



Animanga
Rapport TPI et
documentation
technique

**mai - juin
2020**

Travail Pratique Individuel (TPI)

- Tanguy Cavagna
- Maître d'apprentissage : Pascal Bonvin

Table des matière

- Animanga
 - Table des versions
 - Résumé de l'énoncé
 - Organisation
 - Livrable
 - Matériel et logicles 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

Table des versions

Nº de version	Date	Auteur	Modifications
0.1	2020-25-05	Tanguy Cavagna < tanguy.cvgn@eduge.ch >	Création de la base de la documentation

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-bon.vinp@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	
Documentation du projet contenant le code source au format PDF	
Journal de bord	
Résumé du TPI (1 page A4)	Accès au GitHub

Matériel et logiciels à disposition

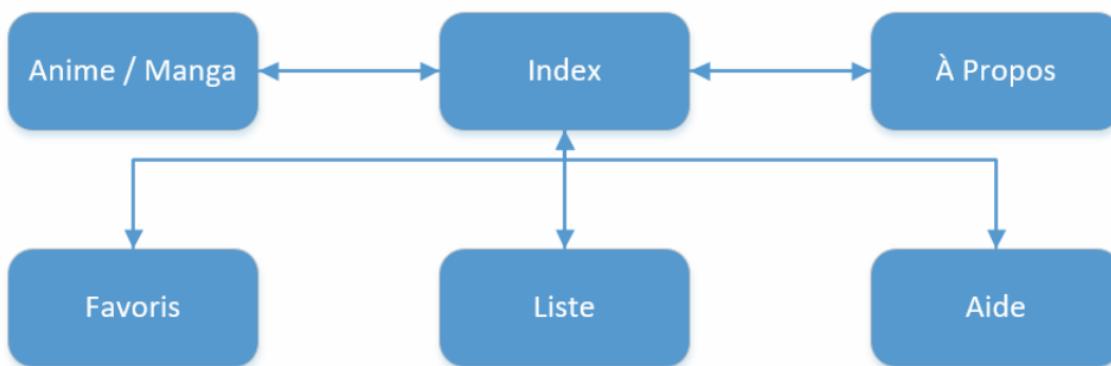
- Un PC standard école, 2 écrans
- Pycharm
- Netbeans
- Suite office
- MySQL, Sqlite3, Flask, connexion internet

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

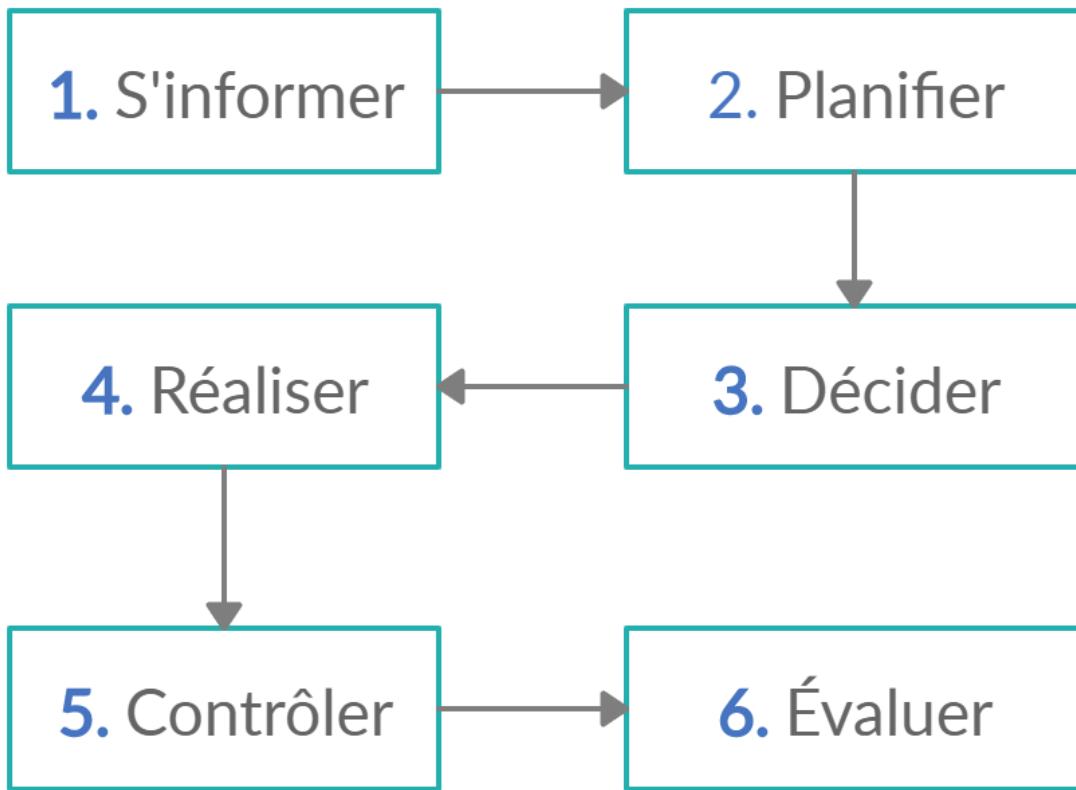


Description succincte du contenu des pages du site

- 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 information 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'utilisé 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. Ses 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. Si 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 tester 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 pourront nous améliorer par la suite. Pour ce faire, nous avons une section dédiée dans le journal de bord répertoriant les problèmes rencontré 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.

Nom	S1 : Inscription à Animanga
Description (<i>user story</i>)	En tant qu'utilisateur non connecté, je peux me créer un compte afin de pouvoir accéder au site.
Critère d'acceptation	<i>Pas encore écrits de tests</i>
Priorité	⌚ Bloquant

Nom	S2 : Connexion à Animanga
Description (<i>user story</i>)	En tant qu'utilisateur non connecté, je peux me connecter afin de pouvoir accéder au site.
Critère d'acceptation	<i>Pas encore écrits de tests</i>
Priorité	⌚ Bloquant

Nom	S3 : Importation des animes
Description (<i>user story</i>)	En tant qu'utilisateur connecté, je peux écraser les animes avec un nouveau set de données.
Critère d'acceptation	<i>Pas encore écrits de tests</i>
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 <i>favoris</i> .
Critère d'acceptation	<i>Pas encore écrits de tests</i>
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	<i>Pas encore écrits de tests</i>
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 pourvoir 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 animés 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 (<i>user story</i>)	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



Annexes



Résumé TPI



Énoncé



Journal de bord

Journal de bord TPI - Tanguy CAVAGNA

J1 : lundi 25 mai 2020

Objectifs

L'objectif de cette journée est de lire l'énoncé dans son intégralité afin de prendre connaissance du cahier des charges, extraire les *user stories* de ce dernier pour pouvoir correctement rédiger mon *product backlog* et enfin rédiger les scénarios de tests fonctionnels, indispensable pour le bon fonctionnement de mon projet.

Déroulement

Je commence ma journée à 8h00. M. Terrond m'a fait parvenir mon énoncé la veille, que j'ai lu avec attention ce dernier. Par ce biais, j'ai complété avec succès la première étape de la **méthodologie en 6 étapes**, méthodologie que je vais utiliser durant tout le déroulement de ce TPI : *S'informer*.

J'ai quelques points incertains concernant mon énoncé dont un quelque peu embêtant. Je poserai mes questions à mon formateur durant la matinée. Je vais maintenant commencer à *Planifier*, secondes étape de la méthodologie utilisée. Je separerai mes journées en tranche de 4 heures, soit par demi-journée, et remplirai des différentes tranches horaires avec les *user stories* extraites de mon cahier des charges.

8h15 : J'ai décidé d'utiliser des alias afin de nommer les jours de travail mis à disposition pour le TPI. Les jours seront nommer de **J1** à **J11**. Voici les alias :

- J1 : lundi 25 mai 2020
- J2 : mardi 26 mai 2020
- J3 : mercredi 27 mai 2020
- J4 : jeudi 28 mai 2020
- J5 : vendredi 29 mai 2020
- J6 : mardi 2 juin 2020
- J7 : mercredi 3 juin 2020
- J8 : jeudi 4 juin 2020
- J9 : vendredi 5 juin 2020
- J10 : lundi 8 juin 2020
- J11 : mardi 9 juin 2020

8h25 : Lors de la création des *user stories* j'ai remarqué qu'il me fallait décider d'une manière de priorisé les tâches. J'ai opté pour me basé sur la méthode **MoSCoW**. Cependant les niveaux de priorité ne correspondaient pas entièrement pour un TPI. J'ai alors décider de modifier les intitulé :

- Must devient ☈ Bloquant
- Should devient ☷ Critique
- Could devient ⚡ Important
- Won't devient ? Secondaire

J'ai aussi décidé d'utiliser la syntaxe suivante afin de présenter mes *user stories* :

Nom	S<n° de la story> : <Nom de la user story>
Description (<i>user story</i>)	<Description de la story pour connaître avec précision le but à atteindre>
Critère d'acceptation	<n° des tests à passé pour valider cette story>
Priorité	<Priorité de la story>

Bilan



Code source