

StellarMap



1 Table des matières

1	Table des matières	2
2	Analyse préliminaire	3
2.1	Introduction	3
2.2	Objectifs.....	3
2.3	Planification initiale	5
3	Analyse / Conception.....	6
3.1	Concept	6
3.2	Stratégie de test.....	6
3.3	Risques techniques	7
3.4	Planification	7
3.5	Dossier de conception	7
3.5.1	Logiciels / Framework utilisé :.....	7
4	Réalisation.....	8
4.1	Dossier de réalisation	8
4.1.1	Classes.....	8
4.1.2	Requête API	8
4.2	Répertoires	12
4.3	Description des tests effectués.....	13
4.4	Erreurs restantes	13
4.5	Liste des documents fournis.....	13
5	Conclusions	14
6	Bibliographie.....	15
7	Table des illustrations	15
8	Lexique.....	15
9	Annexes.....	16
9.1	Planification initiale	16
	18
9.2	Résumé du rapport du TPI / version succincte de la documentation	20
9.3	Journal de travail	20
9.4	Archives du projet.....	20

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 90 heures, entre le 2 mai de 8h50 au 2 juin à 15h20.

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

2.2 Objectifs

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

Sept objectifs spécifiques sont à atteindre :

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

Cahier des charges

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 9.1](#).

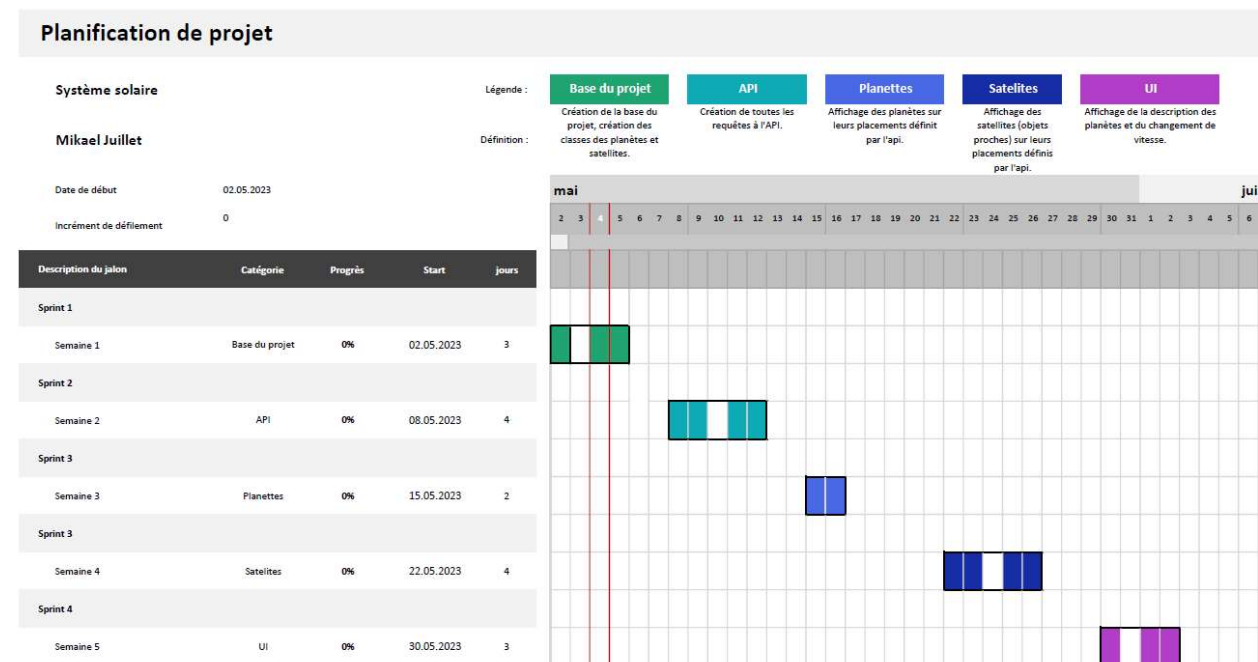


Figure 2 : Planification initiale du projet

3 Analyse / Conception

3.1 Concept

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

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

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

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

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

Documentation :

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

Journal de travail :

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

3.1.1.1 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 actuel, la température etc.

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

3.1.1.2 Maquettes

3.2 Stratégie de test

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

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

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

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

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

3.4 Planification

3.5 Dossier de conception

3.5.1 Logiciels / Framework utilisé :

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

4 Réalisation

4.1 Dossier de réalisation

4.1.1 Classes

4.1.2 Requête API

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

```
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 + "&";
        }
    });

    console.log(fullURL)
    return fullURL;
}
```

4.1.2.1 Variables d'environnement

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

```
.env
1 VITE_API_URL_HORIZON=https://ssd.jpl.nasa.gov/api/horizons.api
2 VITE_API_URL_NEO=https://api.nasa.gov/neo/rest/v1/feed
3 VITE_API_KEY=fakeAPIKEY
```


4.1.2.2 Horizon system

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

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.

Cette requête engendre une erreur corse après quelques recherches j'ai ajouté dans mon 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.

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.

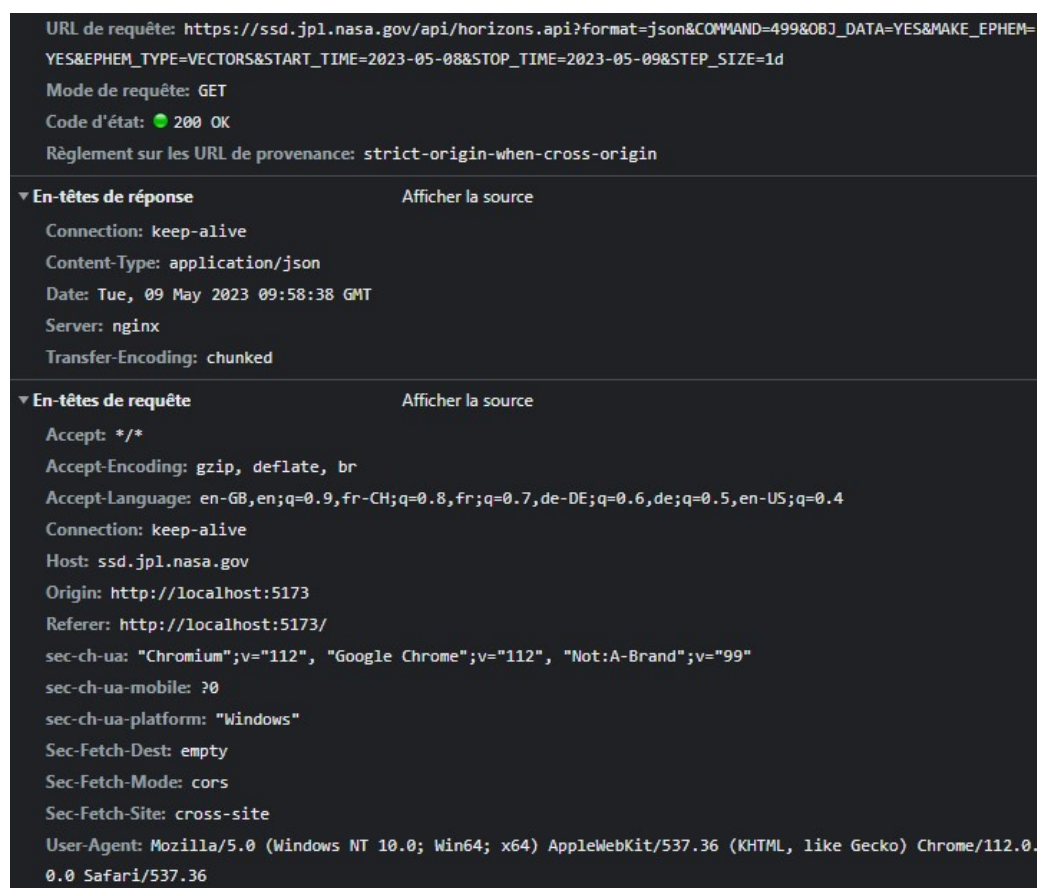


Figure 3 : Etat de la requête Horizon API sans données.

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, puis j'exporterai les données dans un fichier json que je traiterai ainsi afin de ne pas perdre du temps. Si le temps le permet je reviendrais éventuellement sur cette partie en fin de projet, en plus de ceci j'ai créé une [issue](#) sur GitHub afin de maintenir le projet.

4.1.2.2.1 Récupération des données

Les données reçues sont sous forme de String, je dois récupérer des valeurs précise dans ces tableaux c'est pourquoi j'utilise des expression régulière (regex). Celle-ci vont me permettre de récupérer des valeurs en fonction d'une expression régulière dans mon fichier. (mozilla, 2023)

```
*****Re dataFilter.js?t=1683623740838:11
vised:April12,2021Venus299/2PHYSICALDATA(updated2020-Oct-
19):Vol.MeanRadius(km)=6051.84+-0.01Density(g/cm^3)=5.204Massx10^23(kg)=48.685Volume(x10^10km^3)=92.843Siderealrot.
period=243.018484dSid.Rot.Rate(rad/s)=-0.00000029924Meansolarday=116.7490dEqu.gravity/m/s^2=8.870Mom.ofInertia=0.33C
oreradius(km)=~3200GeometricAlbedo=0.65PotentialLove#k2=-0.25GM(km^3/s^2)=324858.592EquatorialRadius,Re=6051.893kmG
M1-
sigma(km^3/s^2)=-0.006Massratio(Sun/Venus)=408523.72Atmos.pressure(bar)=90Max.angulardiam.=60.2"MeanTemperature(K)
=735Visualmag.V(1,0)=-4.40Obliquitytoorbit=177.3degHill'ssphererad.,Rp=167.1Siderealorb.per.,y=0.61519726Orbitspeed
,km/s=35.021Siderealorb.per.,d=224.70079922Escapespeed,km/s=10.361PerihelionAphelionMeanSolarConstant(W/m^2)2759261
42650MaximumPlanetaryIR(W/m^2)153153153MinimumPlanetaryIR(W/m^2)153153153*****
*****
*Ephemeris/API_USERMonMay807:4
2:042023Pasadena,USA/Horizons*****Targetb
odyname:Venus(299){source:DE441}Centerbodyname:Earth(399){source:DE441}Center-
sitename:BODYCENTER*****Starttime:A.D.202
3-May-0400:00:00.0000TDBStoptime:A.D.2023-May-0500:00:00.0000TDBStep-
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),Dxy(km),Dz(km)}Centerradii:6378.137,6378.137,6356.752km{Equator_a,b,pole_c}Outputunits:KM-
SCalendarMode:MixedJulian/GregorianOutputtype:GEOMETRICcartesianstatesOutputformat:3(position,velocity,LT,range,ran
ge-rate)EOPfile:eop.230504.p230727EOPcoverage:DATA-BASED1962-JAN-20T02023-MAY-04.PREDICTS->2023-JUL-
26ReferenceFrame:EclipticofJ2000.0*****J2
000-A.D.2023-May-
0400:00:00.0000TDBX=1.073618289428096E+07Y=1.428930836016723E+08Z=6.291878178878881E+06VX=-3.305499528421466E+01VY=
-1.075164726133031E+01VZ=3.134771643171308E-
01LT=4.784440259297880E+02RG=1.434339105489069E+08RR=-1.317155851465099E+012460069.500000000-A.D.2023-May-
0500:00:00.0000TDBX=7.904291596898273E+06Y=1.419336731964773E+08Z=6.31645136050268E+06VX=-3.249542341508560E+01VY=
-1.145345746249560E+01VZ=2.553209657597706E-
01LT=4.746412310477860E+02RG=1.422938613239617E+08RR=-1.321822214062615E+01$EOE*****
*****TIMEBarycentricDynamicalTime("TDB"orT_eph)outputwasrequested.Thiscontin
uouscoordinatetimeisequivalenttotherelativisticproptimeofaclockatrestinareferenceframeco-
movingwiththesolarsystembarycenterbutoutsidethesystem'sgravitywell.Itistheindependentvariableinthesolarsystemrelati
visticquationsofmotion.TDBrunsatauniformrateofoneSIsecondpersecondandisindependentofirregularitiesinEarth'srotatio
n.CALENDARSYSTEMMixedcalendarmodewasactivesuchthatcalendardatesafterAD1582-Oct-
15(ifany)areinthemodernGregoriansystem.Datespriorto1582-Oct-
5(ifany)areintheJuliancalendarssystem,whichisautomaticallyextendedfordatespriortoit'sadoptionon45-Jan-
18C.TheJuliancalendarisusefulformatchinghistoricaldates.TheGregoriancalendarismoreaccuratelycorrespondstotheEarth'sor
bitalmotionandseasons.A"Gregorian-
only"calendarmodeisavailableifsuchphysicaleventsaretheprietaryinterest.REFERENCEFRAMEANDCOORDINATESEclipticatthetan
dardreferenceepochReferenceepoch:J2000.0X-
Yplane:adoptedEarthorbitalplaneatthereferenceepochNote:IAU76obliquityof84381.448arcsecondswrtICRFX-YplaneX-
axis:ICRFZ-axis:perpendiculartotheX-
Yplaneinthedirectional(+or-)senseofEarth'snorthpoleatthereferenceepoch.Symbolmeaning:JDDBJulianDayNumber,Barycentr
icDynamicalTimeXX-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)RGRRange;distancefromcoordinatecenter(km)RRRRange-
rate;radialvelocitywrtcoord.center(km/sec)ABERRATIONSANDCORRECTIONSGeometricstatevectorshaveNOcorrectionsoraberrati
onsapplied.Computationsby...SolarSystemDynamicsGroup,HorizonsOn-
LineEphemerisSystem4800OakGroveDrive,JetPropulsionLaboratoryPasadena,CA91109USAGeneralsite:https://ssd.jpl.nasa.go
v/Mailinglist:https://ssd.jpl.nasa.gov/email_list.htm*****
*****
6051.84 dataFilter.js?t=1683623740838:20
```

Figure 4 : Données récupérée après suppression des espaces.

```
export function horizonAPI(data) {  
  
  let texteSansEspaces = data.replace(/\s+/g, "");  
  console.log(texteSansEspaces);  
  
  const regex = /Vol\.MeanRadius\(km\)=(\d+\.\d+)/;  
  
  const match = texteSansEspaces.match(regex);  
  
  if (match !== null) {  
    const density = match[1];  
    console.log(density);  
  } else {  
    console.log("Aucune correspondance trouvée.");  
  }  
  
}
```

Figure 5 : Exemple de l'utilisation de regex

4.1.2.3 Near Earth Objects

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

4.2 Répertoires

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

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

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

Architecture des documents :

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

4.3 Description des tests effectués

4.4 Erreurs restantes

4.5 Liste des documents fournis

5 Conclusions

6 Bibliographie

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

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>

day.js. (2023, - -). *day.js*. Récupéré sur day.js: <https://day.js.org/>

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/>

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>

pinia. (2023, - -). *pinia*. Récupéré sur pinia: <https://pinia.vuejs.org/>

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/>

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

7 Table des illustrations

FIGURE 1 : SYSTEME SOLAIRE (FREEPIK, -)	1
FIGURE 2 : ETAT DE LA REQUETE HORIZON API SANS DONNEES.	9
FIGURE 3 : DONNEES RECUPEREE APRES SUPPRESSION DES ESPACES.	10
FIGURE 4 : EXEMPLE DE L'UTILISATION DE REGEX	11

8 Lexique

Aucune entrée d'index n'a été trouvée.

9 Annexes

9.1 Planification initiale

Description	Catégorie	Progrès	Début	Heures prévu
Sprint 1				
Diagramme de classes	Base du projet	0%	03.05.2023	0.75
Diagramme de séquence	Base du projet	0%	03.05.2023	0.75
Création de la classe planète	Base du projet	0%	03.05.2023	1.00
Création de la classe satellite	Base du projet	0%	03.05.2023	1.00
Ajout des opérations dans les classes	Base du projet	0%	05.05.2023	2.25
Ajout de vitejs	Base du projet	0%	05.05.2023	0.25
Création du fichier détenant les codes de planètes	Base du projet	0%	05.05.2023	0.50

Sprint 2				
Création d'un contrôleur	API	0%	08.05.2023	0.25
Création d'un modèle	API	0%	08.05.2023	0.25
Ajout d'un fichier .env	API	0%	08.05.2023	0.25
Récupération de la clef API	API	0%	08.05.2023	0.25
Requêtes de récupération des planètes	API	0%	08.05.2023	3.00
Requêtes de récupération des objets proches	API	0%	09.05.2023	3.00
Requêtes de récupération des images	API	0%	11.05.2023	2.25
Récupération des erreurs dans le contrôleur	API	0%	12.05.2023	0.75
Lien entre la récupération des données et les classes	API	0%	12.05.2023	0.75

Sprint 3

Création de maquettes	Planettes	0%	14.05.2023	0.75
Ajout de three.js	Planettes	0%	14.05.2023	0.25
Création des planètes	Planettes	0%	14.05.2023	0.75
Placement des planètes	Planettes	0%	15.05.2023	0.50
Orbite sidérale	Planettes	0%	15.05.2023	0.50
Orbite autour du soleil	Planettes	0%	15.05.2023	0.75

Sprint 4

Ajout du déplacement utilisateur	Satelites	0%	22.05.2023	2.25
Création des Satélites	Satelites	0%	23.05.2023	1.50
Ajout des lunes	Satelites	0%	25.05.2023	1.50
Placement sur la carte	Satelites	0%	25.05.2023	1.50
Ajout de l'orbite	Satelites	0%	26.05.2023	1.50
Orbite sidérale	Planettes	0%	26.05.2023	1.50

Sprint 5

Création de maquettes	UI	0%	30.05.2023	0.75
Placement du canvas en arrière-plan	UI	0%	30.05.2023	0.75
Ajout de la description des planètes	UI	0%	01.06.2023	2.25
Ecoule d'un clique sur planètes	UI	0%	02.06.2023	0.75
Ajout de changement de vitesse	UI	0%	02.06.2023	1.50

Planification de projet

Système solaire

Mikael Juillet

Date de début 02.05.2023

Incrément de défilement 0

Légende :

Base du projet

Création de la base du projet, création des classes des planètes et satellites.

API

Création de toutes les requêtes à l'API.

Planètes

Affichage des planètes sur leurs placements définis par l'api.

Satellites

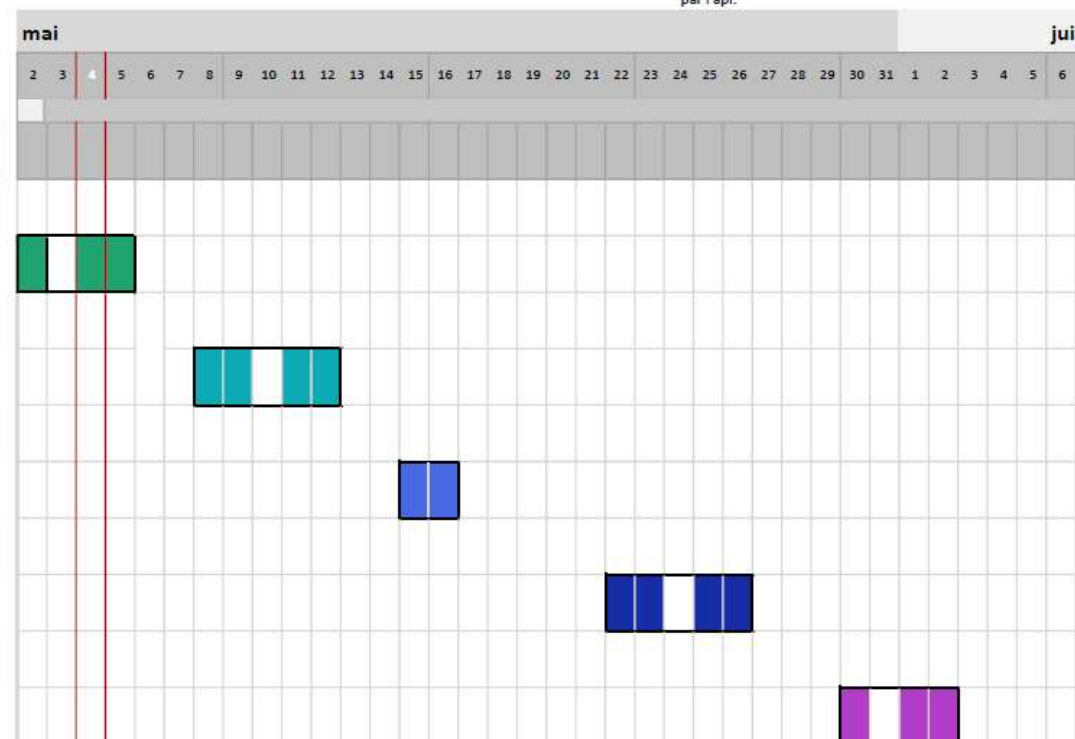
Affichage des satellites (objets proches) sur leurs placements définis par l'api.

UI

Affichage de la description des planètes et du changement de vitesse.

Définition :

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

Système solaire

Mikael Juillet

Date de début 02.05.2023

Légende:

Documentation

Implémentation

Annalyse

Tests

Sprint reviews

* P = Période de 45 minutes

Horaire

Jours
Lundi
Mardi
Mercredi
Jeudi
Vendredi
Samedi
Dimanche

P1	P2	P3	P4	P5	P6	P7	P8	P9
Documentation	Documentation	Documentation	Documentation	Documentation	Documentation	Documentation	Documentation	Documentation
	Documentation	Documentation	Documentation	Documentation	Documentation	Documentation	Documentation	
Documentation	Documentation	Documentation			Documentation	Documentation	Tests	Documentation
	Documentation	Documentation	Tests	Tests	Documentation	Documentation		

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.

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

9.3 Journal de travail

9.4 Archives du projet

Media, ... dans une fourre en plastique