"/> 88603 Technique des systèmes		
Chef de projet:	Nom: BENZONANA	Prénom: Pascal	
	✉: pascal.benzonana@cpnv.ch		
Expert 1:	Nom: MONTEMAYOR	Prénom: Ernesto	
	✉: ernesto@bati-technologie.ch		
Expert 2:	Nom: SAHLI	Prénom: Bertrand	
	✉: bertrand.sahli@eduvaud.ch		
Période de réalisation :	Du mardi 2 mai 2023 à 8h50 au vendredi 2 juin 2023 à 14h05		
Horaire de travail :	Lundi	08h00-12h15	13h30-16h55
	Mardi	08h50-12h15	13h30-16h05
	Mercredi	-	-
	Jeudi	08h00-10h35	13h30-16h55
	Vendredi	08h00-12h15	13h30-15h05
	Pentecôte 29 mai Ascension 18 mai Examen ECG 1er juin matin Pont de l'Ascension 19 mai Toutes les demi-journées ont une pause obligatoire de 15 minutes, sauf si elles commencent à 9h50 ou si elles se terminent à 15h05.		
Voir en annexe en fin de document			
Nombre d'heures :	90 heures		
Planning	Analyse : 20%		
	Implémentation : 45%		
	Tests : 10 %		
	Documentations : 25%		

2 PROCÉDURE

- Le candidat réalise un travail personnel sur la base d'un cahier des charges reçu le 1er jour.
- Le cahier des charges est approuvé par la i-CQ.VD. Il est en outre présenté, commenté et discuté avec le candidat. Par sa signature, le candidat accepte le travail proposé.
- Le candidat a connaissance de la feuille d'appréciation avant de débuter le travail.
- Le candidat est entièrement responsable de la sécurité de ses données.
- En cas de problèmes graves, le candidat avertit au plus vite les deux experts et son chef de projet.
- Le candidat a la possibilité d'obtenir de l'aide, mais doit le mentionner dans son dossier de projet.
- A la fin du délai imparti pour la réalisation du TPI, le candidat doit transmettre par courrier électronique le dossier de projet aux deux experts et au chef de projet. En parallèle, une copie papier du rapport doit être fournie sans délai en trois exemplaires. Cette dernière doit être en tout point identique à la version électronique.

Figure 48 : Cahier des charges 1/4

ANNEXE 1	Cahier des charges	TPI
----------	--------------------	-----

3 TITRE

Carte 3D interactive du système solaire

SUJET

Le projet consiste à créer une carte 3D du système solaire. Les utilisateurs pourront sans autre naviguer dans le système et découvrir les différentes planètes et leurs satellites.

4 MATÉRIEL ET LOGICIEL À DISPOSITION

1 ordinateur standard du CPNV avec Windows 10 professionnel et la suite Office 2016 et des crédits Azure
Les autres logiciels disponibles sont :

- Windows 10
- Suite Office
- Balsamiq
- MySQL Workbench
- PHPStorm
- Outils de gestion de projet : GitHub Project / Notion

5 PRÉREQUIS

Le candidat possède de bonnes connaissances programmation Web et modélisation (I-CT 101, 104, 105, 120, 133, 151) et sait mettre en œuvre les différents concepts nécessaires au développement d'application. Le candidat a acquis déjà une expérience en programmation Web et a déjà travaillé avec une base de données MySQL.

Il a également des connaissances en JavaScript (React.js / Node.js).

6 DESCRIPTIF DU PROJET

L'objectif principal de ce projet est de représenter sur une carte 3D le système solaire. Cette carte sera interactive et intégrera les 8 planètes (Mercure, Vénus, la Terre, Mars, Jupiter, Saturne, Uranus et Neptune) et une étoile (le soleil). Il sera possible de se déplacer dans la carte et devoir les planètes sous d'autres angles. Les planètes se déplaceront circulairement autour de leur étoile ainsi que sur elles-mêmes.

Dans un premier temps, le candidat créera un système solaire contenant les planètes qui circulent autour du soleil et fera en sorte qu'il soit possible de se déplacer sur la carte.

Le placement et la rotation des planètes en temps réels.

Le site récupérera les données de placements et de rotations des planètes via la [plateforme API 1 de la NASA](#) grâce au système [Horizon](#) et utilisera ces données afin déplacer les planètes à leurs emplacements réels et actuels.

Informations des planètes

Les informations sur les planètes telles que leur densité, leur circonférence, leur inclinaison, etc. seront récupérées.

ANNEXE 1**Cahier des charges****TPI**

Ces informations seront ensuite affichées à l'utilisateur quand celui-ci clique sur l'une d'elles. Ces données seront récupérées via la plateforme API de la NASA grâce au système Horizon. Des images ainsi que la météo de certaines planètes peuvent également être récupérées depuis différentes API.

Vitesses de déplacements des planètes.

Cette fonctionnalité consistera à permettre à l'utilisateur de voir en accéléré ou en temps réel le déplacement des planètes.

Objets proches

Cette fonctionnalité a pour objectif de récupérer la liste des objets spatiaux (Astéroïdes, etc..) au sein du système solaire et de les afficher sur la carte. Il est possible de récupérer les informations de-celles-ci depuis la plateforme API de la NASA.

L'application devra être ergonomique, c'est-à-dire simple à utiliser en limitant au mieux les actions utilisateurs (nombre de clics)

7 LIVRABLES

Le candidat est responsable de livrer à son chef de projet et aux deux experts :

- 02.05.2023 : une planification initiale contenant la liste des sprints ainsi que les dates/heures des sprint reviews (confirmées avec le PO).
- A chaque fin de sprint :
 - Un commit signifiant clairement le livrable pour le sprint
 - Une documentation mise à jour
 - Un déploiement du site quand la réalisation aura débuté
- A la fin du TPI, le candidat livrera :
 - Les sources et les données sur le dépôt
 - La documentation sur le dépôt
 - Un CD ou clé USB avec les sources, la documentation et le journal à jour
- Chaque vendredi en fin de journée, le journal de travail devra être transmis par courriel au responsable de projet

8 POINTS TECHNIQUES ÉVALUÉS SPÉCIFIQUES AU PROJET

La grille d'évaluation définit les critères généraux selon lesquels le travail du candidat sera évalué (documentation, journal de travail, respect des normes, qualité, ...).

En plus de cela, le travail sera évalué sur les sept points spécifiques suivants :

1. *La carte s'affiche avec toutes les 8 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 déterminés par l'utilisateur*

ANNEXE 1

Cahier des charges

TPI

9 VALIDATION

	Lu et approuvé le :	Signature :
JUILLET Mikael		
MONTEMAYOR Ernesto		
SAHLI Bertrand		
Pascal BENZONANA		

10 HORAIRES DE TRAVAIL



Figure 51 : Cahier des charges 4/4

11.5 Archives du projet

Site web: <https://stellarmap.mycpnv.ch/>

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>