

## Rapport Projet Rigel - Bonus

Bouilloud Mathias (309979)

Mettler Julien (309999)

### **Bonus 1 : Dessin de la Lune en tenant compte de sa phase**

**But :** Le premier bonus dessine la Lune en fonction de sa phase. Pour déterminer son aspect, nous nous sommes appuyés sur les valeurs de phase et leurs images correspondantes, données dans la partie 1.2 de l'article "Phase de la Lune" de Wikipédia ([https://fr.wikipedia.org/wiki/Phase\\_de\\_la\\_Lune#Cycle](https://fr.wikipedia.org/wiki/Phase_de_la_Lune#Cycle)).

Nous prenons également en compte l'orientation du croissant, selon que l'observateur se trouve dans l'hémisphère Nord ou dans l'hémisphère Sud.

**Mise en œuvre :** Le dessin se fait dans la méthode privée **drawMoonPhase** de SkyCanvasPainter qui est appelée dans **drawMoon**. Le croissant est dessiné à l'aide d'instances de la classe Arc de JavaFX, ou un cercle si la phase est 1.

La position de l'observateur (GeographicCoordinates) de ObserverLocationBean, est passée en arguments de drawMoon et drawMoonPhase.

Cette distinction des cas permet de régler l'argument startAngle de Arc lorsque l'on essaie de dessiner l'arc de cercle correspondant au croissant.

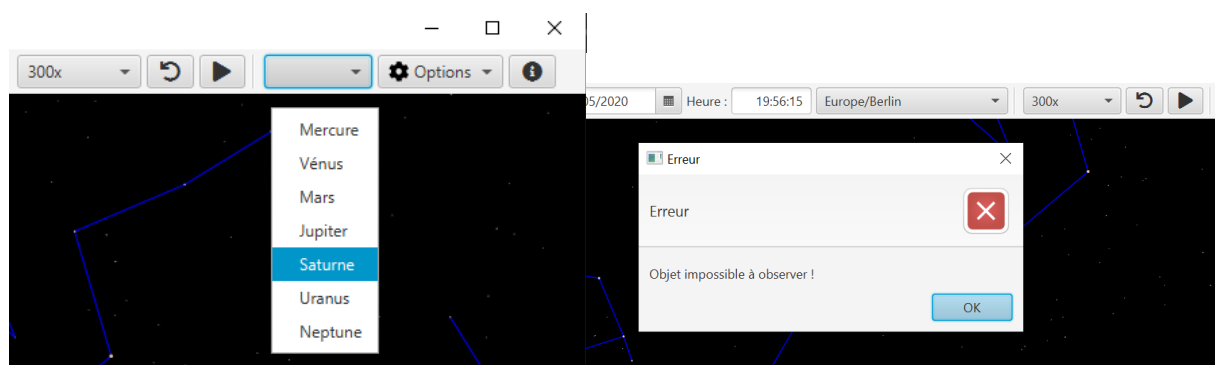
Les bonus qui suivent font partie de la HBox **bonusInterface** de Main, dans la partie droite de la barre de contrôle.

### **Bonus 2 : Sélection d'une planète comme centre d'observation (méthode *planetsMenu* de Main)**

**But :** Le second bonus est un menu déroulant des planètes du système solaire. Une fois sélectionnée, les coordonnées horizontales de la planète deviennent celles du centre de projection. Si la planète n'est pas observable (i.e. latitude non comprise dans  $[0^\circ, 90^\circ]$ ), une fenêtre d'alerte apparaît pour prévenir l'utilisateur.

Il est impossible de sélectionner une planète lorsqu'une animation est en cours.

**Mise en œuvre :** Pour le menu, nous avons utilisé une **ChoiceBox<Planet>** et pour la fenêtre d'alerte, nous avons utilisé une instance **d'Alert**, classe de JavaFX.



Le menu déroulant et l'alerte

### **Bonus 3 : Menu d'options (méthode *optionsMenu* de Main)**

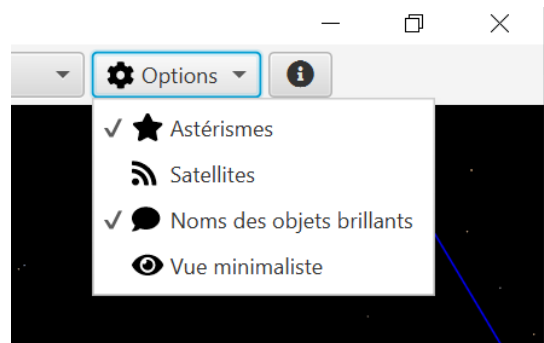
But : Le troisième bonus permet d'activer ou désactiver le dessin des astérismes, des satellites (bonus 4) et des noms des objets brillants (bonus 5).

L'option "Vue minimaliste" désactive toutes ces options, n'affichant que les étoiles et l'horizon. L'utilisateur ne peut alors pas activer les options et doit d'abord désactiver la vue minimaliste pour y parvenir.

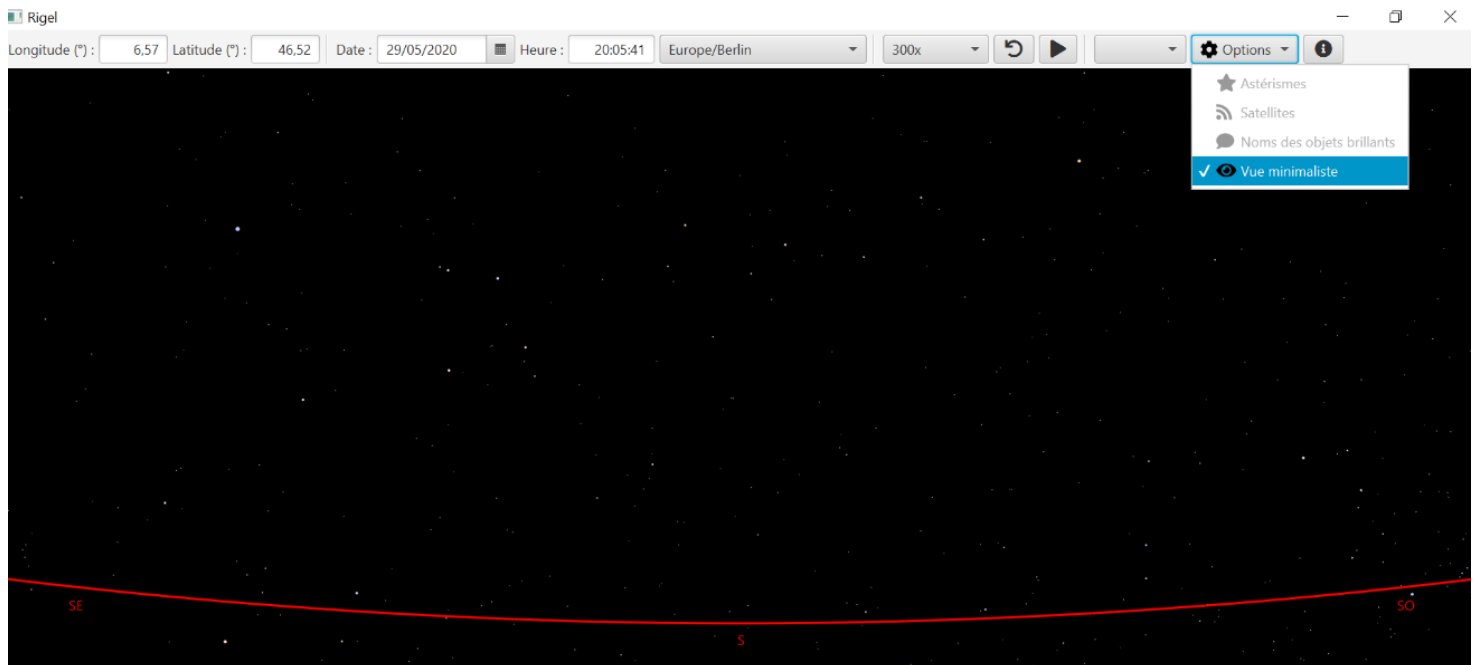
#### Mise en œuvre :

Le menu est un **MenuButton** et chaque option est un **CheckMenuItem**, contrôlée par une SimpleBooleanProperty (asterismEnable, satelliteEnable, nameEnable) de SkyCanvasManager. Leurs valeurs booléennes sont passées en arguments des méthodes de dessin de SkyCanvasPainter pour effectuer le dessin uniquement si la valeur est true.

Le fait de d'activer ou non une option ne fait donc que changer la valeur contenue dans sa propriété correspondante.



*Le menu d'options*



*Le ciel observé en vue minimaliste*

#### **Bonus 4 : Affichage des satellites actifs en orbite autour de la Terre**

But : Le quatrième bonus affiche 375 satellites en orbite géostationnaire autour de la Terre (représentées par des points verts), leur utilisation (communications, météorologie) et leur pays d'origine.

Mise en œuvre : Le catalogue provient du fichier « **active\_satellites.csv** » de type CSV, du site [Active Satellites in Orbit Around Earth](http://www.active-satellites.com/).

Nous avons ajouté trois classes :

- Satellite, héritant de CelestialObject
- SatelliteCatalogue, attribut de ObservedSky et son bâtisseur Builder, inspirées de StarCatalogue
- SatelliteDatabaseLoader, inspirée de HygDatabaseLoader

Le tableau de type double[] contenant les positions des satellites dans le plan est calculé dans le constructeur de ObservedSky puis utilisé dans la méthode drawSatellites de SkyCanvasPainter, à la manière des étoiles dans DrawStars.

Plusieurs précisions sont nécessaires :

- Les objets en orbite géostationnaire ont une inclinaison nulle à l'équateur, ce qui permet de facilement représenter leur orbite dans le plan équatorial, d'où notre choix d'utiliser ce système de coordonnées pour leur position.
- Leur orbite se situe à environ 36 000 km au-dessus de la Terre<sup>1</sup>. Nous avons donc décrit les satellites par une même latitude
- Le catalogue date de Juillet 2016 et n'est donc plus exhaustif

#### **Bonus 5 : Annotations des objets brillants, des planètes, de la Lune, du Soleil**

Le 5e bonus affiche les informations (méthode info()) des objets célestes les plus brillants : le Soleil, la Lune, les étoiles les plus brillantes, à leur position sur le canevas. Ces derniers sont déterminés à leur construction dans CelestialObject, à l'aide d'un attribut booléen isBright qui est **true** si la **magnitude est strictement inférieure à 1**.

Toutes les planètes sont également annotées, y compris celles non visibles à l'oeil nu (e.g. Uranus et Neptune) pour que l'utilisateur repère plus facilement leur position lorsqu'il les sélectionne (bonus 2).

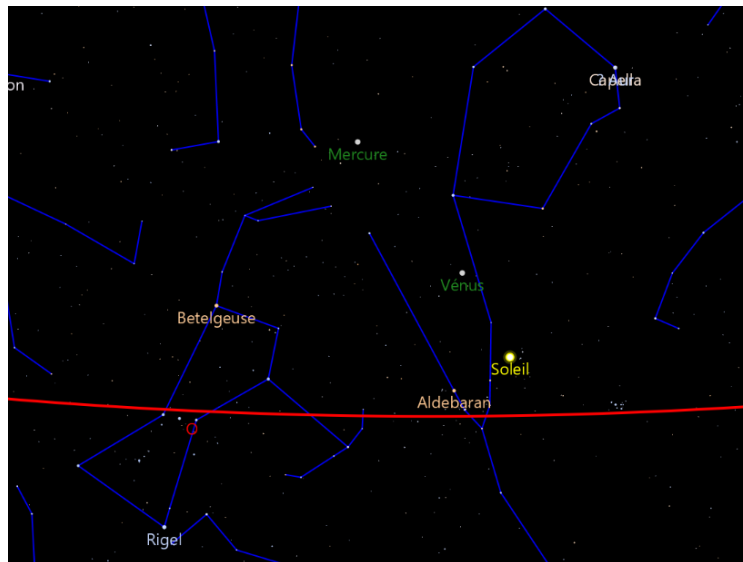
Le dessin du texte (classe Text de JavaFX) est effectué dans la méthode **drawAnnotation** de SkyCanvasPainter, seulement si le paramètre booléen nameEnable est true (bonus 3)

Un code couleur a été choisi pour chaque type d'objet :

- Pour les étoiles, la couleur correspondant à leur température
- Pour les planètes, le vert
- Pour le Soleil, le jaune
- Pour la Lune, le blanc

---

<sup>1</sup> [https://fr.wikipedia.org/wiki/Orbite\\_g%C3%A9ostationnaire](https://fr.wikipedia.org/wiki/Orbite_g%C3%A9ostationnaire)



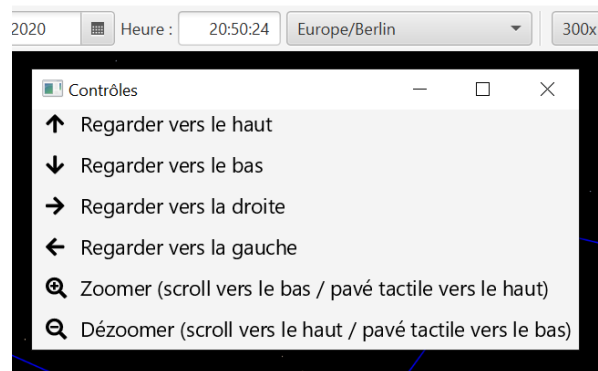
*Un exemple d'annotations d'étoiles, de quelques planètes et du Soleil*

### **Bonus 7 : Bouton d'informations**

But : Pour faciliter la première expérience de l'utilisateur, le dernier bonus ajoute un bouton



dont l'activation affiche une deuxième fenêtre (une nouvelle **Scene**) montrant les contrôles et leur utilité, c'est-à-dire les touches directionnelles et les scrolls de la souris, avec leurs icônes correspondantes.



*La fenêtre d'informations*

### Mise en œuvre :

La fenêtre contient une GridPane où la première colonne est vide pour laisser une marge, la deuxième colonne contient une icône de FontAwesome (caractère Unicode) et la troisième colonne contient son contrôle correspondant.