

Table des matière

- Animanga
 - Table des versions
 - Résumé de l'énoncé
 - Organisation
 - Livrable
 - Matériel et logiques à disposition
 - Descriptif complet du projet
 - Sitemap
 - Description succincte du contenu des pages du site
 - Méthodologie
 - 1. S'informer
 - 2. Planifier
 - 3. Décider
 - 4. Réaliser
 - 5. Contrôler
 - 6. Évaluer
 - Planification
 - Product backlog
 - Diagramme de Gantt
 - Implémentation
 - Base de données
 - Dictionnaire de données
 - Plans de test et tests
 - Périmètre des tests
 - Environnement
 - Scénarios
 - Évolution des tests

Table des versions

N° de version	Date	Auteur	Modifications
0.1	2020-05-25	Tanguy Cavagna < tanguy.cvgn@eduge.ch >	Création de la base de la documentation
0.2	2020-05-26	Tanguy Cavagna < tanguy.cvgn@eduge.ch >	Ajout de la partie <i>implémentation</i> et modification des <i>user stories</i> et <i>tests</i> .

Résumé de l'énoncé

Les informations suivantes sont extraites du cahier des charges du TPI.

Organisation

Élève	Maître d'apprentissage	Experts
Tanguy Cavagna < tanguy.cvgn@eduge.ch >	Pascal Bonvin < edu-bonvinp@eduge.ch >	Nicolas Terrond < nicolas.terrond@sig-ge.ch > Robin Bouille < robin.bouille@gmail.com >

Livrable

Pour les experts et le formateur	Pour le formateur
Planning réel détaillé du projet	Accès au GitHub
Documentation du projet contenant le code source au format PDF	
Journal de bord	
Résumé du TPI (1 page A4)	

Matériel et logiques à disposition

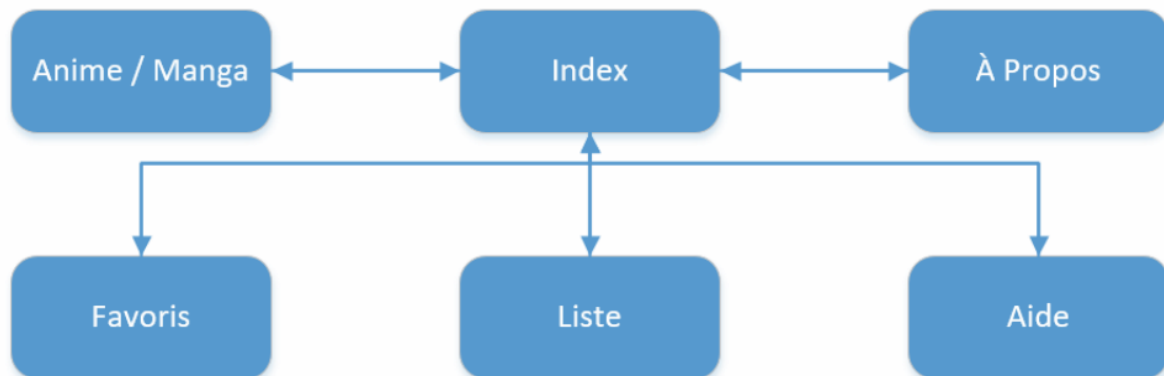
- Un PC standard école, 2 écrans
- Pycharm
- Netbeans
- Suite office
- MySQL, Sqlite3, Flask, connexion internet

Descriptif complet du projet

Lorsqu'il s'agit de réaliser un site web, la tradition de l'école d'informatique encourage l'utilisation de PHP et MySQL. Dans le cas de ce diplôme, il s'agit de réaliser un site local avec Python Flask et de gérer les données dans une base de données Sqlite3. De plus la base de donnée locale est synchronisée unidirectionnellement avec une base de donnée MySQL sur un serveur distant. L'ambition de ce projet est de démontrer qu'il est possible de créer un application WEB sans passer par l'installation d'un serveur apache et d'une base données MySQL. A noter que le candidat est un élève brillant qui maîtrise le langage Python.

Les données initiales qui permettront de remplir la base de données sont accessibles sur github au format json (<https://github.com/manami-project/anime-offline-database>). Elles devront être converties et synchronisées dans la base de données locale qui reste à déterminer.

Sitemap

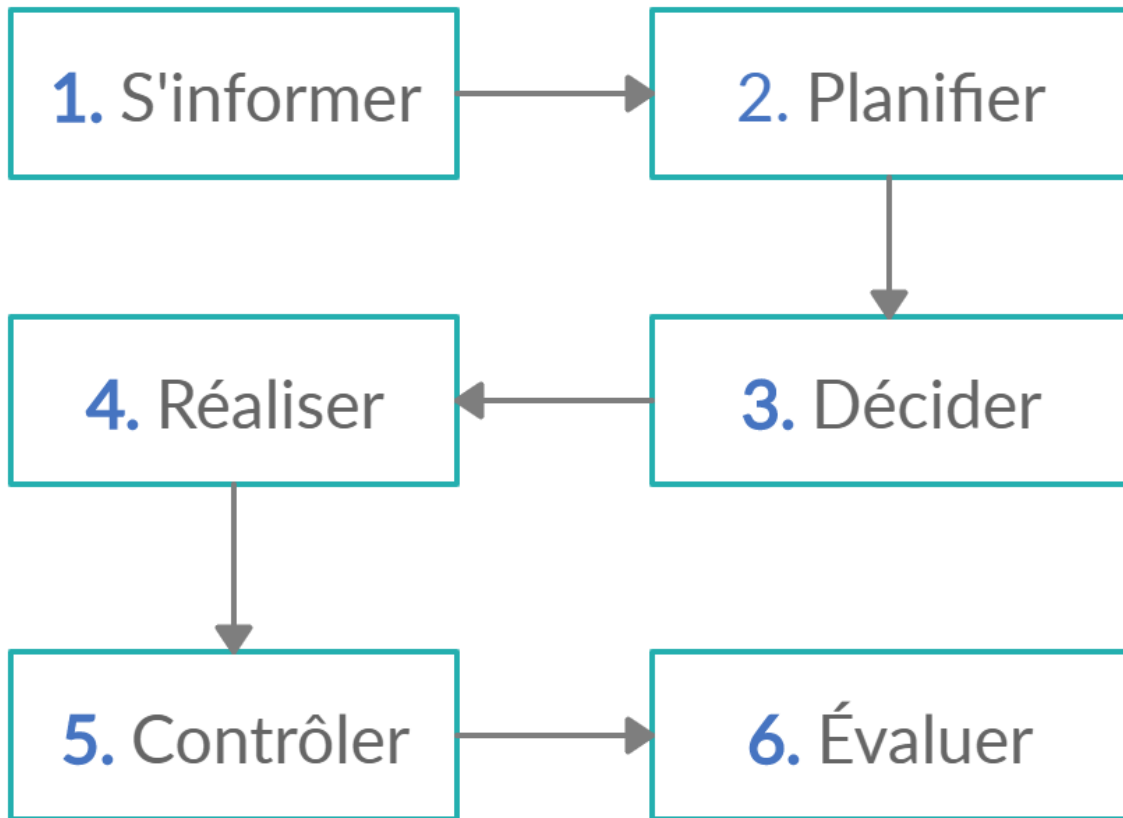


Description succincte du contenu des pages du site

- La page index permet d'avoir le champ de recherche, un bouton "aléatoire", ainsi que les favoris de l'utilisateur et son flux d'activité si connecter.
- La page à propos contient toutes les informations concernant le site ainsi que les librairies utilisées.
- La page connexion permet simplement de se connecter.
- La page inscription, de s'inscrire.
- La page anime / manga permet de voir les informations / actions sur un anime / manga sélectionner (rediriger sur cette page lorsque l'on clique sur un anime / manga sur la page index après une recherche).
- La page aide contient l'aide du site.
- La page profil contient les informations de l'utilisateur ainsi que les listes personnelles, dans lesquelles des animes / manga peuvent être ajoutés, ainsi que leur contenu.
- La page favoris permet de modifier l'ordre des animes / manga mis en favoris ainsi que de les retirer de la liste des favoris.
- La page liste contient un CRUD sur les listes personnelles de l'utilisateur.

Méthodologie

Pour pouvoir planifier correctement ce projet, j'ai décidé d'utiliser la méthode en 6 étapes, décrite ci-dessous :



1. S'informer

La première étape est utile pour pouvoir comprendre le projet dans son ensemble et comprendre toutes les fonctionnalités nécessaires. Il est aussi indispensable de demander d'éclaircir tous les points flous de l'énoncé.

2. Planifier

Le fait de planifier le projet permet de séparer les tâches et de définir des priorités. Les dernières sont les suivantes : ☹ Bloquant , ☠ Critique , ⚡ Important , 🕒 Secondaire .

Pour représenter le planning nous avons utilisé un diagramme de Gantt. Ce type de diagramme permet de visualiser très correctement la progression quotidienne ainsi que les différences entre les prévisions et le réel.

3. Décider

Cette partie nous permet de pouvoir se lancer dans la réalisation du projet. S'il nous reste des points en suspens, c'est le moment de prendre une décision et de se jeter à l'eau une bonne fois pour toute.

4. Réaliser

Nous pouvons enfin nous lancer dans l'implémentation de toutes les fonctionnalités à développer ainsi que la rédaction de la documentation.

5. Contrôler

Pour valider cette étape, nous avons testé chacune des fonctionnalités indépendamment des autres pour correctement vérifier leur fonctionnement dans différents cas d'usage.

Une fois l'application terminée, nous avons pu tester son bon fonctionnement sur plusieurs navigateurs différents pour bien être sûre que tout fonctionne comme prévu dans n'importe quel cas d'utilisation.

6. Évaluer

Une fois toutes les étapes précédentes achevées, nous avons pu nous lancer dans ce qui peut sembler le plus complexe. Nous avons fait une rétrospective de tout ce que nous avons fait avec un regard critique afin de chercher des points sur lesquels nous pourrions nous améliorer par la suite. Pour ce faire, nous avons une section dédiée dans le journal de bord répertoriant les problèmes rencontrés ainsi que les solutions trouvées pour ces derniers. Une conclusion est aussi présente à la fin de ce rapport servant de bilan final au projet.

Planification

Product backlog

Nom	S1 : Inscription à Animanga
Description (user story)	En tant qu'utilisateur non connecté, je peux me créer un compte afin de pouvoir accéder au site.
Critère d'acceptation	Les tests 1.1 et 1.2 passent.
Priorité	🛑 Bloquant

Nom	S2 : Connexion à Animanga
Description (user story)	En tant qu'utilisateur non connecté, je peux me connecter afin de pouvoir accéder au site.
Critère d'acceptation	Les tests 2.1 et 2.2 passent.
Priorité	🛑 Bloquant

Nom	S3 : Importation des animes
Description (user story)	En tant qu'utilisateur connecté, je peux écraser les animes avec un nouveau set de données.
Critère d'acceptation	Le test 3.1 passe.
Priorité	🚩 Critique

Nom	S4 : Rechercher des animes
Description (user story)	En tant qu'utilisateur connecté, je peux effectuer une recherche afin d'ajouter des animes dans mes listes personnelles ou de les mettre en tant que favoris .
Critère d'acceptation	Pas encore écrits de tests
Priorité	🚩 Critique

Nom	S5 : Affichage de la carte d'un anime
Description (user story)	En tant qu'utilisateur connecté, je peux cliquer sur le titre d'un anime présent dans les résultats de ma précédente recherche afin d'accéder à ses informations.
Critère d'acceptation	Pas encore écrits de tests
Priorité	📌 Important

Nom	S6 : Mise à jour d'un anime
Description (user story)	En tant qu'utilisateur connecté, je peux mettre à jour le statut, l'appartenance à une liste personnel ainsi que le statut de favoris d'un anime.
Critère d'acceptation	<i>Pas encore écrits de tests</i>
Priorité	🚩 Important

Nom	S7 : Affichage du profile
Description (user story)	En tant qu'utilisateur connecté, je peux avoir accès à ma page de profile afin de pouvoir voir les statistiques et favoris. Il est également possible de voir la page de profile d'autre utilisateur du site.
Critère d'acceptation	<i>Pas encore écrits de tests</i>
Priorité	🚩 Important

Nom	S8 : Affichage des listes
Description (user story)	En tant qu'utilisateur connecté, je peux avoir accès à ma page de listes afin de voir toutes mes listes et leur contenu. Il est également possible de voir les listes d'autre utilisateur du site.
Critère d'acceptation	<i>Pas encore écrits de tests</i>
Priorité	🚩 Important

Nom	S9 : Gestion des listes
Description (user story)	En tant qu'utilisateur connecté, je peux gérer mes propres listes pour en ajouter, en supprimer, ou modifier leur nom.
Critère d'acceptation	<i>Pas encore écrits de tests</i>
Priorité	🚩 Important

Nom	S10 : Affichage des favoris
Description (user story)	En tant qu'utilisateur connecté, je peux avoir accès à ma page favoris afin de voir tout mes favoris. Il est également possible de voir les favoris d'autre utilisateur du site.
Critère d'acceptation	<i>Pas encore écrits de tests</i>
Priorité	🚩 Important

Nom	S10 : Gestion des favoris
Description (user story)	En tant qu'utilisateur connecté, je peux organiser l'ordre de mes favoris selon mes envies.
Critère d'acceptation	<i>*Pas encore écrits de tests*</i>
Priorité	🚩 Important

Nom	S11 : Affichage de la <i>landing page</i>
Description (user story)	En tant qu'utilisateur non connecté, je n'ai ni accès aux animes ni aux listes. La barre de navigation m'affiche un lien pour me connecter et un autre pour m'inscrire.
Critère d'acceptation	<i>Pas encore écrits de tests</i>
Priorité	🔗 Secondaire

Nom	S12 : Utilisation d'un git
Description (user story)	En tant que développeur, je dois pouvoir faire du versionnage de code source et pouvoir accéder à un dépôt distant Github.
Critère d'acceptation	Le dépôt git local est configuré correctement et le lien sur le dépôt distant a été bien fait.
Priorité	🔒 Bloquant

Nom	S13 : Configuration de la base de données
Description (user story)	En tant que développeur, je dois pouvoir utiliser une base de données SQLite et MySQL ayant un modèle identique à celui donné dans l'énoncé. Pour ce faire, j'ai une classe Python me permettant de faire des requêtes sur la base SQLite et une autre classe me permettant de faire des requêtes sur la base MySQL. J'ai aussi un dump de la structure de la base MySQL dans les fichiers statiques de mon application.
Critère d'acceptation	Les tables <code>animes</code> , <code>status</code> , <code>type</code> , <code>url</code> , <code>list</code> , <code>user</code> , <code>list_has_anime</code> , <code>user_has_list</code> et <code>user_has_favorites</code> ont bien été créés et sont utilisables par les contrôleurs dédiés.
Priorité	🔒 Bloquant

Nom	S13 : Synchronisation MySQL SQLite3
Description (user story)	En tant que développeur, je dois pouvoir synchroniser les bases MySQL et SQLite3 unidirectionnellement pour créer un backup sur serveur distant.
Critère d'acceptation	<i>Pas encore écrits de tests</i>
Priorité	🔗 Secondaire

Nom	S15 : Configuration Flask
Description (user story)	En tant que développeur, je dois configurer l'application Flask afin d'avoir un site hébergé en local et pouvoir communiquer avec la base de données SQLite3.
Critère d'acceptation	<i>*Pas encore écrits de tests*</i>
Priorité	🔒 Bloquant

Nom	S17: Vérifications syntaxique
Description (user story)	En tant que développeur, je peux lancer la commande <code>npm run lint</code> pour vérifier la syntaxe, basé sur le preset Airbnb, des fichiers JavaScript, et la commande <code>python3 -m pylint --output-format=colored</code> pour vérifier la syntaxe des fichiers Python, basé sur les conventions PEP8.
Critère d'acceptation	Les tests 12.1 et 12.2 passent.
Priorité	🔒 Bloquant

Nom	S18: Affichage des activités
Description (user story)	En tant qu'utilisateur connecté, je vois mon fil d'actualité contenant le temps écoulé depuis l'ajout d'un favoris et l'ajout d'un anime dans une liste.
Critère d'acceptation	Pas encore de tests
Priorité	? Secondaire

Diagramme de Gantt

Jour	J1 lu.25	J2 ma.26	J3 me.27	J4 je.28	J5 ve.29	J7 ma.2	J8 me.3	J9 je.4	J10 ve.5	J11 lu.8	J12 ma.9
Lecture de l'énoncé	<div></div>										
Rédaction backlog	<div></div>										
Rédaction scénarios	<div></div>	<div></div>									
Rédaction planning	<div></div>	<div></div>									
S13 : Utilisation d'un git	<div></div>	<div></div>									
S14 : Configuration de la base de données		<div></div>									
S16 : Configuration de Flask		<div></div>									
S3 : Importation des données		<div></div>	<div></div>								
S1 : Inscription à Animanga			<div></div>								
S2 : Connexion à Animanga			<div></div>	<div></div>							
S12 : Affichage de la landing page		<div></div>		<div></div>							
S4 : Rechercher des animes				<div></div>							

Jour	J1 lu.25	J2 ma.26	J3 me.27	J4 je.28	J5 ve.29	J7 ma.2	J8 me.3	J9 je.4	J10 ve.5	J11 lu.8	J12 ma.9
S5 : Affichage de la carte de l'anime											
S6 : Mise à jour de l'anime											
S7 : Affichage du profile											
S10 : Affichage des favoris											
S8 : Affichage des listes											
S9 : Gestion des listes											
S11 : Organisation des favoris											
S18 : Affichage des activités											
S15 : Synchronisation MySQL Sqlite3											
S17 : Vérifications syntaxique											
Tests en profondeur et corrections des bugs											
Tenir à jour la documentation											
Résumé du TPI											
Finalisation / Impression											
Tenue du journal de bord											

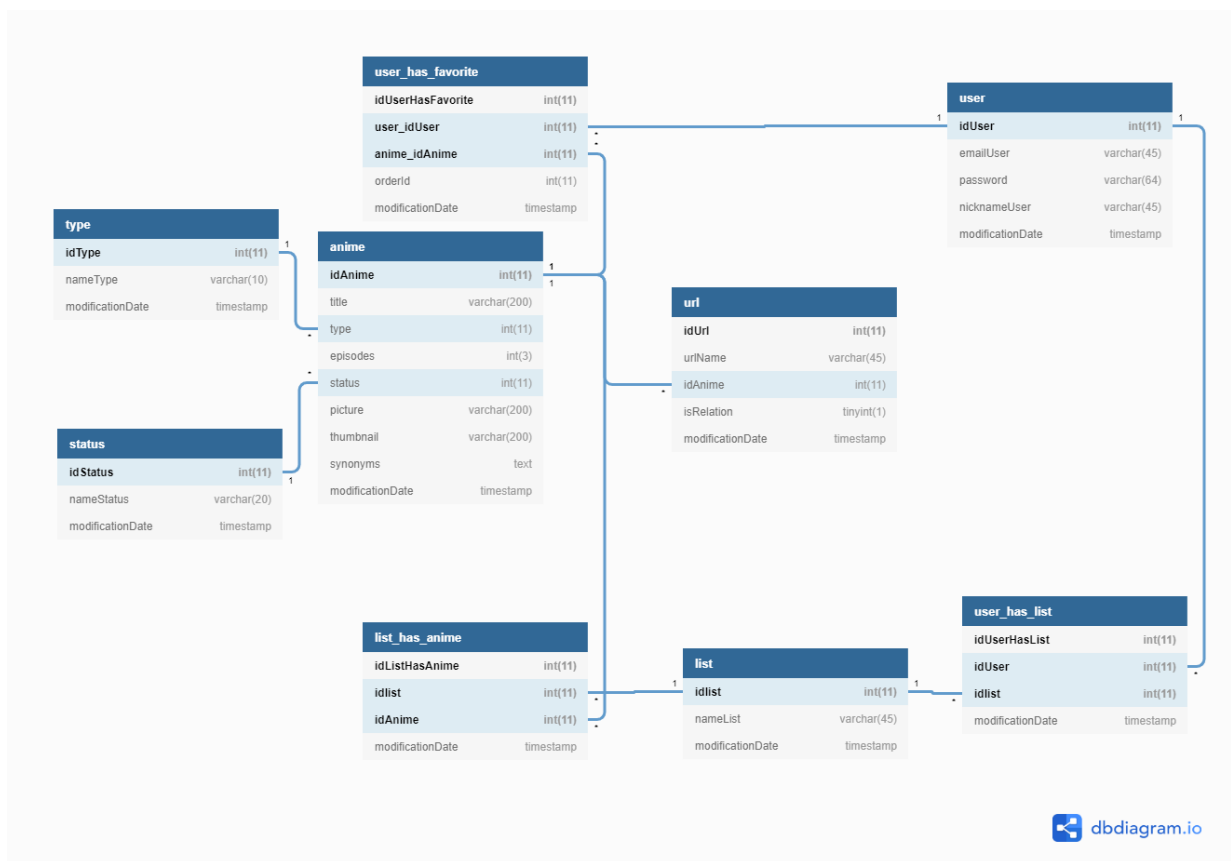
● Planification prévisionnelle
 ● Planification réelle
 ● Planification imprévue

Implémentation

Base de données

Le projet de TPI contient 2 types de base de données différents. La base principale est en Sqlite3 et la base de backup distante est en MySQL. Ces deux bases doivent pouvoir stocker les utilisateurs, les animes, les listes personnelles des utilisateurs, ainsi que les favoris des utilisateurs.

Voici le modèle réalisé :



Dictionnaire de données

anime

Colonne	Type	Null
idAnime	int(11)	Non
title	varchar(200)	Non
#type	int(11)	Non
episodes	int(3)	Non
#status	int(11)	Non
picture	varchar(200)	Non
thumbnail	varchar(200)	Non
synonyms	text	Oui
modificationDate	timestamp	Non

status

Colonne	Type	Null
idStatus	int(11)	Non
nameStatus	varchar(20)	Non
modificationDate	timestamp	Non

type

Colonne	Type	Null
idType	int(11)	Non
nameType	varchar(10)	Non
modificationDate	timestamp	Non

url

Colonne	Type	Null
idUrl	int(11)	Non
urlName	varchar(45)	Non
<i>#idAnime</i>	int(11)	Non
isRelation	tinyint(1)	Non
modificationDate	timestamp	Non

list

Colonne	Type	Null
idList	int(11)	Non
nameList	varchar(45)	Non
modificationDate	timestamp	Non

list_has_anime

Colonne	Type	Null
idListHasAnime	int(11)	Non
<i>#idList</i>	int(11)	Non
<i>#idAnime</i>	int(11)	Non
modificationDate	timestamp	Non

user

Colonne	Type	Null
idUser	int(11)	Non
emailUser	varchar(45)	Non
password	varchar(45)	Non
nicknameUser	varchar(45)	Non
modificationDate	timestamp	Non

user_has_list

Colonne	Type	Null
idUserHasList	int(11)	Non
<i>#idUser</i>	int(11)	Non
<i>#idList</i>	int(11)	Non
modificationDate	timestamp	Non

user_has_favorite

Colonne	Type	Null
idUserHasFavorite	int(11)	Non
#idUser	int(11)	Non
#idAnime	int(11)	Non
orderId	int(11)	Non
modificationDate	timestamp	Non

Plans de test et tests

Périmètre des tests

Pour *Animanga* j'ai mis en place un protocole de test afin que n'importe quel utilisateur puisse naviguer convenablement dans l'application, peu importe son navigateur WEB.

Environnement

Lors de ces tests, j'ai utilisé les navigateurs suivants :

- Mozilla Firefox 76.0.1 (64 bits) sur Windows 10 Entreprise 1903
- Google Chrome 81.0.4044.138 (64 bits) sur Windows 10 Entreprise 1903
- Microsoft Edge Version 81.0.416.72 (64 bits) sur Windows 10 Entreprise 1903

Scénarios

Les scénarios des tests sont détaillés afin que n'importe quelles personne puisse les exécuter. Pour rédiger mes scénarios j'ai utilisé la syntaxe **Gherkin** .


Nom	1.1 Création d'un nouveau compte
User Story	SI : Inscription à Animanga
Situation	Étant donné que je suis un nouvel utilisateur de Animanga, je ne possède pas encore de compte. Quand je clique sur le bouton <i>Inscription</i> , je suis redirigé vers la page d'inscription et je rempli les informations demandées. Alors , je suis redirigé sur la page d'accueil avec mon nouveau compte connecté.
Résultats obtenus	Je clique sur <i>Inscription</i> . Je suis redirigé vers la page d'inscription. Je remplit le formulaire avec des informations valides. Je clique sur <i>Inscription</i> . Je suis redirigé vers la page d'accueil avec mon nouveau compte connecté.
Statut	✓ OK

Nom	1.2 Création d'un nouveau compte (avec erreurs)
User Story	SI : Inscription à Animanga
Situation	Étant donné que je suis un nouvel utilisateur de Animanga, je ne possède pas encore de compte. Quand je clique sur le bouton <i>Inscription</i> , je suis redirigé vers la page d'inscription et je rempli le formulaire avec des informations erronées. Alors , le formulaire est rechargé avec un message d'erreur indiquant ce qu'il ne c'est pas bien passé.
Résultats obtenus	Je clique sur <i>Inscription</i> . Je suis redirigé vers la page d'inscription. Je remplit le formulaire avec un email invalide. Je clique sur <i>Inscription</i> . Un message s'affiche m'indiquant que l'email fourni n'est pas correct.
Statut	✓ OK

Nom	2.1 Connexion avec un compte existant
User Story	S2 : Connexion à Animanga
Situation	<p>Étant donné que je suis un utilisateur de Animanga, j'ai déjà un compte à disposition.</p> <p>Quand je clique sur le bouton <i>Connexion</i> , je suis redirigé vers la page de connexion et je rempli les informations demandées.</p> <p>Alors, je suis redirigé sur la page d'accueil avec mon compte connecté.</p>
Résultats obtenus	Je clique sur <i>Connexion</i> . Je suis redirigé vers la page d'inscription. Je remplie le formulaire avec des informations valides. Je clique sur <i>Connexion</i> . Je suis redirigé vers la page d'accueil avec mon nouveau compte connecté.
Statut	✓ OK

Nom	2.2 Déconnexion
User Story	S2 : Connexion à Animanga
Situation	<p>Étant donné que je suis un utilisateur connecté au site.</p> <p>Quand je clique sur le bouton <i>Déconnexion</i> placé dans le dropdown du menu <i>Utilisateur</i> .</p> <p>Alors, je deviens un utilisateur non connecté et je suis redirigé sur la page de connexion.</p>
Résultats obtenus	Je clique sur <i>Utilisateur</i> et <i>Déconnexion</i> . Je ne suis plus connecté et je suis revenue sur la page de connexion.
Statut	✓ OK

Nom	3.1 Importation des animes
User Story	S3 : Importation des animes
Situation	<p>Étant donné que je suis un utilisateur connecté au site.</p> <p>Quand je clique sur le bouton <i>Écraser tous les animes</i> placé dans le dropdown du menu <i>Utilisateur</i> .</p> <p>Alors, j'écrase toutes les données du site relatives aux animes. Cela comprend les animes en eux même, les favoris ainsi que les animes contenu dans les listes.</p>
Résultats obtenus	Je clique sur <i>Utilisateur</i> et <i>Écraser tous les animes</i> . Je suis redirigé vers la page d'accueil et des alertes s'affiche en haut au centre de l'écran indiquant l'état de la mise à jours des animes.
Statut	✓ OK

Nom	4.1 Recherche des animes
User Story	S4 : Recherche des animes
Situation	<p>Étant donné que je suis un utilisateur connecté au site.</p> <p>Quand je clique sur le bouton  placé dans la barre de navigation et que j'écris "k On" dans le champs de recherche de la modale.</p> <p>Alors, je suis redirigé vers la page d'accueil et les résultats de la recherche affiche l'anime "K-ON!".</p>
Résultats obtenus	L'anime "K-ON!" est présent dans la zone de résultat de recherche.
Statut	✓ OK

Nom	4.2 Recherche des animes avec raccourcis
User Story	S4 : Recherche des animes
Situation	<p>Étant donné que je suis un utilisateur connecté au site.</p> <p>Quand je fais le raccourcis clavier <input type="text" value="Ctrl"/> + <input type="text" value="S"/> et que j'écris "k On" dans le champs de recherche de la modale.</p> <p>Alors, je suis redirigé vers la page d'accueil et les résultats de la recherche affiche l'anime "K-ON!".</p>
Résultats obtenus	L'anime "K-ON!" est présent dans la zone de résultat de recherche.
Statut	✓ OK

Nom	5.1 Affichage de la carte de l'anime
User Story	S5 : Affichage de la carte de l'anime
Situation	<p>Étant donné que je suis un utilisateur connecté au site sur la page d'accueil ayant fait une recherche.</p> <p>Quand je clique sur le titre d'un anime présent dans la zone de résultat de la recherche.</p> <p>Alors, une modale s'affiche contenant l'image de couverture, le titre, le statut de visionnement de l'anime, et les listes personnelles de l'utilisateur.</p>
Résultats obtenus	La modale s'affiche avec le contenu adéquat.
Statut	✓ OK

Nom	6.1 Mise à jour du statut de l'anime sélectionné
User Story	S6 : Mise à jour de l'anime
Situation	<p>Étant donné que je suis un utilisateur connecté au site sur la page d'accueil ayant fait une recherche et ayant ouvert la modale d'informations d'un anime.</p> <p>Quand sélectionne un statut autre que "---".</p> <p>Alors, le combo-box se met à jour avec la nouvelle valeur sélectionnée.</p>
Résultats obtenus	La valeur du combo-box c'est bien mise à jour.
Statut	✓ OK

Nom	6.2 Ajout de l'anime dans une liste personnelle
User Story	S6 : Mise à jour de l'anime
Situation	<p>Étant donné que je suis un utilisateur connecté au site sur la page d'accueil ayant fait une recherche et ayant ouvert la modale d'informations d'un anime.</p> <p>Quand je clique sur une check-box blanche d'une des listes personnelles.</p> <p>Alors, l'état de la check-box ce met à jour et elle se colore en bleu. L'anime est maintenant présent dans la liste personnelle.</p>
Résultats obtenus	L'état de la check-box c'est bien mis à jour et est bien coloré en bleu.
Statut	✗ KO

Nom	6.3 Ajout de l'anime dans les favoris
User Story	S6 : Mise à jour de l'anime
Situation	<p>Étant donné que je suis un utilisateur connecté au site sur la page d'accueil ayant fait une recherche et ayant ouvert la modale d'informations d'un anime.</p> <p>Quand je clique sur le cœur blanc pour ajouté au favoris.</p> <p>Alors, le cœur se colore et l'anime se rajoute dans la zone des favoris de la page d'accueil.</p>
Résultats obtenus	Le cœur c'est coloré et l'anime c'est correctement ajouté dans la zone des favoris de la page d'accueil.
Statut	✓ OK

Nom	6.4 Suppression du statut de l'anime
User Story	S6 : Mise à jour de l'anime
Situation	<p>Étant donné que je suis un utilisateur connecté au site sur la page d'accueil ayant fait une recherche et ayant ouvert la modale d'informations d'un anime.</p> <p>Quand sélectionne le statut "---".</p> <p>Alors, le combo-box se met à jour avec la nouvelle valeur sélectionnée et l'anime n'est plus présent dans aucun autre statut.</p>
Résultats obtenus	La valeur du combo-box c'est bien mise à jour et l'anime n'est effectivement plus présent dans les autres statuts.
Statut	✓ OK

Nom	6.5 Suppression de l'anime d'une liste personnelle
User Story	S6 : Mise à jour de l'anime
Situation	<p>Étant donné que je suis un utilisateur connecté au site sur la page d'accueil ayant fait une recherche et ayant ouvert la modale d'informations d'un anime.</p> <p>Quand je clique sur une check-box bleue d'une des listes personnelles.</p> <p>Alors, l'état de la check-box ce met à jour et se colore en blanc. L'anime n'est plus présent dans la cette liste personnelle.</p>
Résultats obtenus	L'état de la check-box c'est bien mis à jour et est coloré en blanc.
Statut	✗ KO

Nom	6.6 Suppression de l'anime des favoris
User Story	S6 : Mise à jour de l'anime
Situation	<p>Étant donné que je suis un utilisateur connecté au site sur la page d'accueil ayant fait une recherche et ayant ouvert la modale d'informations d'un anime.</p> <p>Quand je clique sur le cœur rose pour supprimer des favoris.</p> <p>Alors, le cœur se colore en blanc et l'anime se supprime de la zone des favoris de la page d'accueil.</p>
Résultats obtenus	Le cœur c'est coloré en blanc et l'anime c'est correctement supprimé de la zone des favoris de la page d'accueil.
Statut	✓ OK


Nom	7.1 Affichage du profile de l'utilisateur connecté
User Story	S7 : Affichage du profile
Situation	Étant donné que je suis un utilisateur connecté. Quand je clique sur <i>Profile</i> dans la barre de navigation. Alors , la page de profile de l'utilisateur connecté s'affiche avec ses statistiques et ses favoris.
Résultats obtenus	La page de profile de l'utilisateur connecté s'affiche correctement et les statistiques ainsi que les favoris sont les siens.
Statut	✓ OK

Nom	7.2 Affichage du profile d'un autre utilisateur
User Story	S7 : Affichage du profile
Situation	Étant donné que je suis un utilisateur connecté. Quand je modifie l'url pour entré <i>"/profile/Test"</i> . Alors , la page de profile de "Test" s'affiche avec ses statistiques et ses favoris.
Résultats obtenus	La page de profile de "Test" s'affiche correctement et les statistiques ainsi que les favoris sont les siens.
Statut	✓ OK

Nom	8.1 Affichage des listes de l'utilisateur connecté
User Story	S8 : Affichage des listes
Situation	Étant donné que je suis un utilisateur connecté. Quand je clique sur <i>Listes</i> dans la barre de navigation. Alors , la page contenant toutes les listes de l'utilisateur connecté s'affiche ainsi que les animés contenu dans ces listes.
Résultats obtenus	La page contenant les listes de l'utilisateur connecté c'est correctement affiché et les animés sont correctement affiché aussi.
Statut	✗ KO

Nom	8.2 Affichage des listes d'un autre utilisateur
User Story	S8 : Affichage des listes
Situation	Étant donné que je suis un utilisateur connecté. Quand je modifie l'url pour entré <i>"/listes/Test"</i> . Alors , la page contenant toutes les listes de l'utilisateur "Test" s'affiche ainsi que les animés contenu dans ces listes.
Résultats obtenus	La page contenant les listes de l'utilisateur "Test" c'est correctement affiché et les animés sont correctement affiché aussi.
Statut	✗ KO

Nom	9.1 Créer une liste
User Story	S9 : Gestion des listes
Situation	Étant donné que je suis un utilisateur connecté. Quand je suis sur la page des listes et que j'écris "Ma nouvelle liste" dans le champs de texte <i>Nouvelle liste</i> et que j'appuie sur <input type="button" value="Enter"/> . Alors , la liste apparaîtra en bas des listes déjà présentes avec une 🗑 à côté.
Résultats obtenus	La liste à bien été ajoutée en base des listes déjà présente.
Statut	✗ KO

Nom	9.2 Supprimer une liste
User Story	S9 : Gestion des listes
Situation	Étant donné que je suis un utilisateur connecté. Quand je suis sur la page des listes et que je clique sur  d'une liste présente. Alors , la liste ne sera plus présente dans les listes présentes.
Résultats obtenus	La liste à bien été supprimer et n'est plus présente dans les listes déjà existante.
Statut	✗ KO

Nom	10.1 Affichage des favoris sur l'accueil
User Story	S10 : Affichage des favoris
Situation	Étant donné que je suis un utilisateur connecté. Quand je suis sur la page d'accueil. Alors , mes favoris sont présent sur la page.
Résultats obtenus	Mes favoris sont bien affiché.
Statut	✗ KO

Nom	10.2 Affichage des favoris du profile
User Story	S10 : Affichage des favoris
Situation	Étant donné que je suis un utilisateur connecté. Quand je suis sur ma page de profile. Alors , mes favoris sont présent sur la page.
Résultats obtenus	Mes favoris sont bien affiché.
Statut	✗ KO

Nom	10.3 Affichage des favoris d'un autre
User Story	S10 : Affichage des favoris
Situation	Étant donné que je suis un utilisateur connecté. Quand je modifie l'url pour entré <code>/favorites/Test</code> . Alors , les favoris de l'utilisateur "Test" s'affiche dans l'ordre que "Test" à décidé.
Résultats obtenus	Mes favoris sont bien affiché.
Statut	✗ KO

Nom	11.1 Organisation des favoris
User Story	S11 : Organisation des favoris
Situation	Étant donné que je suis un utilisateur connecté. Quand je suis sur la page des favoris et je clique sur le bouton <i>Réorganiser les favoris</i> , je peux glisser déposer les animes dans l'ordre que je veux. Je clique sur le bouton <i>Sauvegarder</i> pour enregistrer l'ordre. Alors , mes favoris sont enregistrer dans l'ordre voulu.
Résultats obtenus	Mes favoris ont bien été réorganisé.
Statut	✗ KO

Nom	12.1 Affichage de la landing page
User Story	S12 : Affichage de la landing page
Situation	<p>Étant donné que je suis un utilisateur non connecté.</p> <p>Quand je suis sur le site.</p> <p>Alors, une page d'accueil s'affiche avec comme possibilité : la visite de la page <i>À propos</i>, se connecter et s'inscrire.</p>
Résultats obtenus	La page d'accueil ainsi que la barre de navigation sont affichée correctement pour un utilisateur non connecté.
Statut	✓ OK

Nom	13.1 Respect du preset Airbnb
User Story	S17 : Vérification syntaxique
Situation	<p>Étant donné que je suis un développeur.</p> <p>Quand j'exécute la commande <code>npm run lint</code> dans le dossier de mon projet.</p> <p>Alors, aucune erreur de syntaxe sur la base du preset Airbnb n'est relevée.</p>
Résultats obtenus	<pre>> Animanga@1.0.0 lint /home/cavagnat/Documents/programmation/python/Animanga > eslint '**/*.js' --ignore-pattern node_modules/</pre>
Statut	✓ OK

Nom	13.2 Respect des conventions PEP8
User Story	S17 : Vérification syntaxique
Situation	<p>Étant donné que je suis un développeur.</p> <p>Quand j'exécute la commande <code>python3 -m pylint --output-format=colored packages</code> à la racine de mon projet.</p> <p>Alors, les erreurs <i>Too many return statements</i> n'ont pas besoin d'être prises en compte, tout comme <i>Method could be a function</i> du fichier <code>SqliteController.py</code>.</p>
Résultats obtenus	La note attribuée au code et supérieure à 9.5/10.
Statut	✓ OK

Évolution des tests

N° Test	J1 lu.25	J2 ma.26	J3 me.27	J4 je.28	J5 ve.29	J6 ma.2	J7 me.3	J8 je.4	J9 ve.5	J10 lu.8	J11 ma.9
1.1	×	×	✓	✓	✓	✓	×	×	×	×	×
1.2	×	×	✓	✓	✓	✓	×	×	×	×	×
2.1	×	×	✓	✓	✓	✓	×	×	×	×	×
2.2	×	×	✓	✓	✓	✓	×	×	×	×	×
3.1	×	✓	✓	✓	✓	✓	×	×	×	×	×
4.1	×	×	×	✓	✓	✓	×	×	×	×	×
4.2	×	×	×	✓	✓	✓	×	×	×	×	×
5.1	×	×	×	✓	✓	✓	×	×	×	×	×
6.1	×	×	×	×	✓	✓	×	×	×	×	×
6.2	×	×	×	×	×	✓	×	×	×	×	×
6.3	×	×	×	×	✓	✓	×	×	×	×	×
6.4	×	×	×	×	✓	✓	×	×	×	×	×
6.5	×	×	×	×	×	✓	×	×	×	×	×
6.6	×	×	×	×	✓	✓	×	×	×	×	×
7.1	×	×	×	×	✓	✓	×	×	×	×	×
7.2	×	×	×	×	✓	✓	×	×	×	×	×
8.1	×	×	×	×	×	✓	×	×	×	×	×
8.2	×	×	×	×	×	✓	×	×	×	×	×
9.1	×	×	×	×	×	✓	×	×	×	×	×
9.2	×	×	×	×	×	✓	×	×	×	×	×
10.1	×	×	×	✓	✓	✓	×	×	×	×	×
11.1	×	×	×	×	×	✓	×	×	×	×	×
11.2	×	×	×	×	×	✓	×	×	×	×	×
11.3	×	×	×	×	×	✓	×	×	×	×	×
12.1	×	✓	✓	✓	✓	✓	×	×	×	×	×
13.1	×	✓	✓	✓	✓	✓	×	×	×	×	×
13.2	×	✓	✓	✓	✓	✓	×	×	×	×	×