tpi – Création d’un site web d’initiation à l’astronomie

[Sous-titre du document]

[Date]

CPNV

Auteur Jegatheeswaran Nithujan

Table des matières

[1. Analyse préliminaire 3](#_Toc102661221)

[1.1. Introduction 3](#_Toc102661222)

[1.2. Objectifs 3](#_Toc102661223)

[1.3. Planification initiale 4](#_Toc102661224)

[1.3.1. Dates du projet : 4](#_Toc102661225)

[1.3.2. Tâches 4](#_Toc102661226)

[1.3.3. Diagramme de Gantt 5](#_Toc102661227)

[2. Analyse / Conception 6](#_Toc102661228)

[2.1. Concept 6](#_Toc102661229)

[2.1.1. Base de données 6](#_Toc102661230)

[2.1.2. Maquettes des pages web 8](#_Toc102661231)

[2.2. Stratégie de test 11](#_Toc102661232)

[2.3. Risques techniques 11](#_Toc102661233)

[2.4. Planification 11](#_Toc102661234)

[2.5. Dossier de conception 11](#_Toc102661235)

[3. Réalisation 13](#_Toc102661236)

[3.1. Dossier de réalisation 13](#_Toc102661237)

[3.2. Description des tests effectués 16](#_Toc102661238)

[4. Annexes 16](#_Toc102661239)

[4.1. Sources – Bibliographie 16](#_Toc102661240)

# Analyse préliminaire

## Introduction

Ce projet est réalisé dans le cadre d’une formation FPA (Formation Professionnelle Accélérée) permettant d’obtenir un CFC. Aucun travail n’a été effectué au préalable.

Ce projet a pour but de créer un site web qui a pour thème l’astrologie. Il devra permettre de consulter diverses informations sur les objets du catalogue de Messier et d’organiser des sorties nocturnes tout en offrant la possibilité aux visiteurs du site de s’y inscrire.

Ce projet est réalisé en Python car c’est un langage de programmation polyvalent qui a une communauté active. Le site web communiquera avec un serveur PostgreSQL qui contiendra toutes les données persistantes. Ce SGBDRO (Système de Gestion de Base de Données Relationnel-Objet) a été choisi pour faciliter la mise en ligne du site mais également car il est entièrement gratuit.

## Objectifs

Les objectifs généraux du projet sont :

* Une page d’accueil qui présente le site et ses objectifs
* Une page contenant les objets du catalogue Messier avec la possibilité de faire des recherches et de les trier par ordre alphabétiques ou numériques selon la donnée
* Une page sur les sorties nocturnes disponibles avec la possibilité pour le visiteur du site de s’y inscrire
* Une section réservée aux administrateurs qui permet la création de sorties nocturnes
* Le site sera intégralement responsive
* L’automatisation et la réussite de l’import des données nécessaires (fichier CSV et images sur internet) dans la base de données choisie

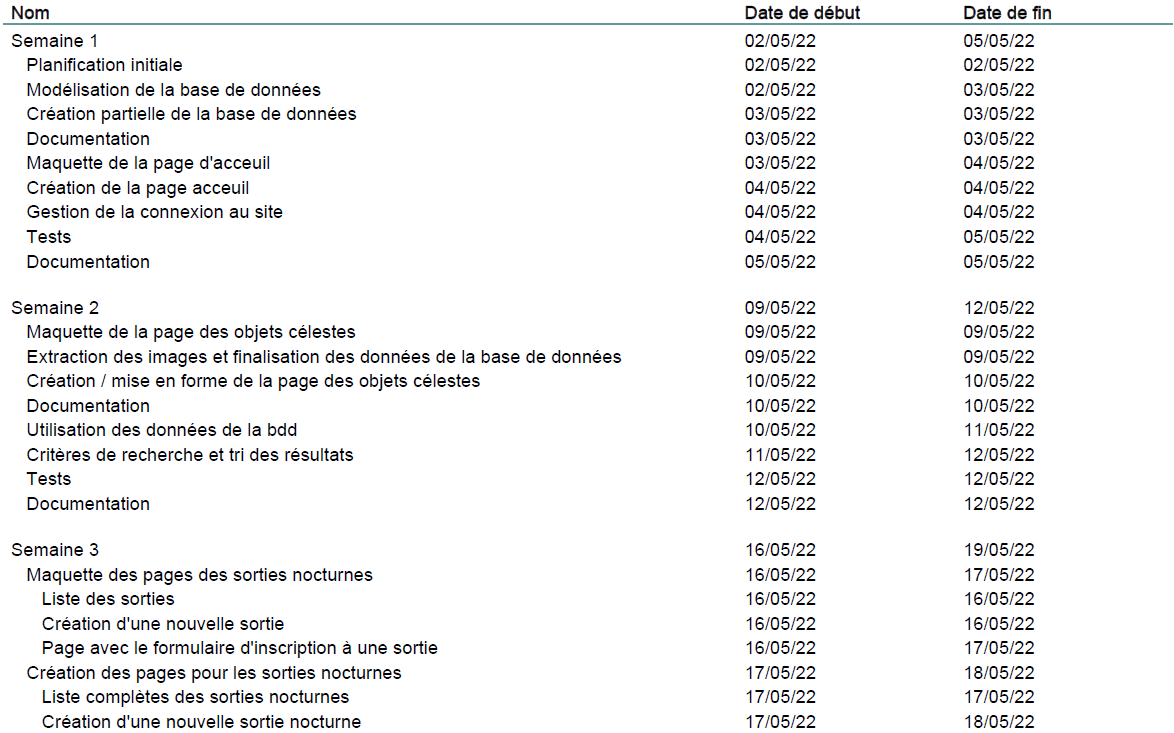
## Planification initiale

Agile est la méthode de gestion de projet choisi. Cette méthode permet de s’adapter en cas de problème

### Dates du projet

Du 02.05.22 au 30.05.22

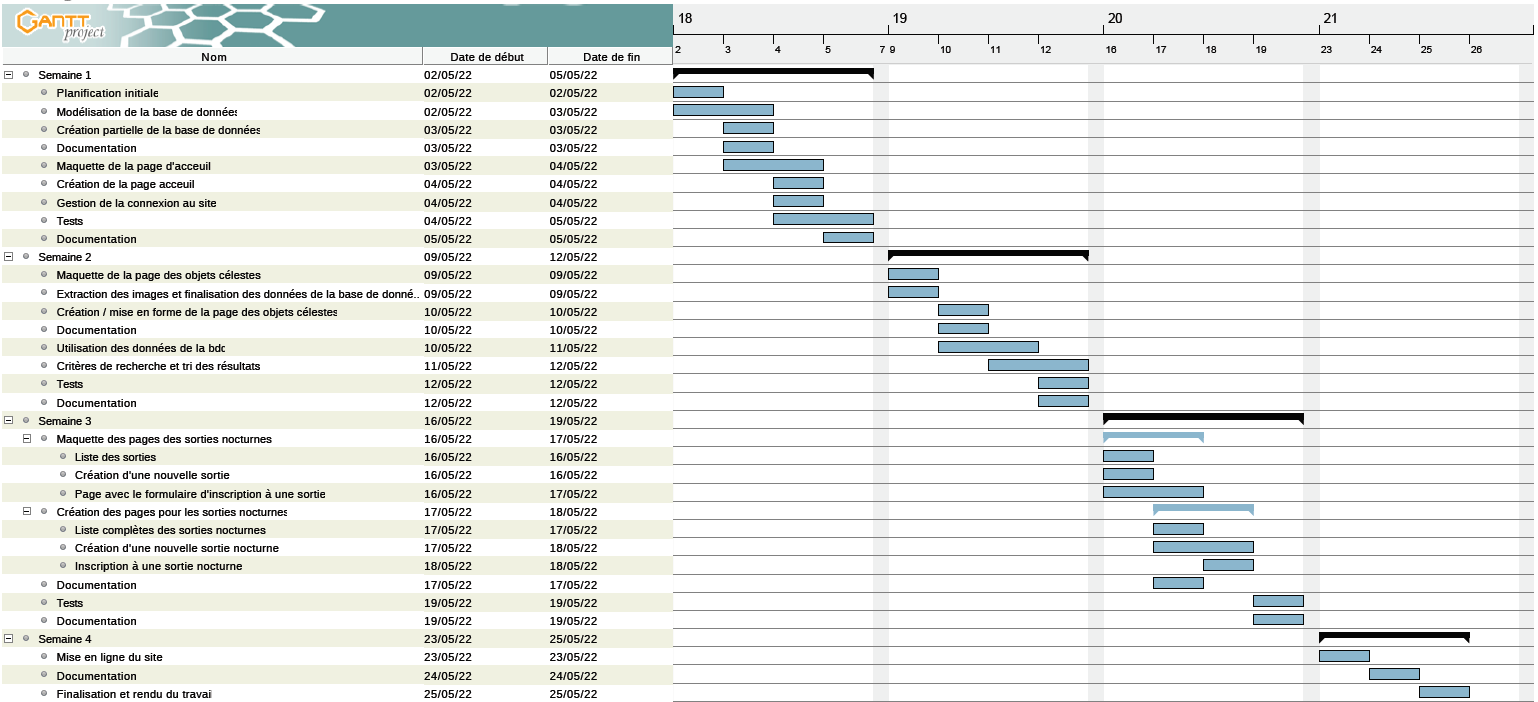
### Tâches



*Rem :*

* Le rendu du travail a été prévu le 25 mai pour cause d’une indisponibilité le lundi 30 mai
* Cette planification implique une gestion de projet en agile, le projet utilise donc l’application web icescrum (A METTRE AILLEURS ???)

### Diagramme de Gantt

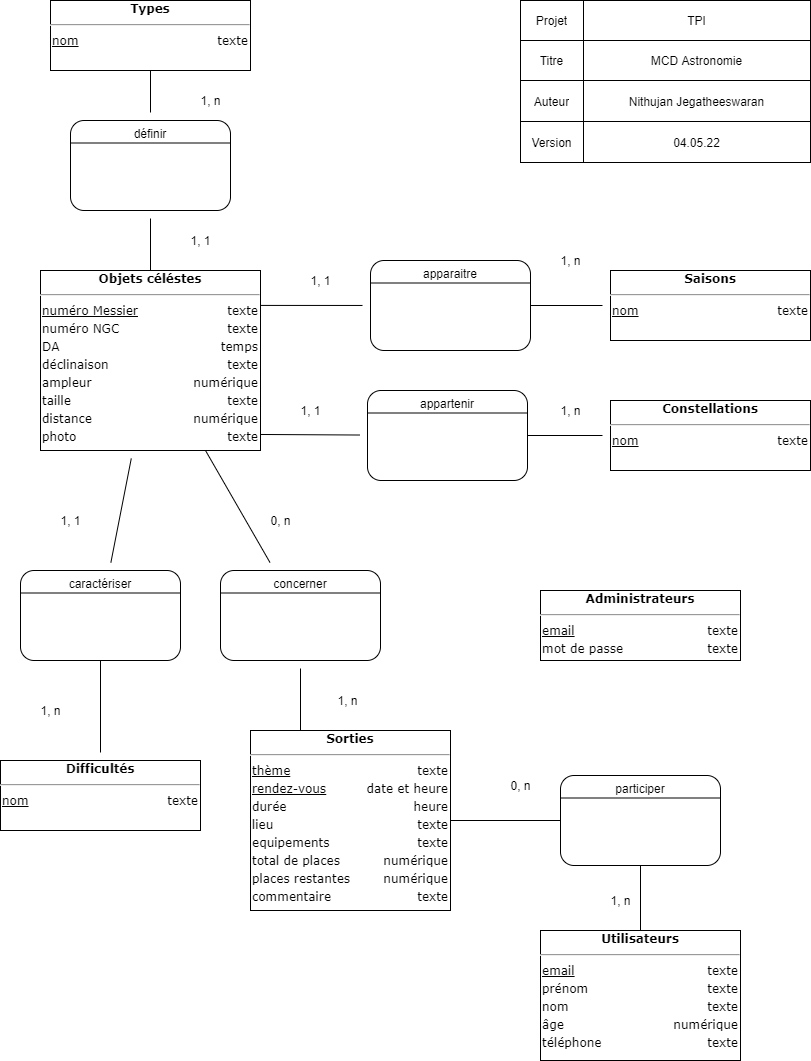


# Analyse / Conception

## Concept

### Base de données

*MCD (Modèle Conceptuel de Données)*

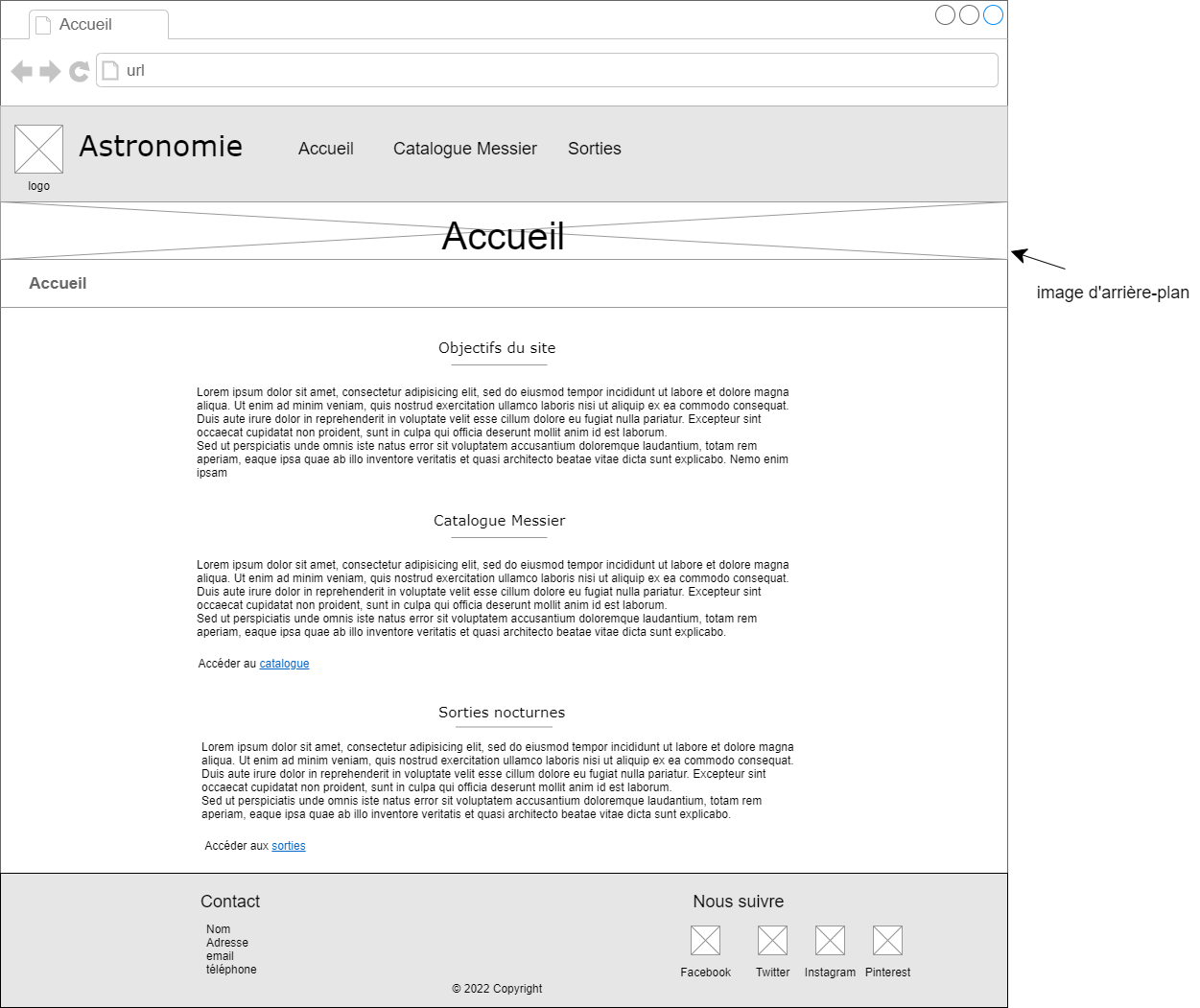


*MLD (Modèle Logique de Données)*

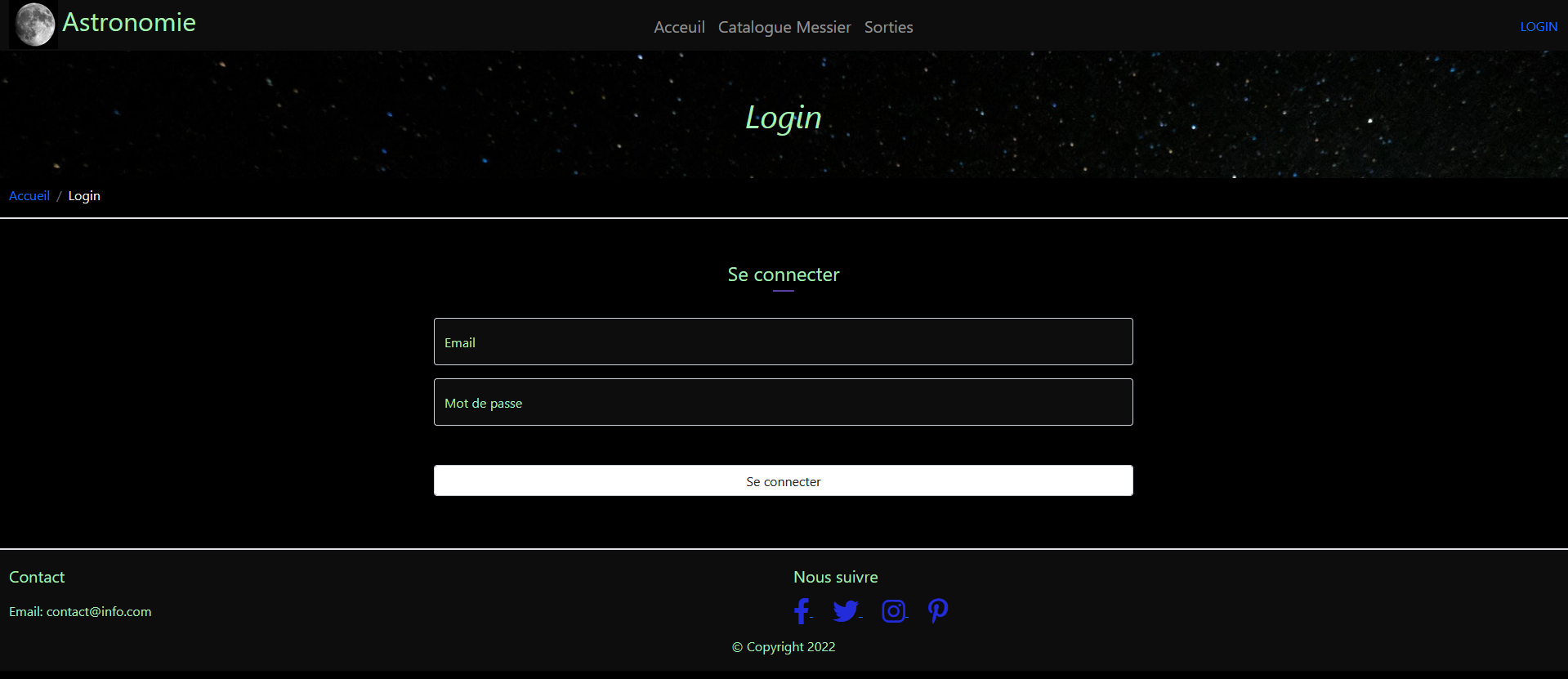
Ajouter la version finale /!\ check lisibilité

### Maquettes des pages web

Page d’accueil



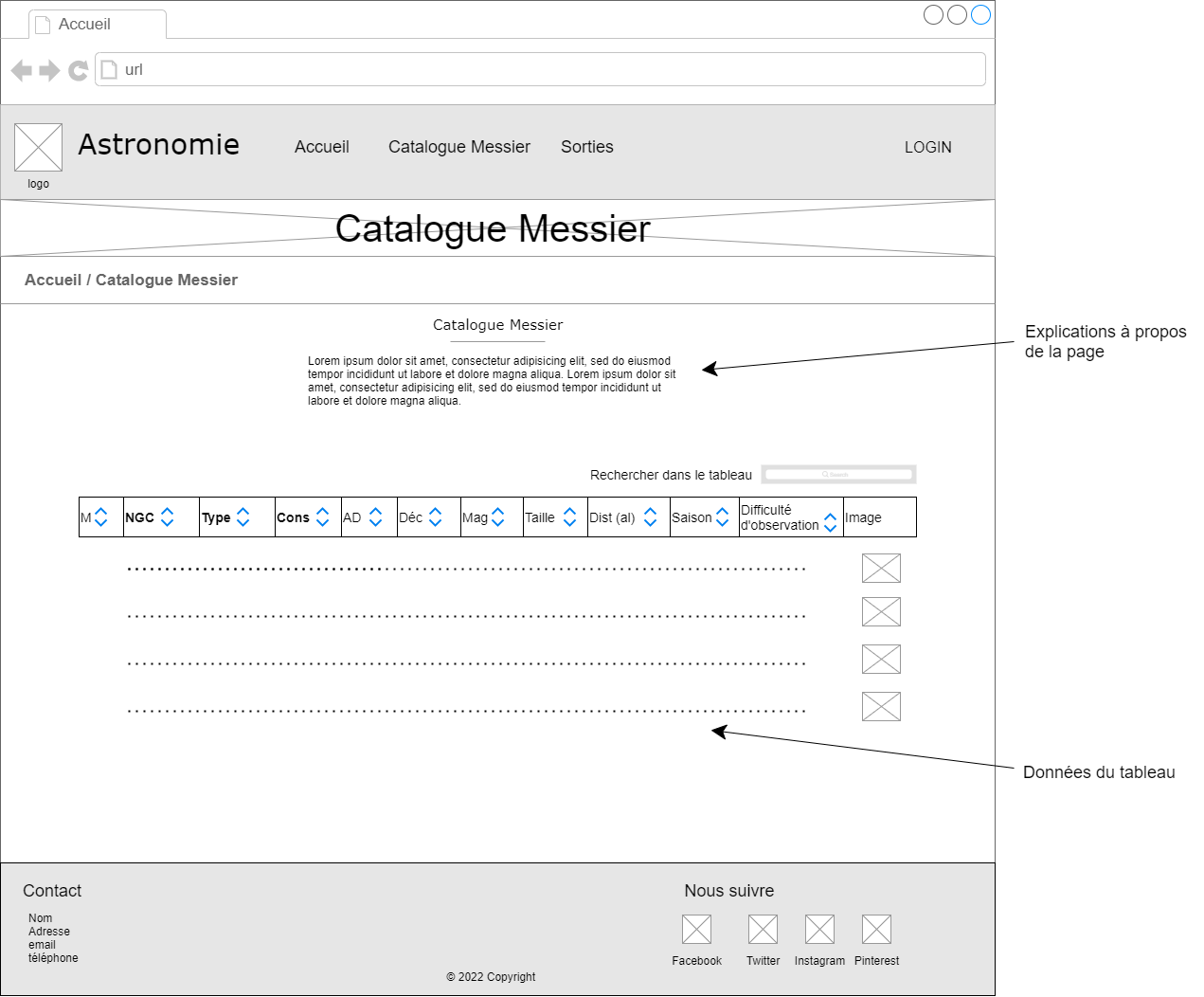
Page de login



*Rem*:

Cette page a été réalisé sans l’aide de maquettes car elle reprend des choses de la maquette de la page d’accueil et les éléments qui varient sont minimes.

Page avec les objets du catalogue Messier :



## Stratégie de test

Les tests de l’application web seront tous manuels. Ils seront faits sur plusieurs navigateurs pour vérifier au maximum la compatibilité avec ces derniers.

## Risques techniques

Les risques techniques majeurs du projet concernent :

* L’import des données depuis un fichier CSV dans la base de données
* Le téléchargement en local d’images d’un site internet
* L’utilisation pour la première fois d’une base de données PostgreSQL

Le cahier des charges fourni des pistes pour remédier à la première tâche. Quant à la seconde, une discussion durant la phase d’analyse avec la cheffe de projet Mme Andolfatto a permis d’apporter des idées de recherche à effectuer. PostgreSQL est quant à lui bien documenté et il est possible de trouver les réponses à beaucoup de questions via des recherches internet.

## Planification

## Dossier de conception

**Logiciels utilisés**

Ce projet est réalisé sur un ordinateur Windows 10 avec l’aide des logiciels suivants :

* Draw.io v15.3.5
* PgAdmin 4 v6.8
* PyCharm 2021.3.3 (Professional Edition)
* Navigateurs web :
  + Firefox
  + Google Chrome
  + Opera

Le logiciel draw.io a été utilisé afin de concevoir les maquettes du site web et le MCD de la base de données. PgAdmin est le client utilisé pour la base de données PostgreSQL. Il est fourni par défaut quand on télécharge PostgreSQL sur le [site officiel](https://www.postgresql.org/). PyCharm a été utilisé pour toute la partie développement du site, que ce soit le code HTML/CSS ou python. Les différents navigateurs web sont utilisés pour effectuer des tests sur plusieurs environnements.

**Base de données**

*Table administrators*

Table sans lien avec les autres qui contient les adresses mail et mot de passe des administrateurs du site. Ces informations sont nécessaires lorsqu’ils veulent se connecter pour gérer le site.

*Table celestial\_objects*

Table regroupant les informations des objets célestes du catalogue de Messier. Les données utilisées proviennent du fichier CSV disponible sur le site [starlust](https://starlust.org/fr/le-catalogue-messier/) (Cette page n’est plus disponible depuis au moins le 12.05.2022).

*Tables constellations, observation\_difficulties, seasons et types*

Ces différentes tables décrivent les entrées de la table *celestial\_objects*. Ces différents éléments ont été placés dans des tables séparées car ils se répètent sur les diverses entrées de *celestial\_objetcts*.

*Table outings*

La table *outings* est destinée à recevoir les données du formulaire permettant d’ajouter de nouvelles sorties nocturnes Ce formulaire peut être rempli par les administrateurs lorsqu’ils sont connecté au site.

*Table users*

La table *users* permet d’enregistrer les données des personnes qui se sont inscrits à une sorites nocturnes. Les utilisateurs ne pouvant pas se connecter ……….

*Table de jointure user\_has\_outing et outing\_has\_celestial\_object*

La table *user\_has\_outing* sert à définir la relation entre les utilisateurs et les sorties auxquelles ils se sont inscrits.

La table *outing\_has\_celestial\_object* fait le lien entre *outings* et *celestial\_objects* car le formulaire qui concerne les sorties permettra de préciser, ou non, les objets célestes qui vont être observés.

**Desing Web**

Afin de respecter au mieux le critère d’un site responsive exigé par le cahier des charges, des classes Bootstrap ont été utilisées pour la création du site.

# Réalisation

## Dossier de réalisation

**Base de données**

Les images des objets du catalogue de Messier, ont été téléchargées via une extension Firefox: [DownThemAll](https://www.downthemall.org/). Ces dernières viennent du site <https://www.messier-objects.com/messier-catalogue/>. Un script PowerShell a été utilisé pour renommer les images importées en local. Les images de la page [Wikipedia](https://en.wikipedia.org/wiki/Messier_object) concernant le catalogue de Messier avaient des noms qui compliquaient le renommage de celles-ci par un script. Le script utilisé a permis de renommer les fichiers et de les placer dans le dossier *assets/* du site web. La valeur des chemins vers ces images a été placée dans le fichier pictures.csv

Pour les données de la table celestial\_objects, un autre script a dû être utilisé : extract-fk-values.ps1. Ce script a pris les valeurs uniques des colonnes constellation, meilleur saison, difficulté d’observation et meilleure saison pour les placer dans d’autre fichiers CSV à part.

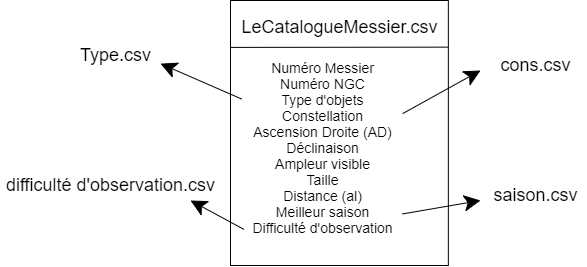
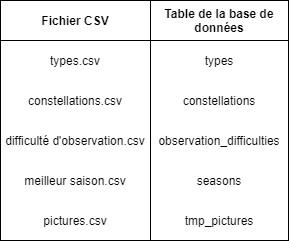


Diagramme 1- Colonnes exportées vers des fichiers à part

Les fichiers CSV ont d’abord été partiellement importé dans la base de données. Le script SQL « copy-csv.sql » a été créé afin d’inscrire les données des tables sans clés étrangères. Les données entières du CSV ont été placées dans une table intermédiaire *tmp\_celestial\_objects*. Une opération similaire a été faite pour les données du fichier pictures.csv mais ce fichier contient une colonne en plus pour le numéro Messier de l’objet représenté.

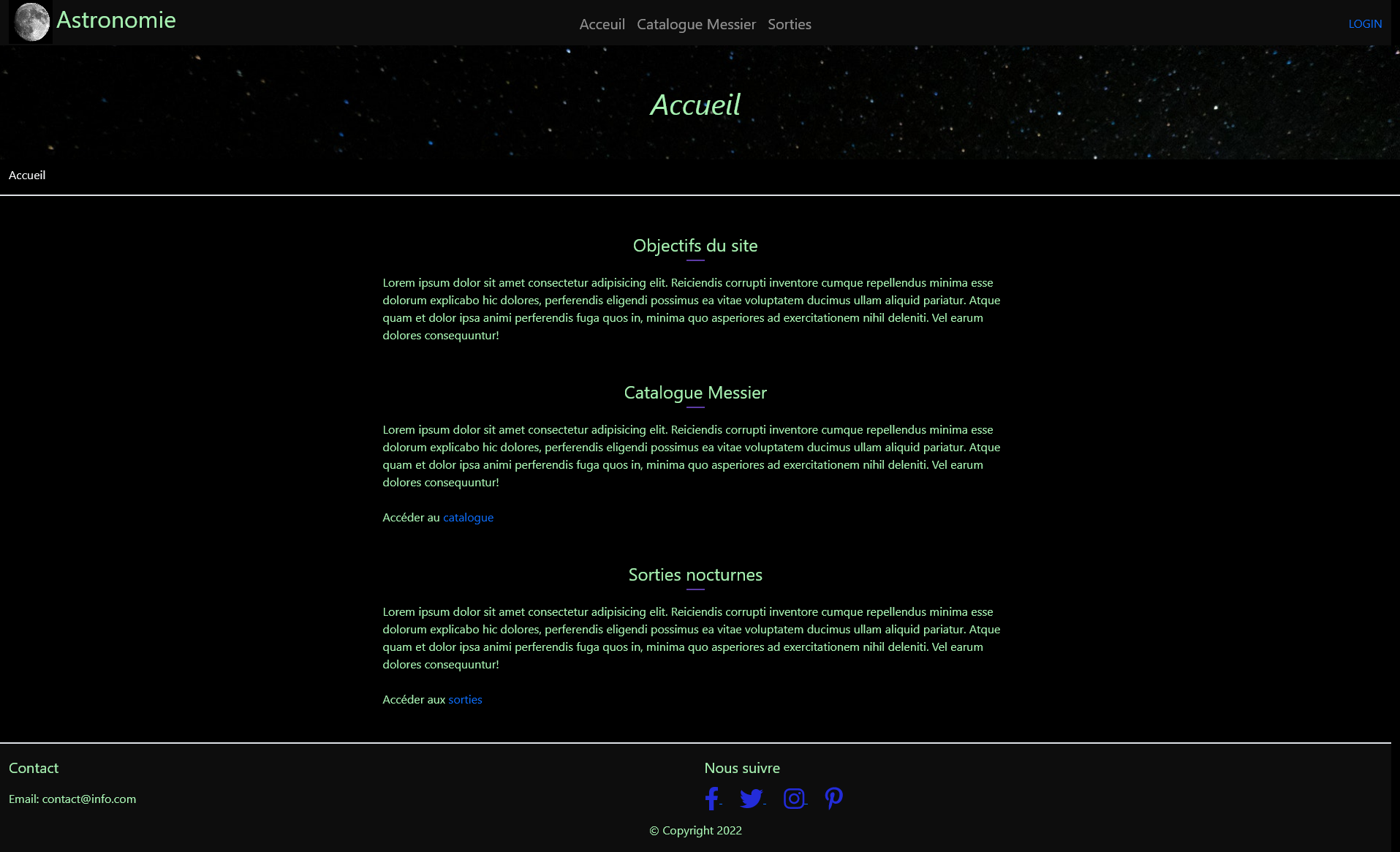


Cette dernière action a conclu l’import intégral des données du fichier CSV principal. Le transfert des données depuis les tables temporaires vers la table *celestial\_object*s a été réalisée via un script CSV. Ce script a repris les valeurs des colonnes qui ne contenaient pas de clé étrangères de la table temporaire et les a insérées directement dans la table de destination. Les clés étrangères ont été ajoutées à *celestial\_object*s en comparant les valeurs des colonnes de la table temporaire avec les colonnes ‘noms’ des tables concernées. Finalement les chemins des images ont pu être inscrit via une comparaison du numéro de Messier de l’entrée du tableau des objets célestes à celui de l’entrée de la table *tmp\_pictures*

**Site web**

Voici le rendu final des pages qui été conçu via des maquettes :

Page d’accueil :



Changer les lorem ipsum ??

## Description des tests effectués

# Annexes

## Sources – Bibliographie

|  |  |
| --- | --- |
| **Date** | **Source** |
| 03.05 |  |
|  | <https://dba.stackexchange.com/questions/86349/postgresql-csv-to-db-translate-foreign-keys> |
|  | <https://dataschool.com/learn-sql/importing-data-from-csv-in-postgresql/> |
|  | <https://stackoverflow.com/questions/61841200/i-would-like-to-select-only-unique-items-from-a-csv-file-in-powershell> |
|  | <https://docs.microsoft.com/en-us/powershell/module/microsoft.powershell.utility/export-csv?view=powershell-7.2> |
|  | <https://stackoverflow.com/questions/19450616/export-csv-exports-length-but-not-name> |
|  | <https://www.catapultsystems.com/blogs/how-to-export-and-import-foreign-characters-with-powershell/> |
|  | <https://stackoverflow.com/questions/32459269/exporting-a-variable-to-a-csv-file> |
|  | <https://dba.stackexchange.com/questions/130105/convert-a-lookup-value-to-foreign-key-when-a-query-is-run> |
|  | <https://www.postgresql.org/docs/current/sql-copy.html> |
|  | <https://dba.stackexchange.com/questions/130105/convert-a-lookup-value-to-foreign-key-when-a-query-is-run> |
|  | <https://stackoverflow.com/questions/12060546/remove-timestamp-precision-from-now-result-in-postgresql> |
|  | <https://www.postgresql.org/docs/current/datatype-numeric.html#DATATYPE-NUMERIC-DECIMAL> |
|  | <https://fr.wikipedia.org/wiki/Syst%C3%A8me_de_gestion_de_base_de_donn%C3%A9es_relationnel-objet> |
| 04.05 |  |
|  | <https://www.postgresql.org/docs/current/datatype-datetime.html> |
|  | <https://www.astronomes.com/> |
|  | <https://www.afastronomie.fr/manifestations> |
|  | <http://www.planete-astronomie.com/> |
|  | <https://hubblesite.org/science/solar-system> |
|  | <https://www.nasa.gov/topics/humans-in-space> |
|  | <https://astronomynow.com/magazine/> |
|  | <https://mdbootstrap.com/how-to/bootstrap/navbar-transparent/> |
|  | https://getbootstrap.com/docs/5.1/utilities/text/#text-transform |
|  | https://getbootstrap.com/docs/5.1/components/navbar/ |
|  | <https://www.toptal.com/developers/gitignore> |
| 05.05 |  |
|  | <https://jinja.palletsprojects.com/en/3.1.x/templates/?highlight=macro> |
|  | https://getbootstrap.com/docs/5.1/forms/overview/ |
|  | <https://stock.adobe.com/fr/252441635/> |
|  | <https://www.pexels.com/fr-fr/photo/groupe-d-etoiles-1341279/> |
|  | https://stackoverflow.com/questions/26332319/finding-local-postgresql-database-url |
| 09.05 |  |
|  | https://getbootstrap.com/docs/5.0/utilities/borders/ |
|  | https://www.pexels.com/fr-fr/photo/ciel-etoile-998641/ |
|  | https://getbootstrap.com/docs/5.0/utilities/text/#font-weight-and-italics |
|  | https://fjolt.com/article/css-states |
|  | https://www.w3schools.com/python/python\_try\_except.asp |
|  | https://docs.sqlalchemy.org/en/14/errors.html#error-e3q8 |
| 10.05 |  |
|  | https://flask.palletsprojects.com/en/2.1.x/quickstart/#sessions |
|  | Logo du site : https://www.pexels.com/fr-fr/photo/photo-de-la-lune-47367/ |
|  | Dernière date d’ajout d’un objet au catalogue Messier : <https://en.wikipedia.org/wiki/Messier_110> |
|  | PowerShell - remplacer du texte par un autre : <https://www.tutorialspoint.com/how-to-find-and-replace-the-word-in-a-text-file-using-powershell> |
|  | Template de tableau: <https://datatables.net/examples/styling/bootstrap5> |
|  | Choisir une base de données en psql : <https://www.tutorialspoint.com/postgresql/postgresql_select_database.htm> |
|  | Utilisation de chemin relatif en psql : <https://dba.stackexchange.com/questions/158466/relative-path-for-psql-copy-file> |
|  | Documentation postgres sur le type SERIAL : <https://www.postgresql.org/docs/current/datatype-numeric.html#DATATYPE-SERIAL> |
|  | Copy csv dans la bdd : <https://stackoverflow.com/questions/24483582/copy-only-some-columns-from-an-input-csv> |
| 11.05 |  |
|  | <https://stackoverflow.com/questions/40904836/how-to-get-n-files-in-a-directory-order-by-last-modified-date> |
|  | <https://stackoverflow.com/questions/52152228/how-to-filter-a-list-with-a-list-of-strings-in-powershell> |
|  | <https://stackoverflow.com/questions/11816218/renaming-files-in-powershell-using-the-folder-name> |
|  | <https://stackoverflow.com/questions/51818485/increment-variable-in-powershell-from-within-if-statement-within-a-foreach-loop> |
|  | Fichier pour créer la bdd : <https://stackoverflow.com/questions/7804698/how-to-create-sql-script-of-postgres-database-schema> |
|  | Télécharger de images de site : <https://www.wikihow.com/Download-All-Images-on-a-Web-Page-at-Once> |
| 12.05 |  |
|  | <https://docs.microsoft.com/en-us/powershell/module/microsoft.powershell.core/about/about_regular_expressions?view=powershell-7.2> |
|  | <https://stackoverflow.com/questions/54882043/powershell-variable-assignment-vs-pipeline> |
|  | <https://stackoverflow.com/questions/27970441/powershell-string-does-not-contain> |
|  | Powershell aller vers le répertoire du script <https://stackoverflow.com/questions/4724290/powershell-run-command-from-scripts-directory> |
| 16.05 |  |
|  | Types dans SQLAlchemy : <https://docs.sqlalchemy.org/en/14/core/type_basics.html> |
|  | Syntaxe command « copy » psql : <https://stackoverflow.com/questions/48872965/postgres-copy-syntax> |
|  | Chemin relatif pour copy : <https://dba.stackexchange.com/questions/158466/relative-path-for-psql-copy-file> |
|  | Jointure dans Flask-SQLAlchemy : <https://flask-sqlalchemy.palletsprojects.com/en/2.x/models/?highlight=join> |
|  | Template de tableau : <https://www.datatables.net/manual/installation>  <https://datatables.net/manual/styling/classes> |
| 17.05 |  |
|  | jQuery : <https://releases.jquery.com/> |
|  | Commande pg\_dump : <https://www.postgresql.org/docs/14/app-pgdump.html> |
|  | Doc Flask – passer des messages : <https://flask.palletsprojects.com/en/2.1.x/patterns/flashing/> |
|  | Exemple de message passé via Flask : <https://www.tutorialspoint.com/flask/flask_message_flashing.htm> |
|  | Variable d’environnement PostgreSQL : <https://www.postgresql.org/docs/current/libpq-envars.html> |
| 18.05 |  |
|  | Bootstrap 5 et les formulaires : <https://getbootstrap.com/docs/5.0/forms/input-group/> |
|  | Conventions html : <https://www.w3schools.com/htmL/html5_syntax.asp> |
|  | WTForms - Validateurs de base : <https://wtforms.readthedocs.io/en/3.0.x/validators/> |
|  | Templates HTML pour les select : <https://csshint.com/free-css-select-boxes-snippets/> |
|  | Bootstrap-select : <https://developer.snapappointments.com/bootstrap-select/>  Examples de champs <select> : <https://developer.snapappointments.com/bootstrap-select/examples/#basic-examples> |
|  |  |
|  |  |