

# **TEAM JT : DWARF**

Groupe 7



MARTIN Joris  
JHISTARRY Thomas  
INZOUDINE Ayache

REMY Emeryck  
SIGNORET Tanguy  
PAPPALARDO Mattéo



# Sommaire :

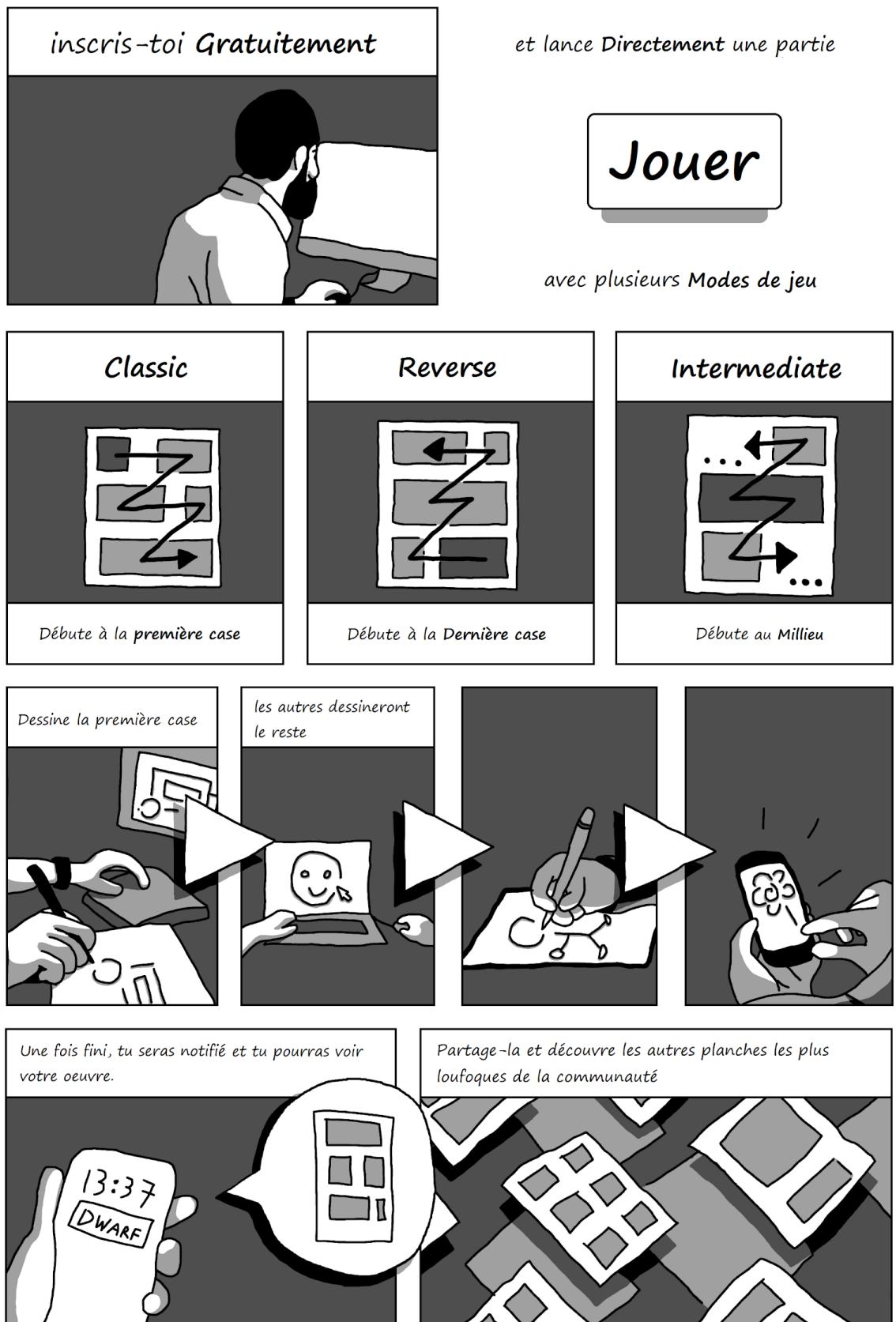
<b>Storyboard</b>	<b>2</b>
<b>Cadrage du projet</b>	<b>5</b>
Objet du projet	5
L'objet du projet	5
Nom et logo du projet	5
Parties prenantes	5
Maîtrise d'ouvrage	5
Maîtrise d'oeuvre	5
Objectifs du projet	6
Objectifs principaux	6
Objectifs secondaires	6
Analyse de l'existant	7
Les contraintes	11
Les risques	12
<b>Expression du besoin</b>	<b>14</b>
Ciblage	14
Les besoins	14
Hiérarchisation	15
Besoins fonctionnels	15
Besoins non-fonctionnels	15
Besoins techniques	15
Les critères de qualité	16
Qualité logicielle	16
Qualité ergonomique	17
Critère de test	18
Tests utilisateurs	19
Retour de tests	19
Analyse des retours	19
<b>Solutions techniques envisagées</b>	<b>23</b>
Langages et utilisation	23
Caractéristiques techniques	24
Caractéristiques de notre application	24
Caractéristiques du serveur	25
<b>Organisation du travail et découpage du projet</b>	<b>25</b>
Mode de fonctionnement	25
Outils	27
<b>L'application</b>	<b>27</b>
DWARF	27
Screenshot de l'application	28
<b>Annexes</b>	<b>30</b>

# 1. Storyboard

Commençons par vous présenter de manière intuitive le principe même de notre application à travers ce storyboard.



**Figure 0 :** Storyboard page 1



**Figure 0 bis :** Storyboard page 2

## 2. Cadrage du projet

### a. Objet du projet

#### I. L'objet du projet

L'objectif du projet est de réaliser une application web pour concevoir une planche de bande dessinée en collaboration. Chaque utilisateur devra réaliser une case d'une planche, en ayant connaissance uniquement de la précédente ou suivante, suivant le mode de jeu sélectionné. Le projet s'inspire donc du concept du jeu intitulé "cadavre exquis".

Le produit sera gratuit et accessible à tous via un navigateur web ou une application mobile avec une connexion internet.

#### II. Nom et logo du projet

Le projet se nomme DWARF. C'est un acronyme pour "**D**raw **W**ith **A**mazing **R**andom **F**riends". Le logo représente donc un nain portant un crayon à la place de l'iconique pioche, agrémenté du slogan "Tous à la mine !!!" (jeu de mot avec mine de crayon) (*cf. Figure 1*).



**Figure 1 :** Logo de DWARF

### b. Parties prenantes

#### I. Maîtrise d'ouvrage

Par le sujet de ce module, les professeurs représentent la maîtrise d'ouvrage, c'est à eux que nous faisons les revues d'avancement et à qui nous livrerons le produit au final.

#### II. Maîtrise d'oeuvre

La maîtrise d'oeuvre de projet DWARF est notre équipe développeurs et concepteurs, composée de :

- **Ayache INZOUDINE** : rédaction dossier, base de données
- **Thomas JHISTARRY** : base de données, développement back-end
- **Joris MARTIN** : rédaction dossier, conception, développement full-stack
- **Mattéo PAPPALARDO** : rédaction dossier, développement full-stack

- **Emeryck REMY** : design, ergonomie, conception
- **Tanguy SIGNORET** : chef de projet, rédaction dossier, conception, développement front-end

## c. Objectifs du projet

### I. Objectifs principaux

Notre objectif premier est de réaliser une application fonctionnelle sur navigateur web. Cela implique donc :

- La réalisation d'une case de la planche à l'aide d'un environnement de dessin simple, efficace et ergonomique.
- La conception d'une planche de BD via une interface de création de partie.
- La visualisation et la notation des planches réalisées.
- La recherche d'une planche spécifique ou non afin de la continuer à son tour.
- L'inscription et la connexion à l'application.
- L'implémentation d'un mode de jeu.
  - Normal : La case de départ est située au début de la BD et l'avancement de cette dernière se fait dans le sens "normal" (début vers la fin).

### II. Objectifs secondaires

Dans un second temps, si tous les objectifs principaux sont accomplis, nous envisageons d'ajouter certaines options à notre application, notamment :

- L'adaptation de l'application en format natif sur tablette et smartphone.
- L'ajout, à l'interface de dessin, d'un outil permettant l'incorporation d'éléments de dessins pré-existants (Exemples : Personnage, gommettes, ...).
- L'ajout d'un mode de jeu supplémentaire :
  - Reverse : Création de la planche en commençant par la dernière case.
  - Intermediate : La case de départ est située au milieu de la BD, et l'utilisateur peut choisir de continuer avant ou après.
- La possibilité d'avoir plusieurs modes de visualisation des planches comme :
  - Un visionnage classique (statique).
  - Un visionnage des planches sous forme d'un livre où l'on tourne les pages.

- Un visionnage de l'histoire imaginée par les utilisateurs sous forme de flipbook.

#### d. Analyse de l'existant

Après des recherches approfondies sur l'environnement de notre projet (la bande dessinée), nous n'avons trouvé aucune application/site web correspondant exactement à notre idée.

Il existe cependant plusieurs applications de dessin synchrone, ainsi que plusieurs applications assistant à la création de bandes dessinées. Le tableau 1 ci-dessous présente trois de ces applications :

Nom	Description	Points intéressants	Critiques
<b>Draw Together!</b>	<p>Application mobile, petit jeu de cadavre exquis en multijoueur synchrone : une personne dessine la tête, une autre le corps d'un animal / personnage imposé, le tout en 30s chacun.</p> <p>Le résultat s'affiche et les joueurs peuvent réagir à leur création.</p> <p>Une galerie est présente sur la page d'accueil, montrant les dessins récents.</p> <p>(cf. <b>Figure 2</b>)</p>	<p>Rapide, ergonomique, simple.</p> <p>Réactions au dessins produit à la fin (emojis).</p>	<p>Le jeu est inutilement synchrone.</p> <p>Le concept est vite "essoufflé", le personnage à dessiner n'est découpé qu'en 2 parties (tête et corps).</p> <p>Les personnages / animaux à dessiner sont tirés d'une liste fournie par l'application, c'est-à-dire que le jeu peut vite tourner en rond et redonner les mêmes sujets.</p>

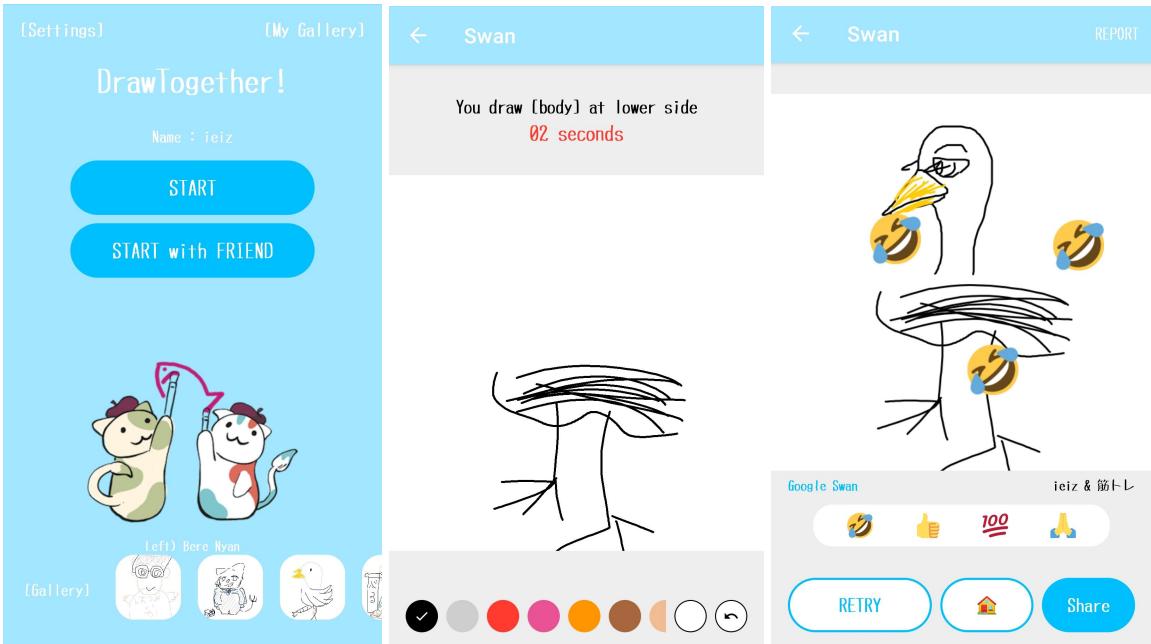
	<p>Site web, jeu de dessin en multijoueur synchrone. Une personne dessine une chose qu'il a choisi parmi une liste de propositions et les autres joueurs doivent deviner ce qu'il dessine en tapant le mot correspondant.</p> <p>(cf. Figure 3)</p>	<p>Amusant, rapide, ergonomique.</p> <p>Plusieurs langues sont disponibles pour les mots à (faire) deviner.</p> <p>Petits avatars amusants.</p>	<p>Les œuvres réalisées sont éphémères.</p>
	<p>Site web permettant de créer des bandes dessinées à partir d'une banque d'images comportant :</p> <ul style="list-style-type: none"> <li>• des personnages personnalisables</li> <li>• des arrière-plans de cases</li> <li>• des bulles de texte</li> <li>• des formes, des pictogrammes</li> <li>• des images importées par l'utilisateur</li> </ul> <p>(cf. Figure 4)</p>	<p>Interface très ergonomique.</p> <p>Il est très facile d'arriver à un résultat convaincant.</p> <p>Aucune compétence en dessin n'est nécessaire.</p> <p>Disposition des cases modifiable à la volée.</p>	<p>Impossible d'ajouter un dessin fait main sur le site, il faut importer une image depuis son ordinateur.</p> <p>Pas d'espace de partage des storyboards réalisés.</p>

Tableau 1

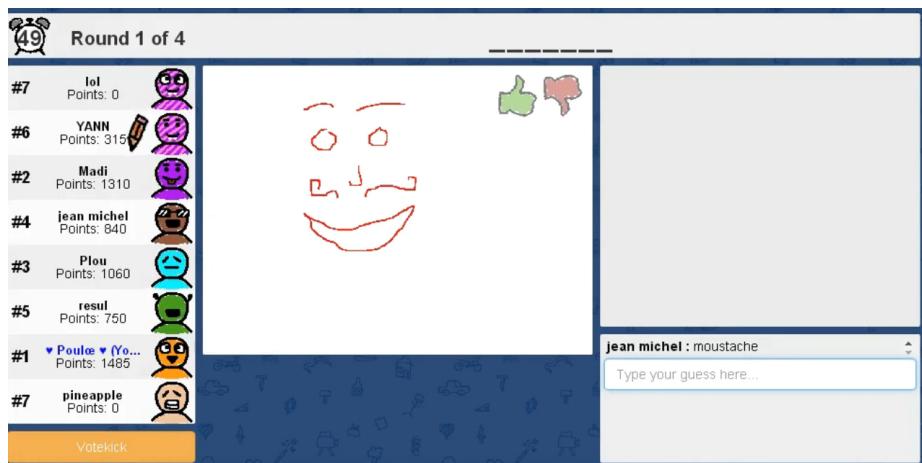
Il existe aussi plusieurs applications permettant de tout simplement dessiner. En voici un petit exemple avec une courte présentation de Gimp (le logiciel avec lequel nous avons notamment réalisé notre storyboard) :

Nom	Description	Points intéressants	Critiques
	<p>Logiciel Open Source permettant la réalisation de composants graphiques complexes grâce à de nombreux outils de dessin.</p> <p>(cf. Figure 5)</p>	<p>Le logiciel est très complet.</p> <p>Permet la réalisation de dessins de grande qualité.</p>	<p>Difficile à prendre en main pour un débutant, certaines manipulations pourtant simples ne sont pas intuitives.</p>

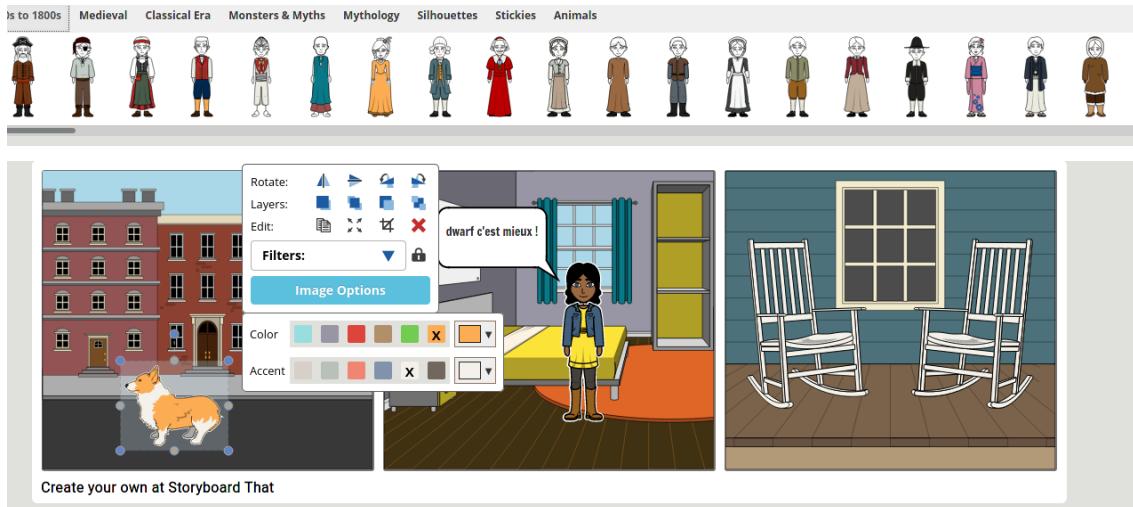
Tableau 1 bis



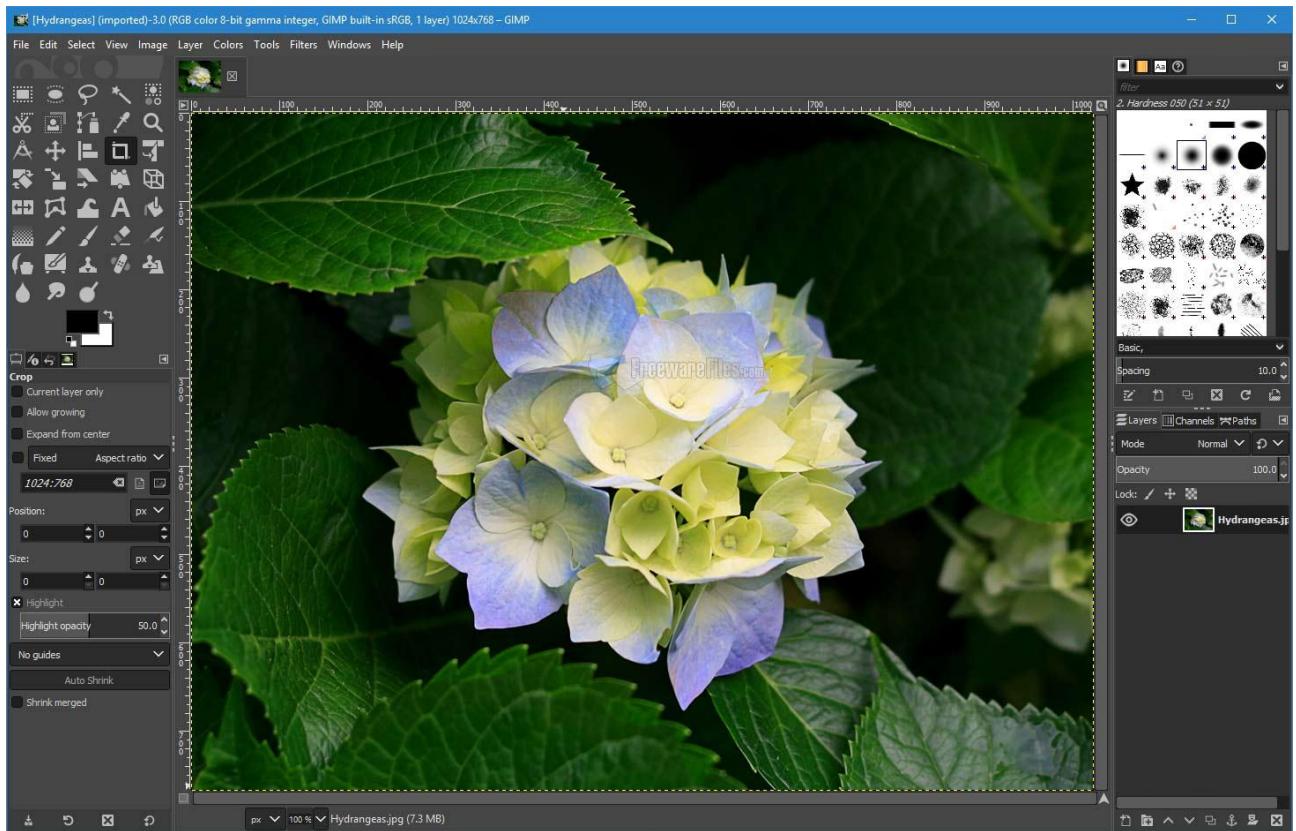
**Figure 2 :** Trois captures d'écrans de l'application “DrawTogether!”. La première présente le menu principal de l'application. La deuxième présente l'interface de jeu et de dessin, où le but ici était de dessiner le corps d'un cygne. La troisième présente l'interface de fin, où le dessin final est présenté aux deux joueurs.



**Figure 3 :** Une capture d'écran présentant l'interface de jeu de Skribbl.io. L'avatar ayant un crayon à côté de son nom est celui qui est actuellement en train de dessiner. Les autres joueurs doivent taper le mot correspondant au dessin dans la zone de texte en bas à droite.



**Figure 4 :** Une capture d'écran de l'interface de création de bandes dessinées de "StoryboardThat". Les différentes images, bulles, environnements, etc. peuvent être choisies en haut de l'application, et la bande dessinée elle-même ainsi que les options de personnalisations sont en bas.



**Figure 5 :** Voici l'interface du logiciel GIMP. On retrouve un espace de travail central. Et sur les côtés, divers outils de dessins ou de correction d'image.

## e. Les contraintes

Durant l'élaboration de cette première partie du projet, il nous est apparu plusieurs contraintes.

Tout d'abord, les règles concernant **l'utilisation des données des utilisateurs**, leur stockage et leur transfert doivent s'effectuer de manière sécurisée. De plus, il nous a semblé important d'être vigilant concernant la publication des œuvres collaboratives réalisées dans le cadre de notre application. C'est pourquoi nous avons fait le choix de placer chaque image réalisée sous une licence Creative Commons qui autorise des personnes tiers à utiliser ces derniers. Il existe différents modèles de cette licence. Nous avons retenu le modèle **CC-BY-NC** qui autorise le partage et la modification, mais pas l'utilisation à des fins commerciales, car c'est celui qui est le plus adapté à la situation décrite.

Au vu de la définition du projet, il paraît essentiel, de bien se rappeler, que l'application doit être **disponible, accessible et facilement utilisable** sur un maximum de supports.

Dans ce même cadre, il nous faudra garder à l'esprit que l'application doit être **supportée par un maximum de plateformes**, de navigateurs, y compris pour des utilisateurs ne possédant pas de supports avec une grande puissance de calcul.

De plus, il est important de garder à l'esprit que les utilisateurs **ne doivent pas être bloqués** si jamais ils se retrouvent **sans connexion internet**.

Comme exprimé précédemment, notre application nécessitera un système de **modération à posteriori** en lien avec la possibilité d'apparition de contenu inapproprié au sein de l'application. Cela implique aussi la mise en place d'un système de blacklist des pseudonymes.

Nous devrons aussi s'assurer que les données de nos utilisateurs soient **stockées de manière sécurisée** et en respectant les principes exposés dans le Règlement Général sur la Protection des Données (RGPD).

Nous sommes aussi confrontés à une **contrainte budgétaire** qui nous oblige à n'avoir qu'un seul serveur supportant l'ensemble de l'application ainsi que ses différents composants.

Enfin, il faudra anticiper la possibilité de travail à distance afin de conserver notre équipe dans son intégralité face au risque de la **COVID-19**.

## f. Les risques

Il existe plusieurs risques en lien avec notre projet :

**Compatibilité** : Notre application sera en partie disponible sur navigateur internet. Certains de ces derniers peuvent se retrouver incompatibles avec les technologies utilisées dans le fonctionnement de l'application et donc rendre cette dernière incompatible.

**Contenu** : Le risque d'avoir des publications inappropriées sur l'application. Cette dernière étant ouverte à tout public, y compris les personnes mineures, il est possible que ces personnes se retrouvent en face de contenu choquant ou blessant. Un système de modération doit donc être implémenté, qui permettra de modérer les pseudonymes des utilisateurs ainsi que les dessins qu'ils réaliseront. Ce système sera accessible par les administrateurs. Ces derniers auront accès à toutes les données de l'application (utilisateur, planche et case), sur lesquels ils pourront agir.

**Matériel** : Un risque matériel lié à la machine qui héberge l'application, si cette dernière s'avère hors d'état de marche, notre application ne serait plus disponible aux utilisateurs. Il faudrait donc envisager d'avoir une machine de secours, qui prendra le relais en cas de panne de la machine principale.

**Équipe** : Étant donné le contexte actuel, notre équipe est confrontée au risque de la perte d'un membre si ce dernier s'avère en contact de la COVID-19. Il faudra donc anticiper la possibilité de travail à distance afin de conserver notre équipe dans son intégralité face au risque COVID-19 grâce à des logiciels tels que Zoom ou Discord.

**Public** : Notre application cible un public très général et donc des personnes de toute tranche d'âge. Le design de l'application peut donc rencontrer un risque de manque d'ergonomie.

**Données** : Les données sensibles des utilisateurs peuvent être interceptées ou extraites, ce qui représente un risque important concernant la protection des données. Ces données devront donc impérativement suivre le Règlement Général sur la Protection des Données. (RGPD)

**Connexion** : Les utilisateurs auront recours à une connexion internet afin d'afficher les réalisations ou même afin de soumettre sa réalisation. Nous devons donc prendre en compte le risque de perte de connexion internet auprès des utilisateurs grâce à un système de mise en cache.

**Retard** : Nos délais de rendus sont restreints et parfois courts. Il existe donc un risque de retard lors de la soumission des rendus.

**Fonctionnalité** : Par corrélation avec le dernier risque énoncé, notre application peut s'avérer non-fonctionnelle ce qui représente un risque majeur car notre objectif premier est de rendre cette dernière utilisable et fonctionnelle.

#### Stratégie de mitigation utilisé pour ces risques :

**Compatibilité** : acceptation ⇒ Notre application sera adaptée afin qu'elle soit disponible sur un maximum de supports, mais si toutefois un navigateur se montre incompatible, nous n'aurons pas d'autre choix que de l'accepter.

**Contenu** : protection ⇒ Modération périodique des différentes publications postées.

**Matériel** : acceptation ⇒ Comme décrit dans les contraintes, nous ne pouvons pas avoir un serveur de secours, nous acceptons donc ce risque.

**Équipe** : réduction ⇒ Le contexte actuel (confinement) réduit le risque lié à la COVID-19 mais il n'est pas inexistant.

**Public** : réduction ⇒ Nous allons faire tester notre application à un large public afin de réduire au maximum ce risque.

**Données** : évitement et réduction ⇒ Nous allons sécuriser au maximum le stockage et le transit des informations qui ont lieu sur notre application.

**Connexion** : acceptation ⇒ Ce risque dépend uniquement de l'utilisateur, nous ne pouvons pas agir sur ce dernier et donc nous devons l'accepter.

**Retard** : évitement ⇒ La mise en place d'un Trello et d'un diagramme de GANTT nous permettent d'éviter ce risque. (cf. Annexe 3, Figure 7).

**Fonctionnalité** : réduction ⇒ L'ensemble des méthodes et outils de travail instauré nous permettent de réduire au maximum ce risque.

		Impact		
		Faible	Modéré	Fort
Probabilité	Faible	Compatibilité	Public	Contenu - Fonctionnalité
	Modéré	Équipe	Contenu - Retard	Données
	Fort	Connexion		

Tableau 2 : Matrice de criticité des risques

### 3. Expression du besoin

#### a. Ciblage

Notre application se positionne comme un moyen de divertissement pour tous types d'âges, ne demandant pas de compétences d'artiste pour participer, utilisable partout et à n'importe quel moment.

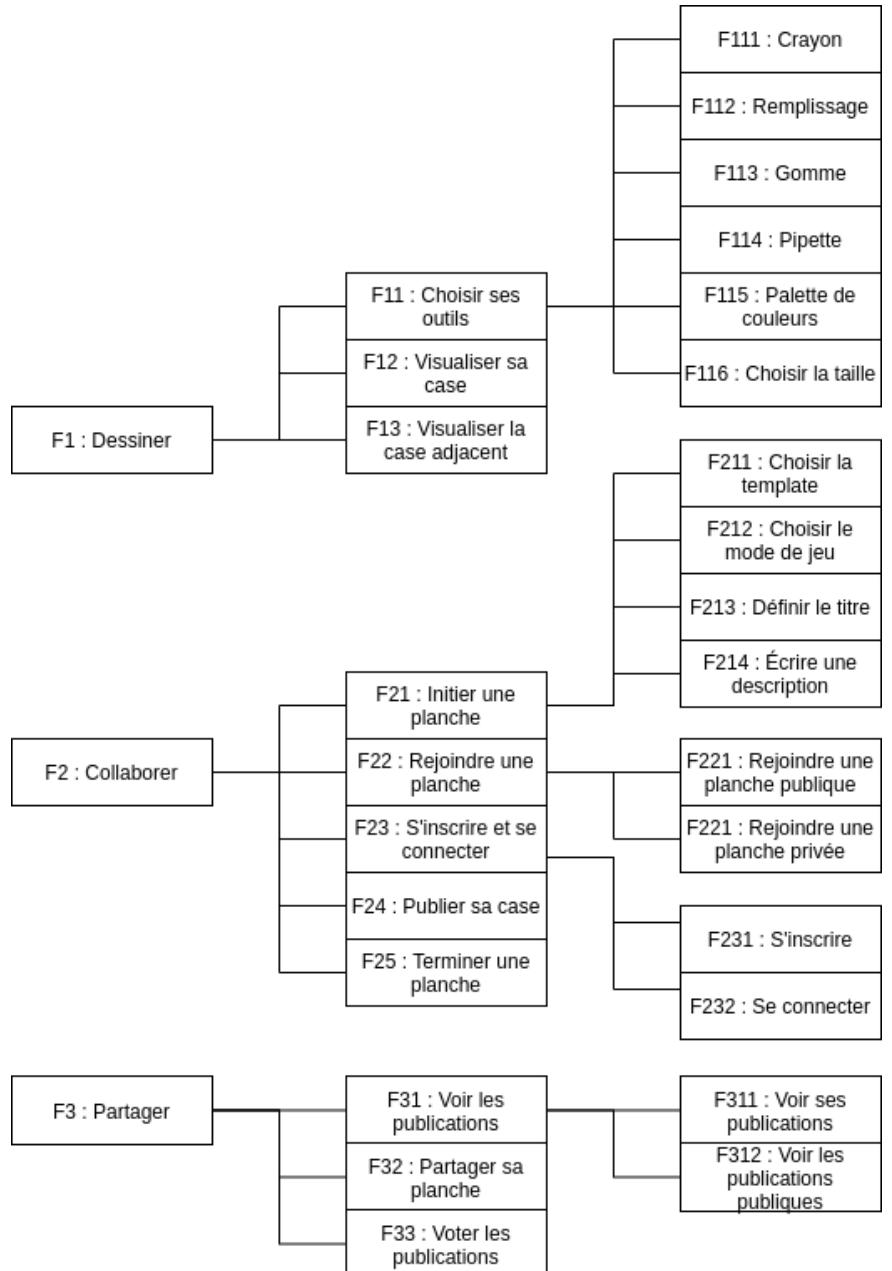
Notre cible est donc très large : nous souhaitons toucher les plus jeunes comme les plus âgés, de tous horizons, afin de réunir les générations autour d'un jeu familial.

#### b. Les besoins

Nous identifions trois besoins principaux: **Dessiner**, **Collaborer** et **Partager**.

Ces trois besoins principaux s'expriment à leur tour des besoins secondaires. Par exemple, pour dessiner, il faut pouvoir choisir ses outils, visualiser sa case ou encore visualiser la case adjacente. Ces besoins secondaires peuvent à leur tour exprimer d'autres besoins.

L'ensemble des besoins est donc modélisé à travers un diagramme (*cf. Figure 6*).



**Figure 6 :** Diagramme FAST des besoins.

## I. Hiérarchisation

L'ordre d'importance de nos besoins suit l'ordre décrit dans le diagramme FAST (*cf. Figure 6*), c'est à dire :

- 1 - Dessiner : L'utilisateur doit pouvoir réaliser un dessin sur l'application.
- 2 - Collaborer : L'utilisateur doit pouvoir initier une nouvelle partie ou réaliser.
- 3 - Partager : L'utilisateur doit pouvoir visualiser les créations finies et en cours.

## II. Besoins fonctionnels

Afin que notre application soit fonctionnelle, il faut que les utilisateurs puissent :

- Avoir accès à une interface de dessin permettant la réalisation du premier dessin d'une planche ou d'un dessin intermédiaire en pouvant visualiser la création précédente.
- Collaborer à plusieurs sur une même planche de bande dessinée.
- Avoir accès à une interface permettant de naviguer entre les planches finalisées et les planches en cours de réalisation.
- Avoir accès à une interface de connexion et d'inscription permettant la connexion à son compte ou l'enregistrement d'un nouveau compte.

## III. Besoins non-fonctionnels

Pour que notre application réponde aux attentes de l'utilisateur, il faut que :

- Elle puisse être utilisée sur un maximum de support :
  - PC : plusieurs navigateurs, plusieurs tailles d'écran
  - Tablettes : plusieurs systèmes d'exploitations, plusieurs tailles d'écran
  - Smartphones : idem
- Les interfaces soient ergonomiques et intuitives.
- Les données (planches, cases et utilisateurs) soient conservées de manière sécurisée.
- La navigation soit rapide et fluide.
- L'affichage soit correct pour les différents composants.

## IV. Besoins techniques

Au vu des besoins fonctionnels et non fonctionnels exprimés précédemment, il apparaît certains besoins techniques :

- Un serveur qui pourra supporter l'application, son fonctionnement en arrière-plan et une base de données.
- Un appareil permettant l'utilisation de l'application comme un ordinateur, smartphone ou tablette.
- Une connexion internet.

## c. Les critères de qualité

### I. Qualité logicielle

De manière à fournir une application sûre, nous avons décidé d'utiliser les critères de qualité logiciel de la norme ISO 9126 (critères SQUARE). Ils nous apparaît important de détailler certains d'entre eux :

- **La fiabilité** : Comme notre application sera disponible sur tout type de support, elle visera donc un public très large et elle se doit donc d'être fiable. Cela signifie qu'elle doit avoir une certaine tolérance aux fautes du côté utilisateur. Cela inclut aussi la capacité de récupération, que ce soit des données utilisateur ou de jeux.
- **La sécurité** : Comme notre produit a pour but d'être disponible et distribué à un plus grand nombre de personnes, nous nous devons d'être particulièrement attentif à la sécurité de ce dernier.

Pour cela, nous avons mis en place un firewall sur le serveur que nous utilisons, afin de filtrer les requêtes reçues et envoyées.

Nous avons configuré le .htaccess en fonction du composant "React Router", ce qui implique qu'il est impossible de se déplacer dans le système de fichier du serveur. En effet, ce framework fonctionne sous forme de composant ce qui signifie que nous avons une seule et unique page, il n'est donc pas possible d'insérer des données dans la query string.

Les mots de passe sont hachés avant qu'ils ne soient enregistrés dans la base de données.

Et pour finir, un certificat SSL a été posé sur notre domaine dwarf.jorismartin.fr, ce qui encrypte toutes les requêtes entre le client et le serveur.

- **La facilité d'usage** : Comme énoncé dans le critère précédent, notre application visera un public large et de toute tranche d'âge, il est donc impératif que cette dernière soit utilisable de la manière la plus simple possible. Dans la même démarche, elle se doit d'être intuitive et facilement compréhensible, c'est-à-dire que les utilisateurs doivent facilement naviguer au sein de l'application et qu'ils doivent facilement s'y retrouver s'ils y retournent ultérieurement.
- **La portabilité** : Ce critère est sans doute l'un des plus importants. En effet, notre application a pour but de se retrouver disponible sur tous types de supports. Ce critère se décompose en quatre attributs :
  - La facilité d'installation : Ce qui nous concerne uniquement sur la partie d'installation sur tablette et mobile, mais cela est respecté car il s'agira ni plus ni moins que d'une simple application à télécharger. La partie sur ordinateur n'est pas concernée car il s'agit d'un simple site internet.
  - La facilité de migration et l'adaptabilité : Ce qui est aussi respecté car, comme expliqué plus tard, notre application web s'adapte facilement sur des supports plus petits grâce à une technologie permettant de transférer cette dernière en une application mobile, tout en gardant son fonctionnement qui reste le même.

- La conformité : Enfin, l'ensemble de notre application respectera l'ensemble des normes existantes la concernant, tel que les normes sur les bases des données ou encore sur le modèle de développement.

Afin de s'assurer du bon fonctionnement de notre application, notre équipe effectuera une grande série de tests. Mais comme cela n'est pas suffisant, nous soumettrons notre application à des utilisateurs "testeurs" qui nous feront un retour du fonctionnement de l'application.

## II. Qualité ergonomique

Dans la même démarche de qualité, l'ergonomie de l'application sera soumise à des utilisateurs testeurs sur les différents supports. Ces derniers nous feront part de la même manière de leur expérience sur le point ergonomique.

Pour permettre une analyse plus performante de l'interface, nous utiliserons les critères de qualité de *Bastien et Scapin*. Il nous paraît important de spécifier certains d'entre eux :

- **Le guidage** : l'utilisateur devra avoir un parcours optimal au sein de notre application, car n'importe quel usager doit pouvoir s'y repérer. Afin de rendre l'expérience utilisateur optimale, il devra avoir :
  - un feedback rapide
  - des incitations lui permettant de se déplacer aisément selon les différentes interfaces ...
- **La gestion des erreurs** : si l'utilisateur réalise un action auquel il n'est pas autorisé, les messages d'erreur retournés devront lui permettre de comprendre sans effort et rapidement son erreur.
- **La charge de travail** : le consommateur de l'application aura une charge de travail réduite ; les étapes et la lecture entre les différentes vues seront simples et limitées.
- **La compatibilité** : notre application a pour vocation d'être utilisable sur un maximum de plateforme, c'est pourquoi nous ferons au mieux pour que toutes nos implémentations soient adaptables sur un maximum de support. Voici ci-dessous un tableau résumant la compatibilité de notre application sur différents navigateurs web (*cf. Tableau 3*) :

Navigateur						
Version	1909	83.0	87.0.4280.88	74.0.3897.0	14.0	87.0.664.60
Compatibilité	✗	✓	✓	✓	(malgré quelques bugs non dérangeants)	✓

**Tableau 3** : Tableau de compatibilité des navigateurs internet

- **Homogénéité** : afin que l'utilisateur puisse s'adapter facilement selon son support d'utilisation, nous mettrons en place des interfaces quasiment identiques et cela peu importe le support d'utilisation.

### III. Critère de test

Dans un soucis de réussite, nous avons listé différents éléments qui doivent répondre à des critères précis de test :

Element	Description	Critère de test
Vue de dessin	Dessin réalisable	Le dessin est bien affiché sur l'interface dédiée
	Aide au dessin	Il est possible d'avoir une aide au dessin, incluant la visualisation de la case précédente
	Interface intuitive	La majorité des utilisateurs test (en dehors de l'équipe de développement) a réussi à dessiner sans aide
Vue de connexion	Création d'un compte	La vue permet de se créer un compte
		La création d'un compte est bien sauvegardé
	Connexion	A partir des comptes déjà créé, il est possible de se connecter
Vue des planches	Visualiser des planches	Il est possible de visualiser les dernières planches

Tableau 4 : Critères de test

Pour valider l'ensemble de ces critères, nous avons soumis notre application à un public le plus large possible, de manière à ce que ces tests soient le plus représentatif de notre public cible. Par la suite, ces mêmes personnes ont rempli un formulaire de retour les questionnant sur les divers points cités ci-dessus. (cf. Tableau 4)

## IV. Tests utilisateurs

### 1. Retour de tests

Afin de réaliser la plus grande série de tests utilisateurs, nous avons partagé massivement notre application. Nos constat sur les réseaux sociaux, nos amis et nos familles ont copieusement participé et nous les en remercions.

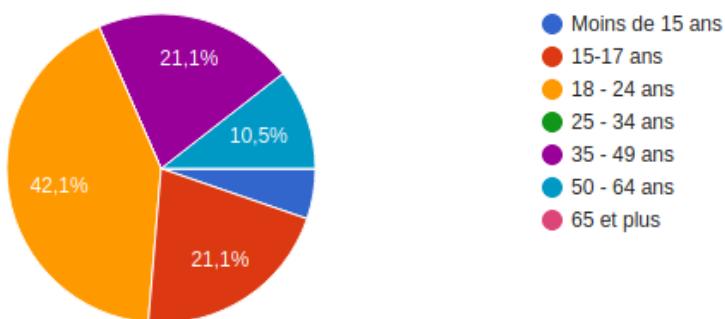
Nous avons eu la création de 29 utilisateurs. Suite à leur expérience 65% d'entre eux ont rempli le formulaire de retour. Ces nombreux retours nous ont permis et permettront de continuer à améliorer cette application.

### 2. Analyse des retours

L'un de nos objectifs était de toucher un public le plus large possible. Nous avons donc demandé aux utilisateurs ayant testé l'application de renseigner leur tranche d'âge afin de pouvoir visualiser si notre objectif allait être atteint. Voici donc ci-dessous les résultats de cette question (*cf. Figure 7*). Sur un effectif de 19 personnes, les tranches d'âge sont partagées. On constate que notre public principal reste assez jeune avec plus 60% de personnes âgés de moins de 24 ans. Néanmoins, plus de 30% de nos utilisateurs sont âgés de plus de 35 ans. Nous pouvons donc conclure que notre objectif est partiellement rempli.

Votre tranche d'âge

19 réponses

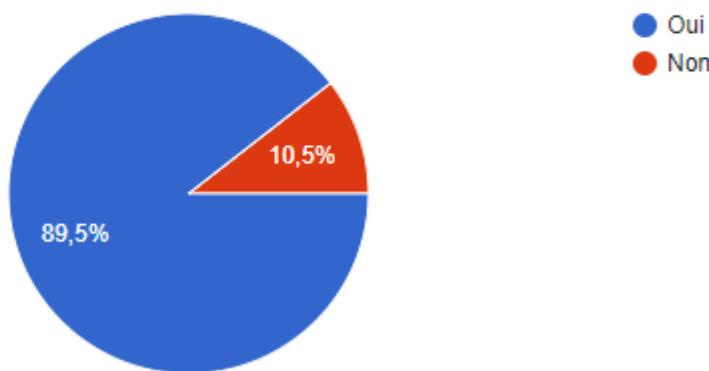


**Figure 7** : Public ayant participé aux tests.

Nous nous sommes par la suite renseigné si les utilisateurs avaient pu accéder à l'application (*cf. Figure 8*). Comme l'indique le graphique, environ 90% de nos utilisateurs ont pu accéder à l'application. Nous avons donc cherché par la suite l'utilisateur ,représentant les 10% restants, qui n'a pas pu accéder à l'application afin de pouvoir se renseigner sur la cause du non-fonctionnement de l'application chez lui. Après l'avoir contacté, ce dernier nous a expliqué qu'il s'agissait d'une erreur et qu'il a bien pu jouer. L'ensemble de nos utilisateurs ont donc pu accéder à l'application et donc on pu visualiser les planches de BD réalisées. L'objectif est donc validé.

Avez-vous pu accéder à l'application ?

19 réponses



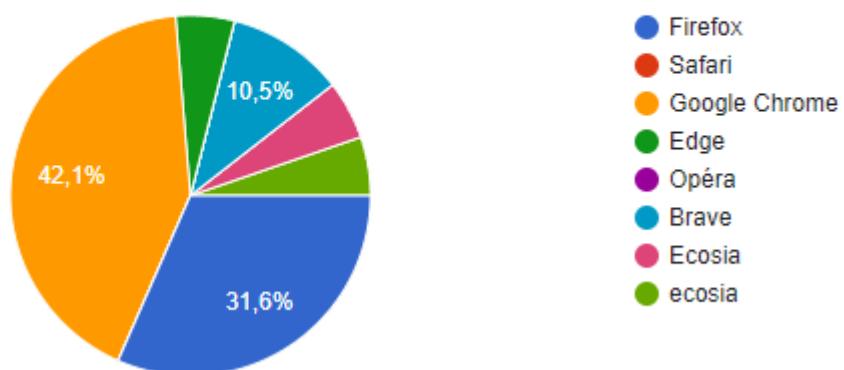
**Figure 8 :** Accès à l'application

Un second objectif était une compatibilité avec les navigateurs internet la plus large possible. Nous avions déjà réalisé une étude sur ce point (cf. Tableau 3). Ceci dit, il nous a paru logique de faire une seconde étude plus large que la précédente afin de confirmer les résultats trouvés précédemment. Étant donné que l'ensemble de nos testeurs ont pu accéder à l'application, l'ensemble des navigateurs utilisés ces derniers recensé dans le graphique ci-dessous sont donc compatibles avec notre application (cf. *Figure 9*).

(*Certains de nos utilisateurs ont confondu navigateur internet avec navigateur de recherche, le graphique possède donc quelques navigateurs de recherche. Cette présence n'a aucun impact sur notre étude*)

Quel navigateur avez-vous utilisé ?

19 réponses



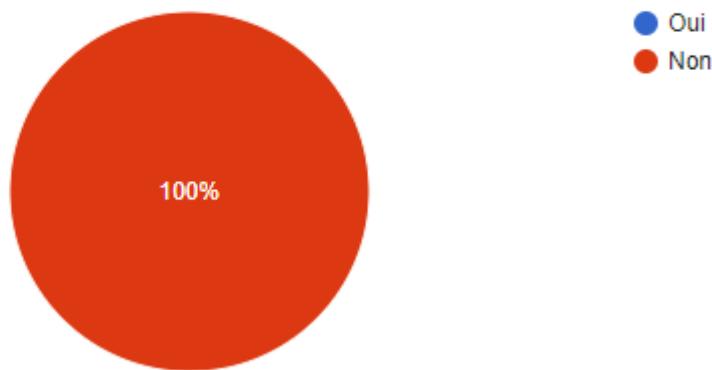
**Figure 9 :** Public ayant participé aux tests.

Une fois sur l'application l'utilisateur ne peut que visualiser les bandes dessinées fini. Pour jouer il est donc impératif de s'inscrire ou de se connecter. C'est pourquoi nous devons avoir un système d'inscription et de connexion qui fonctionne à la perfection. Il nous a paru

évident de demander aux testeurs s'ils ont rencontré des difficultés lors de cette démarche. Comme l'indique le graphique ci-dessous, le système d'inscription, connexion à correctement fonctionné pour l'ensemble de nos utilisateurs (*cf. Figure 10*).

Avez-vous eu des difficultés à vous inscrire ou vous connecter ?

19 réponses

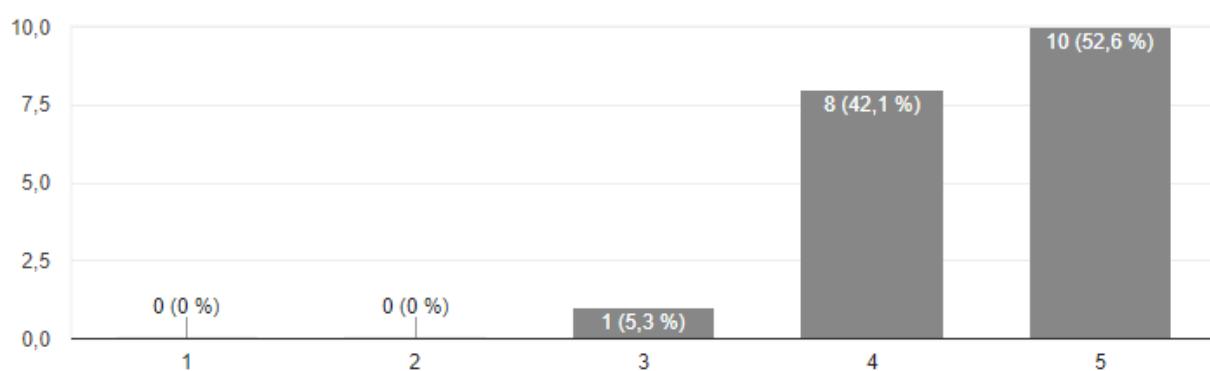


**Figure 10 :** Difficulté dans l'inscription ou la connexion

Étant donnée que notre application a pour vocation à toucher un public large, nous nous devions de rendre cette dernière la plus intuitive possible. Voici ci-dessous un graphique permettant de visualiser si nos utilisateurs ont trouvé l'application intuitive en la notant de 1 à 5 (*cf. Figure 11*). Nous constatons donc que plus de 50 % de nos utilisateurs ont trouvé notre application assez intuitive.

L'application était elle assez intuitive ?

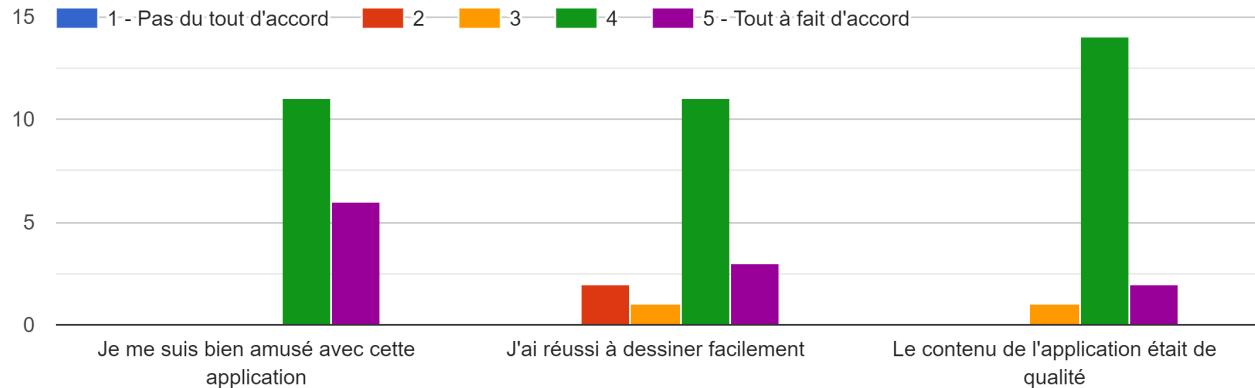
19 réponses



**Figure 11 :** Intuitivité de l'application

L'une de nos préoccupations lors de la réalisation de ce projet, fut une expérience utilisateur amusante, simple et rapide.

Grâce à ces nombreux retours, nous savons que notre application a connu un certain succès. Plus de 50% des utilisateurs s'y sont bien amusés et ont trouvé le contenu de qualité (*cf. Figure 12*). Ces retours sont donc en adéquation avec nos espérances. Cependant, certains utilisateurs ne se sont pas trouvés pleinement satisfaits de leur expérience de dessin (*cf. Figure 12*).



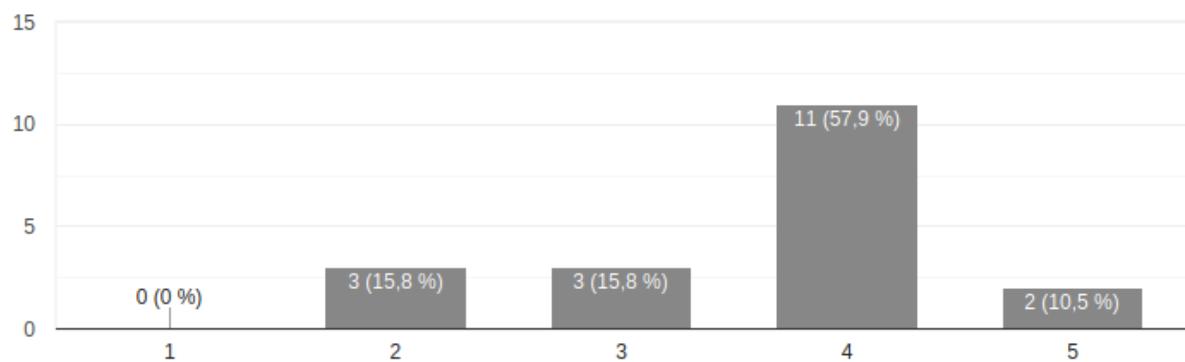
**Figure 12 :** Expérience de jeu

Nous avons également voulu savoir s'ils ont trouvé la réalisation d'un dessin rapide. Dans l'ensemble, les utilisateurs sont satisfaits de la rapidité. Cependant, pour environ 30% des utilisateurs (*cf. Figure 13*) pour qui cela n'est pas suffisant.

Pour corriger ces remarques, nous allons continuer à perfectionner les fonctionnalités de DWARF afin de gagner en simplicité et en rapidité.

#### La réalisation d'un dessin est-elle rapide ?

19 réponses



**Figure 13 :** Réalisation d'un dessin

Afin d'améliorer notre application, nous avons demandé aux testeurs à la fin du questionnaire si une fonctionnalité leur avait manqué ou s'ils avaient des idées d'option ou de nouveauté à intégrer. Suite à cette question, certaines des options demandées étaient déjà en cours d'implémentation comme "l'affichage en grand de la bande dessinée

lorsqu'on clique dessus" ou "avoir l'historique des BD que j'ai réalisées". Cependant, un grand nombre de ces retours concerne l'ajout d'outils à l'interface de dessin comme "avoir plus de couleur", "ajout de texte" ou encore "savoir le numéro de la case de BD que je vais réaliser". Certaines de ces options sont envisagées et nous allons essayer de tout mettre en place afin de répondre à ces manques.

## 4. Solutions techniques envisagées

### a. Langages et utilisation

Le noyau fonctionnel de notre application peut être implémenté avec de nombreux langages comme PHP, Java, JavaScript, C++ ou encore d'autres langages de programmation.

Ce dernier sera réalisé à l'aide du langage **PHP** notamment car il est adapté au web et maîtrisé par l'ensemble du groupe.

Nous disposerons aussi d'une base de données qui pourrait être PostgreSQL, MySQL ou encore Sqlite.

**PostgreSQL** a été retenue car la manipulation est maîtrisée par tous notamment grâce aux cours suivis au cours des deux premiers semestres réalisés à l'IUT.

Les données stockées dans cette base communiqueront depuis le back-end vers la front-end au format **JSON** (JavaScript Object Notation), qui sont facilement interprétables par le PHP comme par le JavaScript.

Enfin, notre application doit être adaptée pour tous types de supports. Nous avons différentes possibilités afin d'arriver à cette fin. Réaliser le front-end en HTML et CSS de manière responsive, et de manière encore plus simple grâce à un framework CSS comme Bootstrap, ou alors, utiliser des frameworks JavaScript comme Angular ou React qui permettent de réaliser facilement des applications web en page unique ce qui évite le chargement ou recharge de page web.

Nous avons fait le choix en premier lieu de réaliser le front-end (visuel) qui sera disponible sur ordinateur à l'aide du framework **React.js**. Ce choix est pertinent car il nous permettra facilement d'implémenter le front-end sur tablette et mobile à l'aide de **React Native** qui est un autre framework permettant d'utiliser React.js avec les fonctionnalités native sur smartphone.

Nous avons résumé notre analyse des technologies susceptibles dans le tableau suivant (*cf. Tableau 5*) :

Front-end	Les plus	Les moins
React + React Native	<i>Possibilité d'utiliser du code très ressemblant à la version web pour l'implémentation</i>	<i>Coût d'apprentissage</i>

	<i>d'une application native</i>  <i>Une connaissance du groupe maîtrise bien la technologie, donc possibilité de lui poser des questions</i>	
Angular	-	<i>Coût d'apprentissage</i>
Vue.js	-	<i>Coût d'apprentissage</i>
HTML / CSS	<i>Connu et maîtrisé par tous</i>	<i>Difficile à gérer quand l'application grandit</i>
Back-end	Les plus	Les moins
PHP	<i>Connu et maîtrisé par tous</i>	<i>Technologie vieillissante</i>
Node.js	<i>Intégration avec React simplifiée</i>	<i>Coût d'apprentissage</i>
Next.js	<i>Intégration avec React encore plus simplifiée</i>	<i>Coût d'apprentissage</i>
PostgreSQL	<i>Connu et maîtrisé par tous</i>  <i>Bien intégrable dans PHP</i>	-
MySQL	<i>Le plus populaire, donc bien documenté</i>	<i>Coût d'apprentissage</i>
MariaDB	-	<i>Coût d'apprentissage</i>
MongoDB	<i>Intégrable facilement dans un environnement NodeJS</i>	<i>Coût d'apprentissage</i>  <i>Apprentissage du NoSQL</i>  <i>Pas intégrable nativement dans PHP</i>

**Tableau 5 :** Analyse des technologies

## b. Caractéristiques techniques

### I. Caractéristiques de notre application

Notre application sera conçue sous le modèle de développement appelé “Modèle Vue Contrôleur”, plus connu sous l’acronyme MVC.

Le modèle correspond à l’architecture logique de notre application. Ce dernier communiquera de manière directe avec le contrôleur qui sera chargé d’effectuer toutes les opérations de traitement de l’application. Enfin le contrôleur enverra les informations à communiquer à l’utilisateur à la vue qui sera la partie graphique et visible de notre

application (*Annexes 4, 5, 6, 7, 8*). De ce fait, notre application fonctionne de manière asynchrone.

## II. Caractéristiques du serveur

Le serveur hébergeant notre application est un VPS (Virtual Private Server) loué chez l'hébergeur français OVH. Le système d'exploitation est Debian 10 (Buster), choisi pour sa stabilité et ses performances. Nous avons adopté Apache comme serveur web car c'est une solution populaire très documentée.

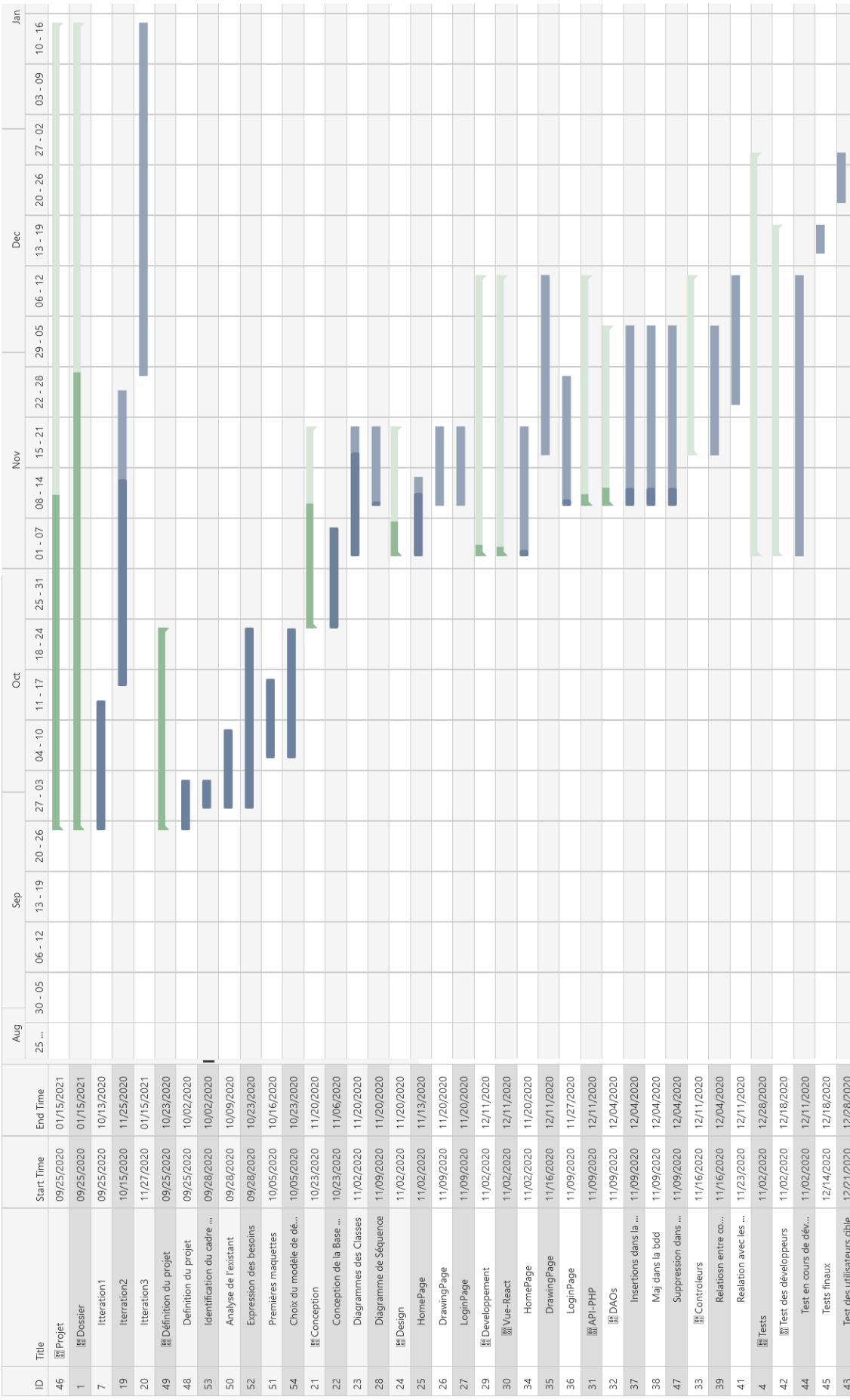
# 5. Organisation du travail et découpage du projet

## a. Mode de fonctionnement

Notre équipe fonctionnera en utilisant les principes de gestion semi-agile, c'est-à-dire une simplification d'une simplification des itérations (réduction du nombre de revues avec la maîtrise d'ouvrage).

Notre équipe se réunira tous les vendredis afin d'effectuer une mise au point de l'avancement des tâches définies la semaine précédente. Par la suite, nous fixerons les tâches à réaliser pour le vendredi suivant.

Depuis la première itération, nous réalisons la conception générale de notre application. Suite à cela, nous commencerons la réalisation d'une première maquette fonctionnelle. Une fois la maquette réalisée, nous implémenterons, fonctionnalités par fonctionnalités le reste de l'application.



**Figure 14 : Road-map prévisionnelle pour la première et deuxième itération**

## b. Outils

Afin de mener à bien notre projet, nous définissons plusieurs outils dont nous allons avoir besoin :

Les diagrammes de conception de notre application ont été conçus avec **Visual Paradigm**. La première version des maquettes non fonctionnelles a été réalisée avec le site **Draw.io**, et la seconde version avec **Figma**, un outil collaboratif de création d'interfaces.

Pour pouvoir collaborer tous ensemble sur ce projet, nous utiliserons des outils tels que **git** et le serveur **gricad-gitlab** (*cf. Annexe 1*) mis à disposition par l'université, ainsi que **Google Docs** (*cf. Annexe 2*) qui nous permet de modifier ce dossier tous en même temps.

Afin de correctement gérer notre avancement au cours du projet, nous utiliserons un outil de méthodologie de gestion de projet kanban appelé **Trello** (*cf. Annexe 3*).

Nous utilisons également des conversations textuelles et vocales **Discord** pour communiquer efficacement en dehors des heures de cours prévues pour notre projet (et également lors des heures prévues à cause de la Covid-19).

Dans le but de débugger notre back-end sans modifier la front-end, nous utilisons le site **apitester.com** qui permet d'envoyer des requêtes HTTP GET ou POST à notre back-end et de visualiser les données JSON renvoyées. L'outil simule donc les requêtes de notre front-end à notre back-end.

Pour faciliter les déploiements des modifications faites par chacun d'entre nous sur le git, nous avons mis en place **un script bash** sur le serveur permettant de *git pull*, puis compiler le client React et le déployer sur le serveur apache, et enfin déployer les scripts PHP sur ce même serveur.

## 6. L'application

### a. DWARF

Notre application DWARF est donc disponible sur le web à l'adresse <https://dwarf.jorismartin.fr>

Cette dernière est en constante évolution et nous continuons à implémenter des options, nouveauté à ce jour.

## b. Screenshot de l'application

Voici des screenshots de l'application actuellement disponibles.

*Il peut y avoir quelques différences entre les screenshots et l'application sur le web due à l'intégration continue.*

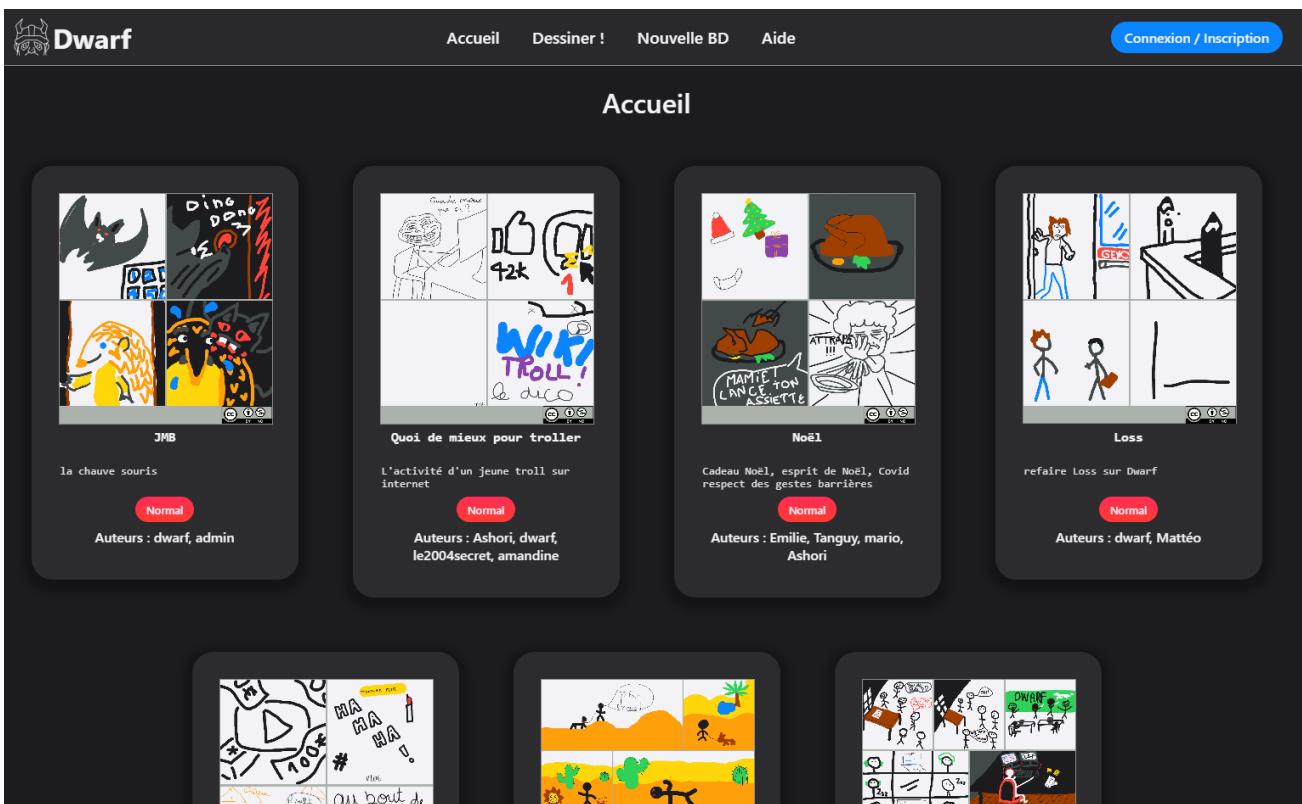


Figure 15 : Page principale de DWARF

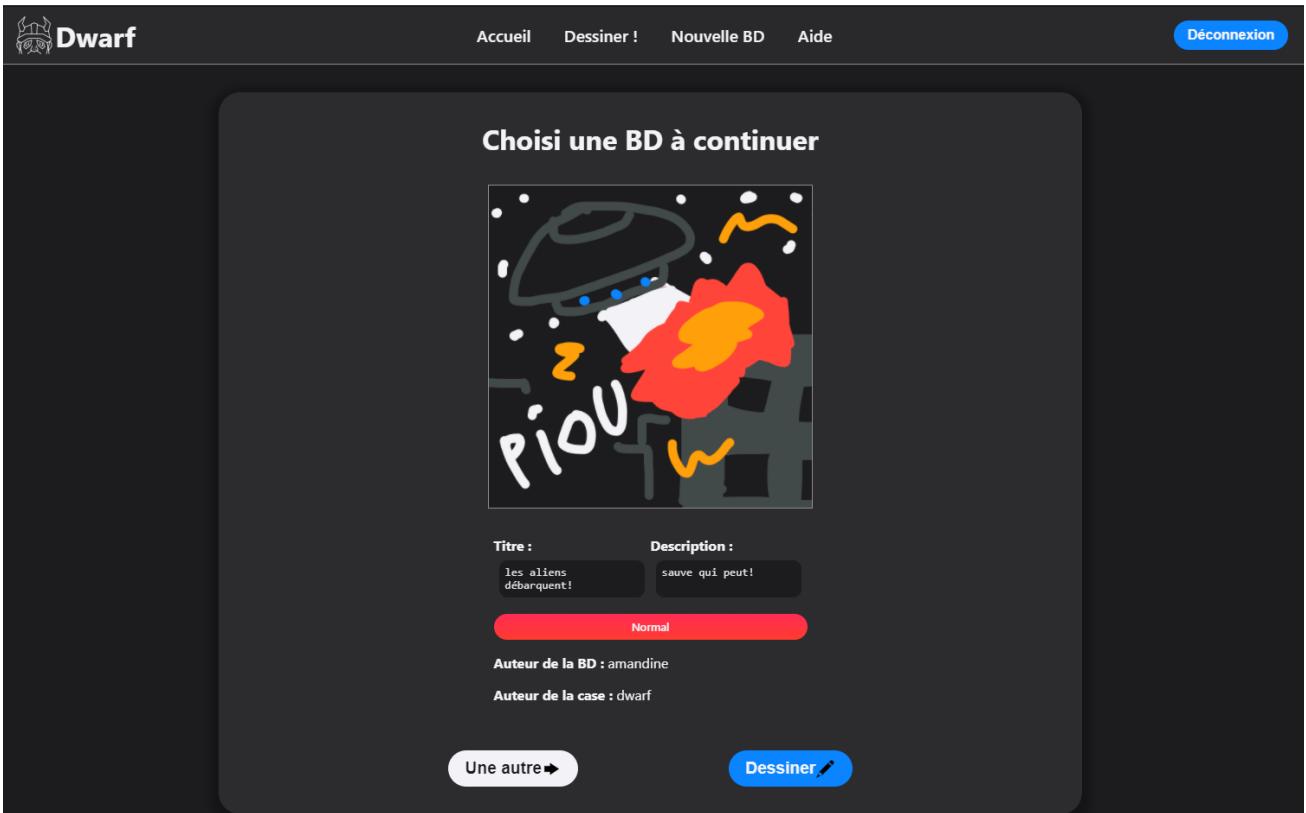


Figure 16 : Page de sélection des BD à continuer

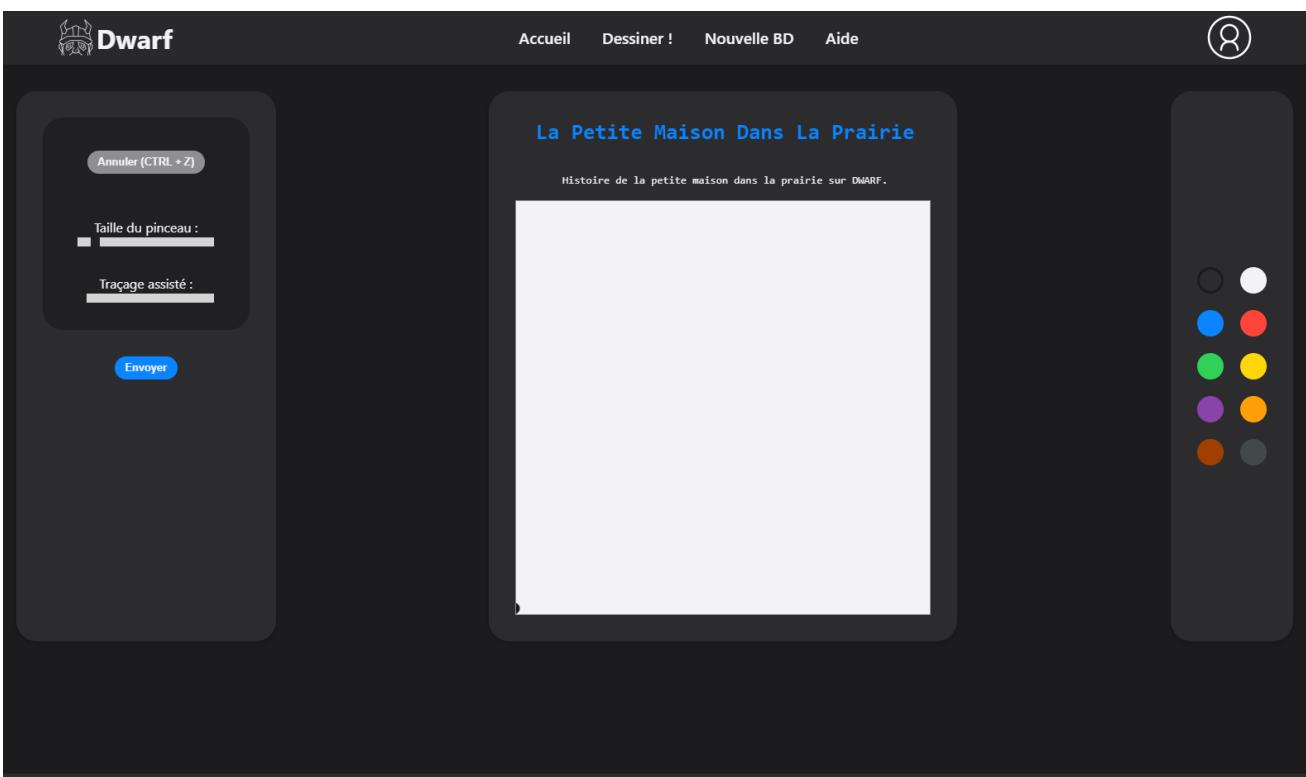


Figure 17 : Page de dessin

## 7. Annexes

The screenshot shows a GitLab project page for a repository named 'rendus'. The sidebar on the left contains links for Project overview, Details, Activity, Releases, Repository, Issues (0), Merge Requests (0), CI / CD, Operations, Packages & Registries, Analytics, Wiki, and Snippets. The main content area displays project details: 157 Commits, 1 Branch, 1 Tag, 6 MB Files, 6 MB Storage. It also shows that the project is forked from an inaccessible project and lists integrations with chamilo and discord. Below this is a commit history showing a merge branch 'master' by Tanguy Signoret. A table lists recent commits with columns for Name, Last commit, and Last update. The commits include '.gitlab/issue\_templates', 'api', 'bdd', and 'docs'.

**Annexe 1 :** Gricad qui est une plateforme de l'UGA qui utilise l'outil git qui permet de stocker et de garder une version à jour de notre projet auprès de tous les membres de notre équipe.

The screenshot shows a list of files shared with the user. The files are:

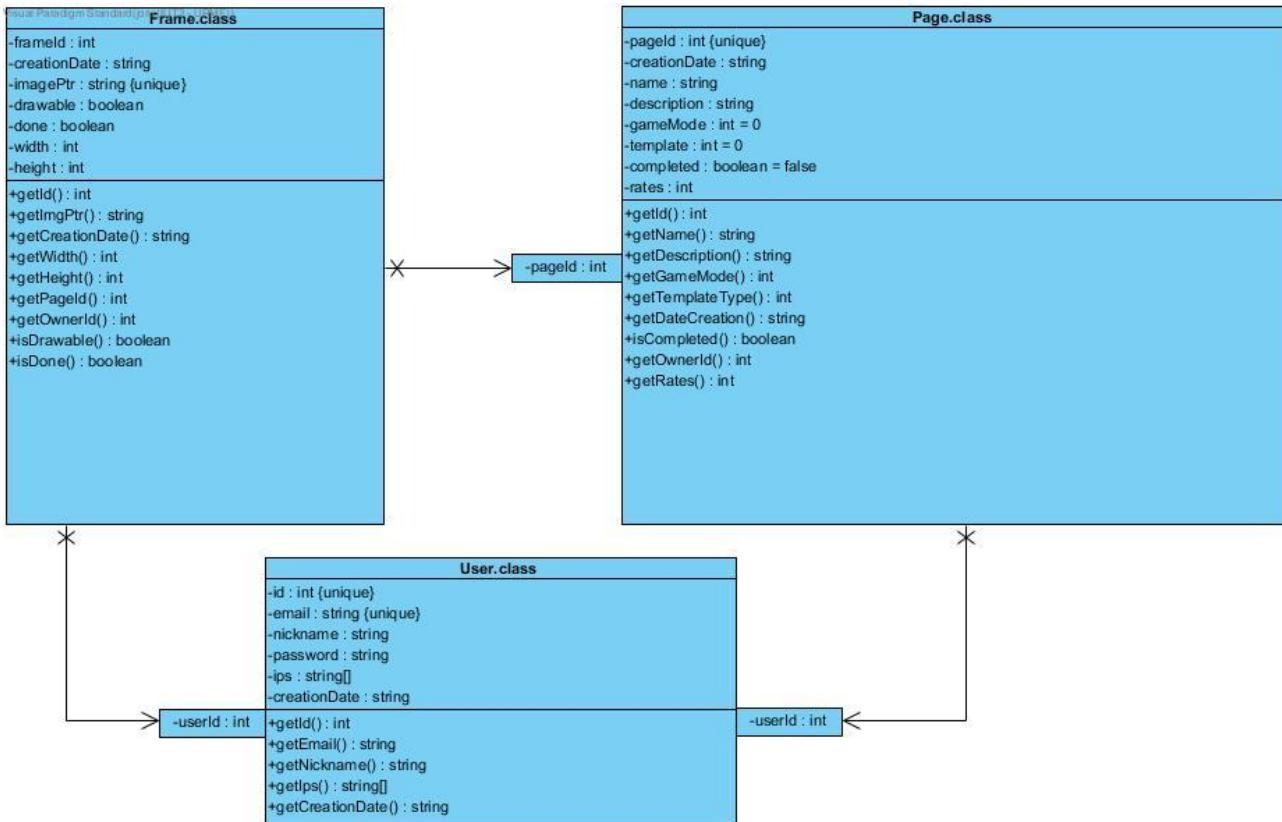
- BRAINSTORMING
- CR\_Retour\_Iteration1
- Presentation1
- Presentation2
- TEAM DWARF

Each file entry includes the name, owner (Tanguy Signoret), last modified date (23 oct. 2020 or 16 oct. 2020), and file size (indicated as "–").

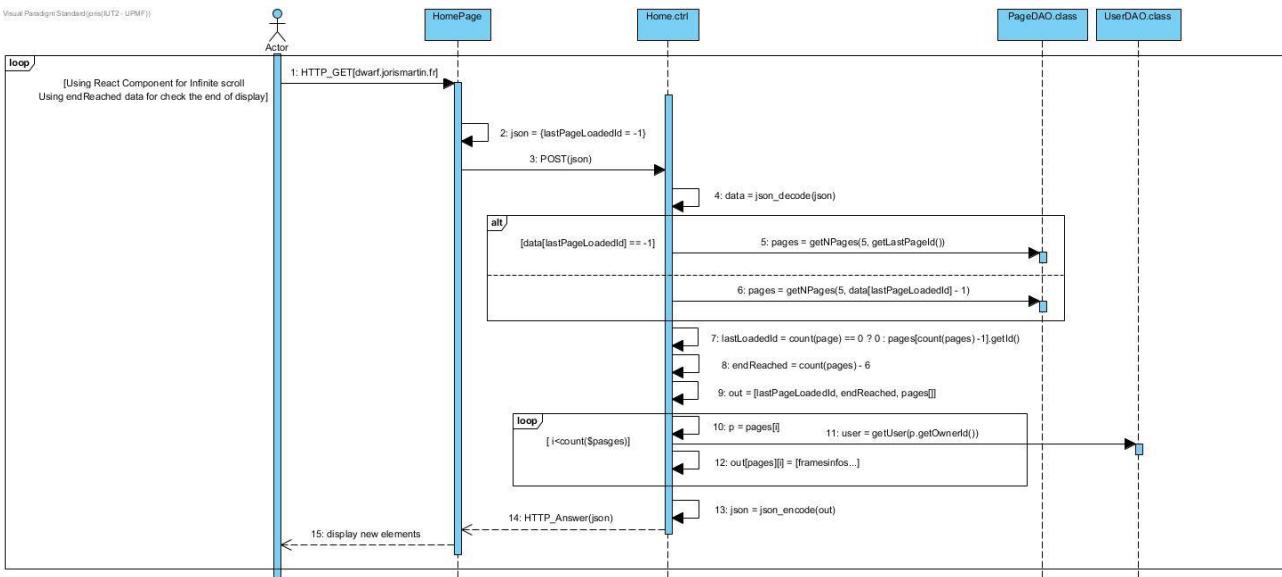
**Annexe 2 :** Google Drive qui permet l'édition de fichier, notamment de ce dossier, de manière simultané entre plusieurs utilisateurs. Permet aussi de stocker des fichiers.

The screenshot shows a Trello board with five columns: Options, To Do, Doing, Done, and Testing. The Options column contains notes about tablet/mobile adaptation, native functionality, personal space, moderation system, and adding more cards. The To Do column has cards for 'To Do' (1 item), 'Inscription', 'Vue planche en cours, à compléter', 'Système de notation', 'implémentation des tempalte', and 'implémentation des modes de jeu'. The Doing column has cards for 'Doing' (1 item), 'Maquette -> figma.com', 'Design interface', 'Conception - Diagramme de séquence', 'Implémentation design (CSS)', and 'Continuer le dossier'. The Done column has cards for 'Done' (2 items), 'Autentification', 'bien configurer apache (.htaccess) et les routes pour supporter le refresh et les accès avec url directe', 'Affichage temporaire des planches en cours sur la page d'accueil', 'Implémentation création d'une planche (formulaire)', and 'Implémentation espace de dessin'. The Testing column has cards for 'Testing' (1 item), 'BDD - Construction', and 'gantt - itération 1 et itteration 3'. There is also a '+ Add another card' button in each column.

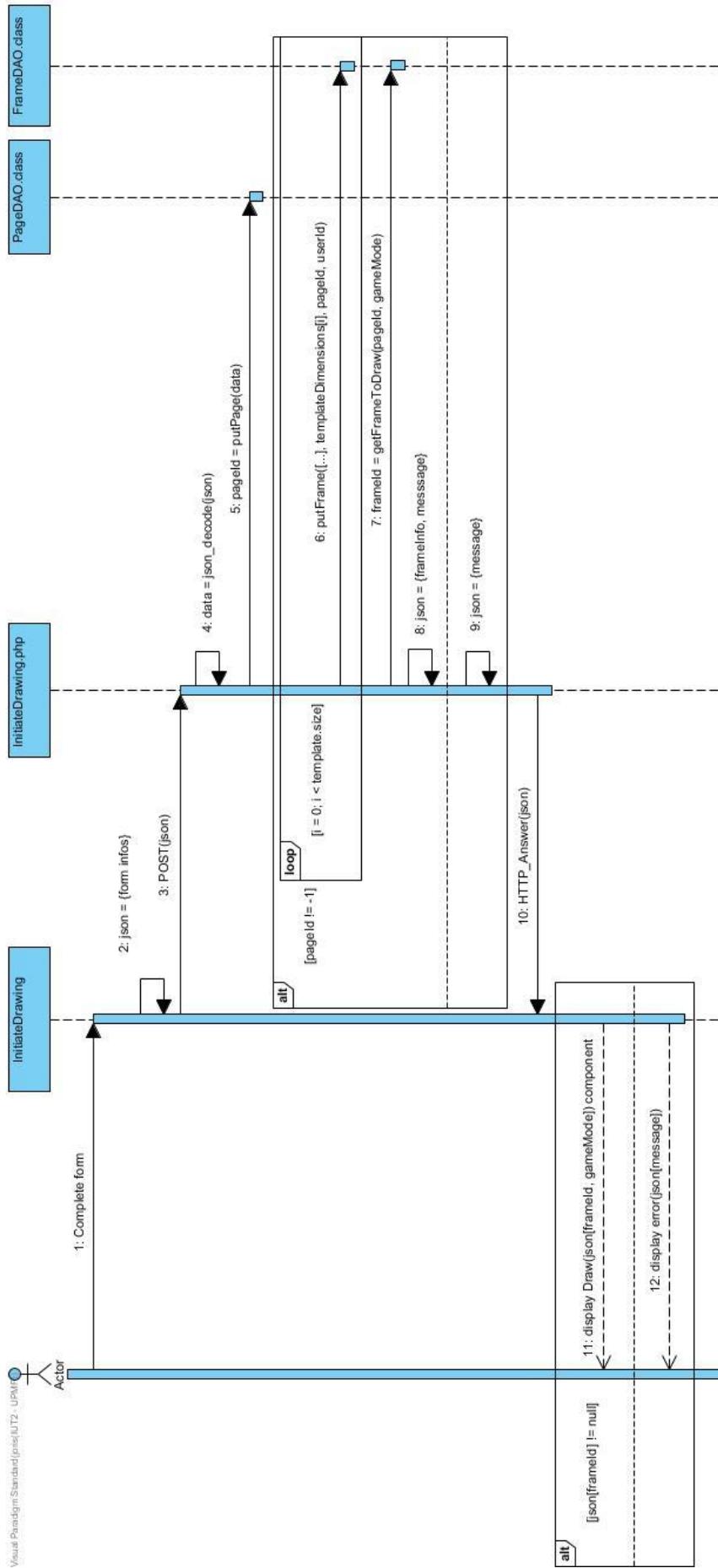
**Annexe 3 :** Trello qui est un outil permettant de constater l'avancement des tâche



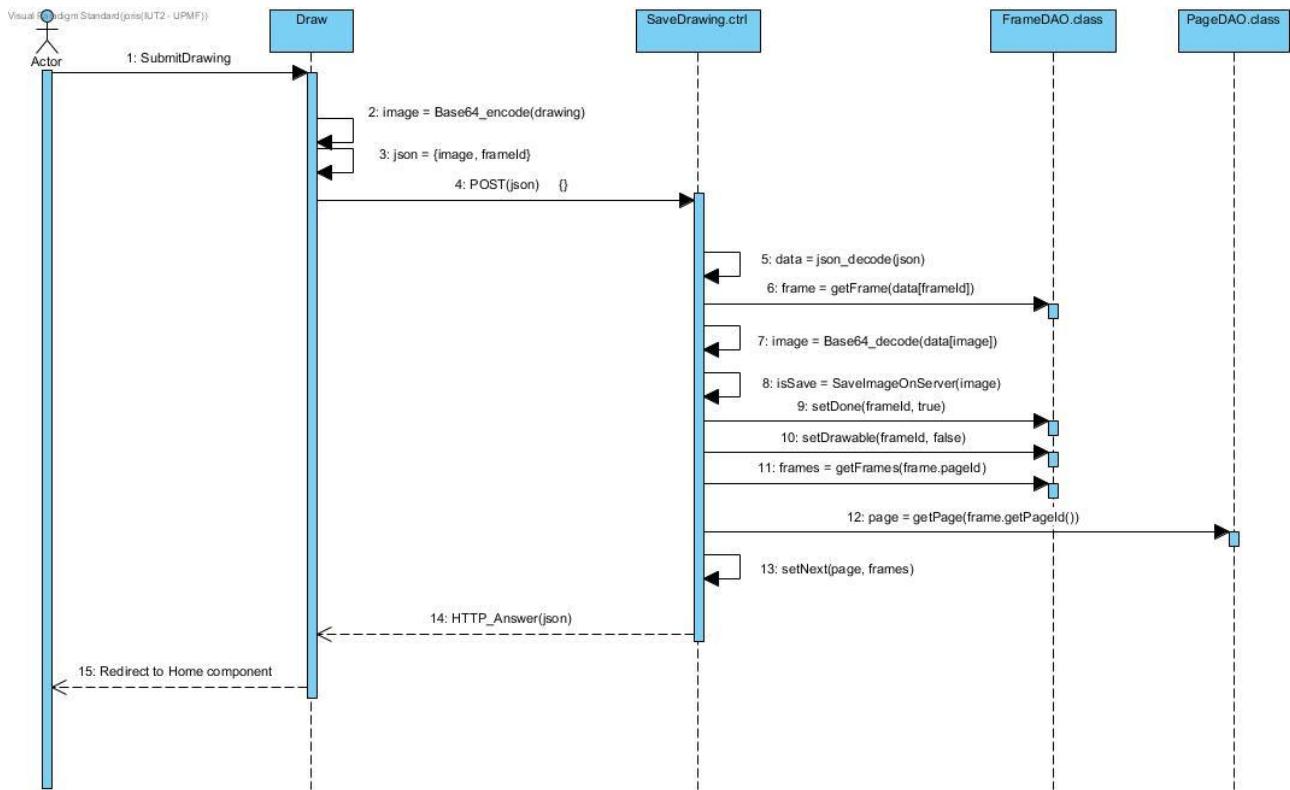
**Annexe 4 : Conception ⇒ Diagramme de classes du modèle**



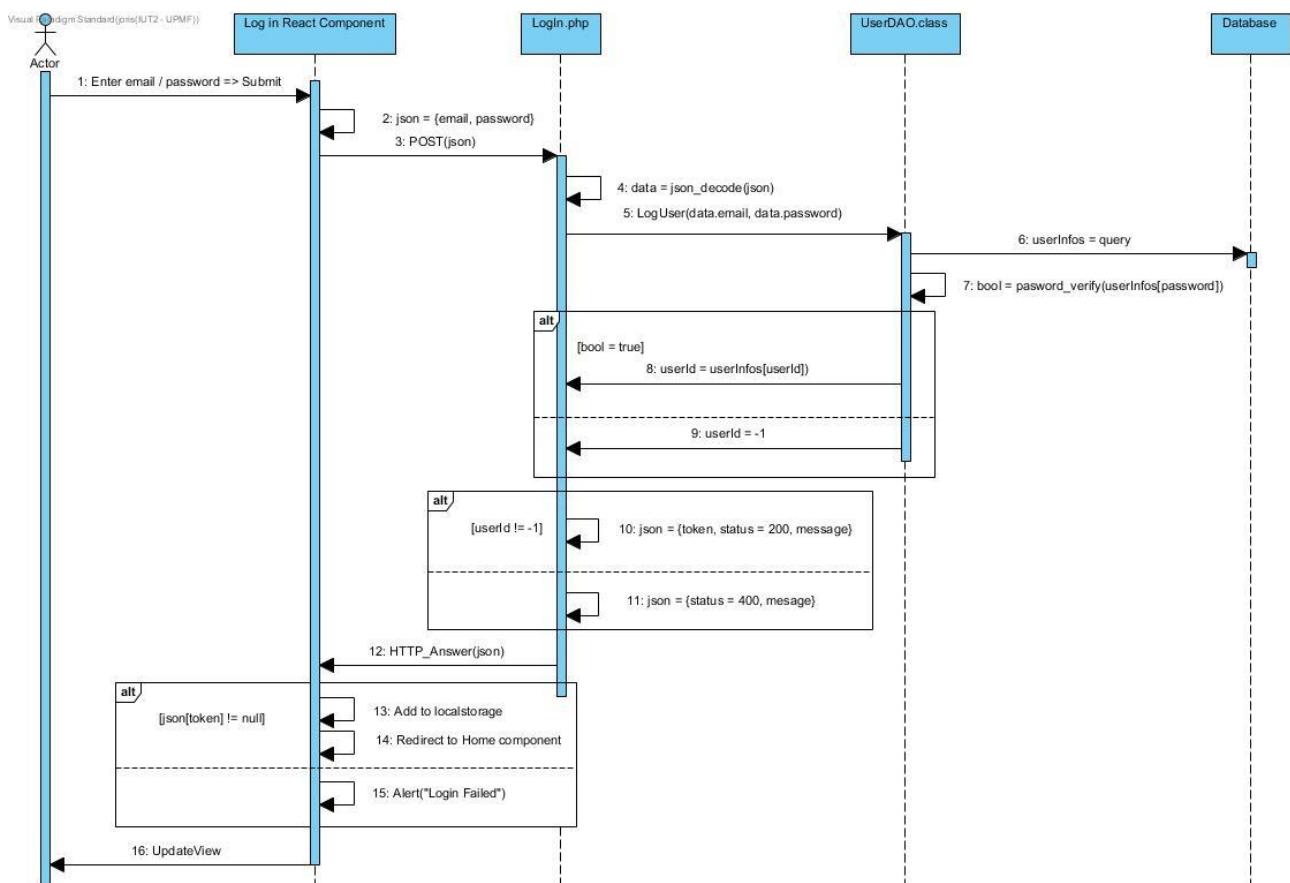
**Annexe 5 : Conception ⇒ Diagramme de Séquence de l'affichage des BD sur la page d'accueil. (Diagramme simplifié due à l'utilisation d'un module React “Infinite Scroll” qui rend la modélisation difficile)**



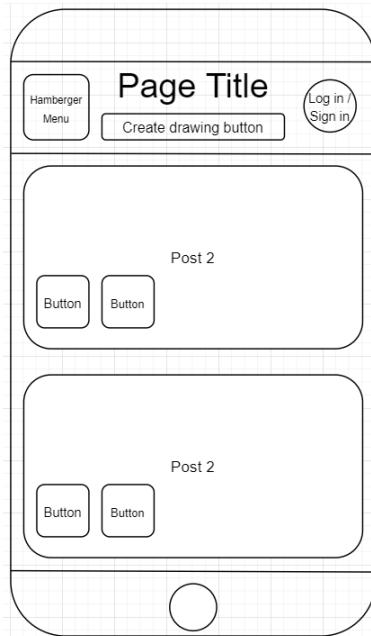
## Annexe 6 : Conception ⇒ Diagramme de Séquence de la soumission du formulaire de création d'une planche



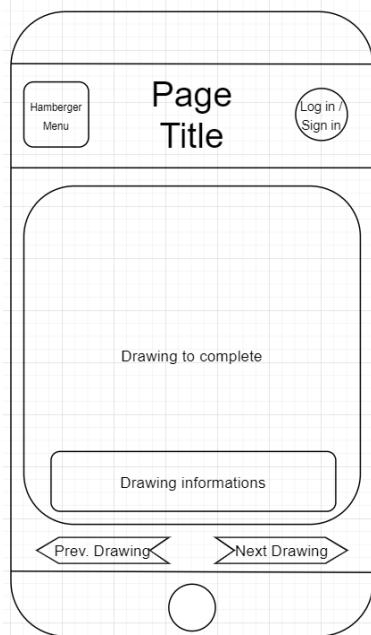
## Annexe 7 : Conception ⇒ Diagramme de Séquence de la soumission d'un dessin. (Diagramme temporaire)



**Annexe 8 : Conception ⇒ Diagramme de Séquence de la connexion**



**Annexe 10 : Première version maquette smartphone - Page principale**

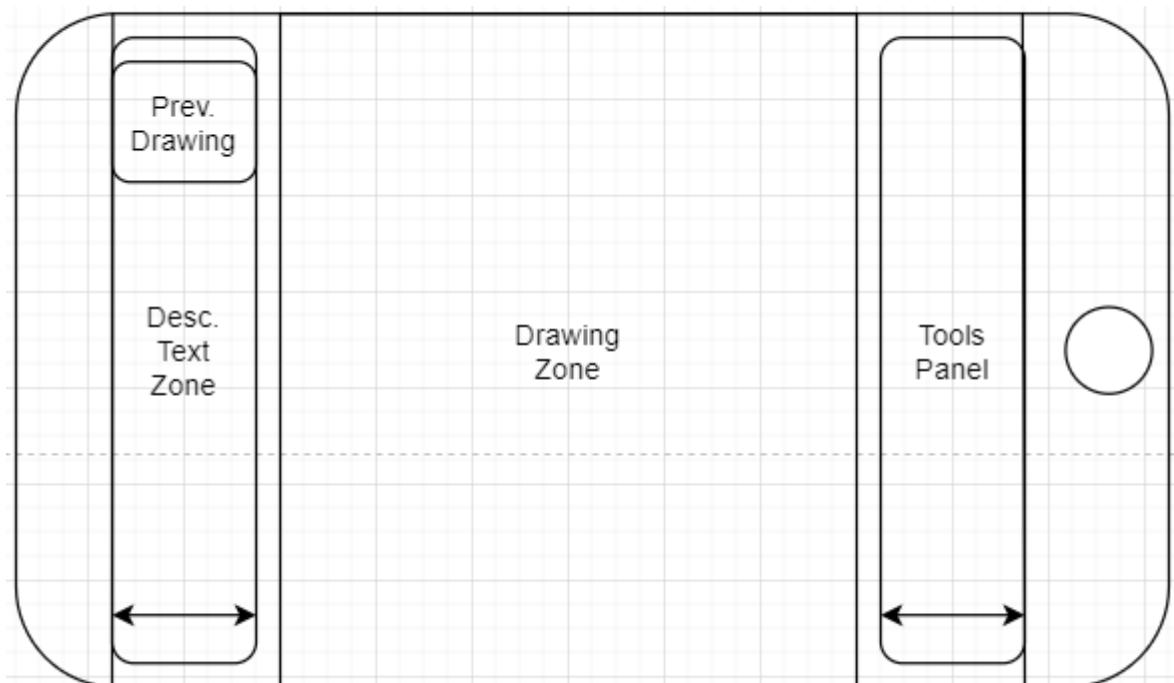


**Annexe 11 : Première version maquette smartphone - Navigation entre les planches en cours**

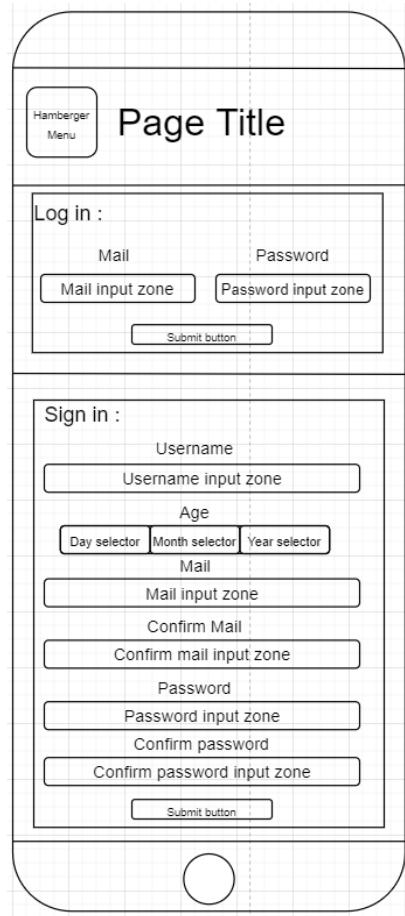
The image shows a smartphone wireframe with rounded corners. Inside, there is a form for creating a planche. At the top left is a "Hamburger Menu" button, and at the top right is a "Submit button". The main area contains the following fields:

- Title / Theme :** A "Text input" field.
- Game mode :** Three radio buttons labeled "Normal", "Reverse", and "Intermediate".
- Description / constraint :** A "Text input" field.
- Template :** Four buttons labeled "Template 1", "Template 2", "Template 3", and "Template n".
- Game type:** Two buttons labeled "Public" and "Private".

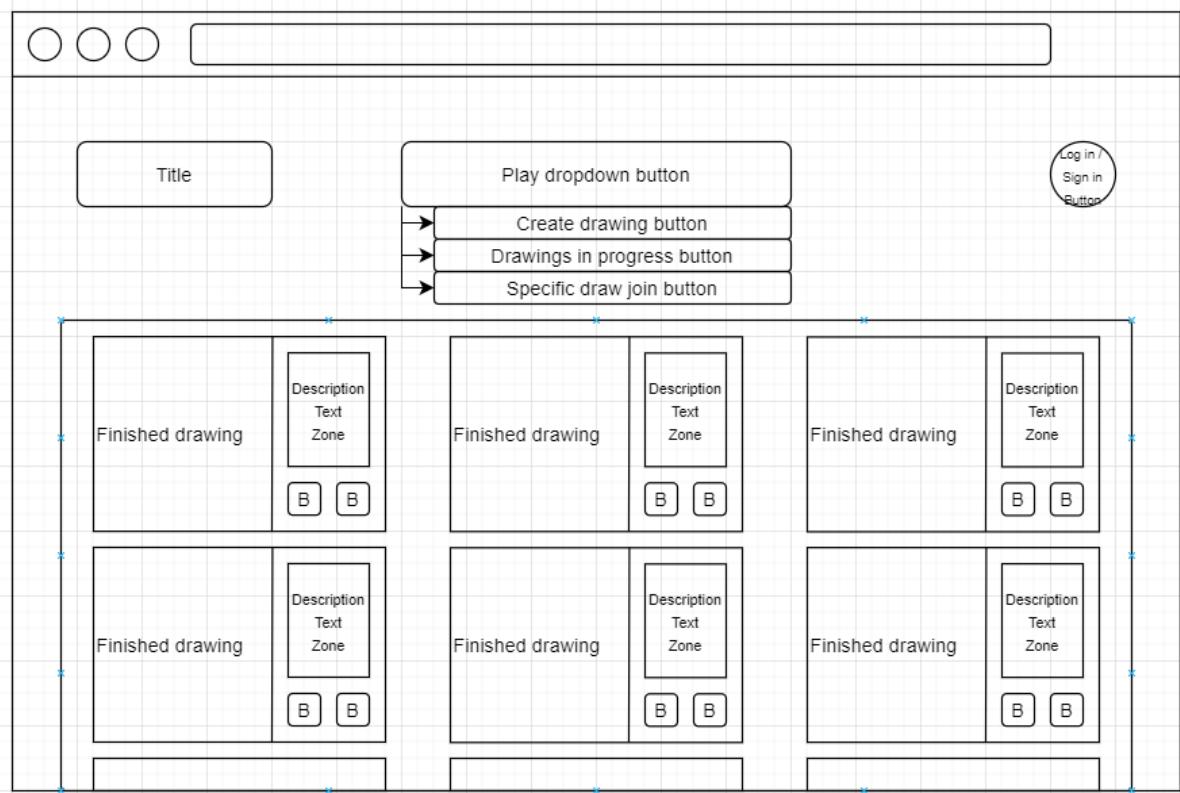
**Annexe 12 :** Première version maquette smartphone - Formulaire de création de planche



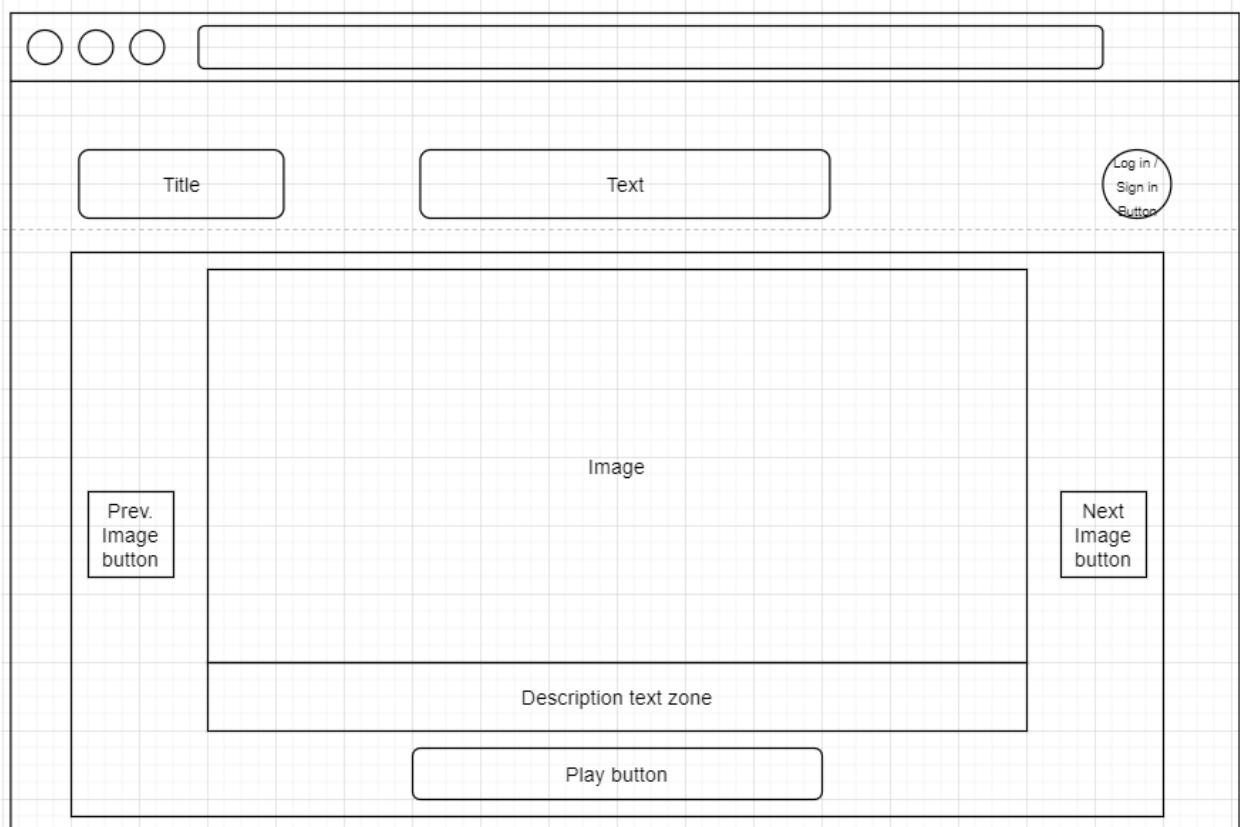
**Annexe 13 :** Première version maquette smartphone - Interface de dessin



**Annexe 14 :** Première version maquette smartphone - Interface de connexion /  
Inscription



**Annexe 15 :** Première version maquette PC / Tablette - Page principale

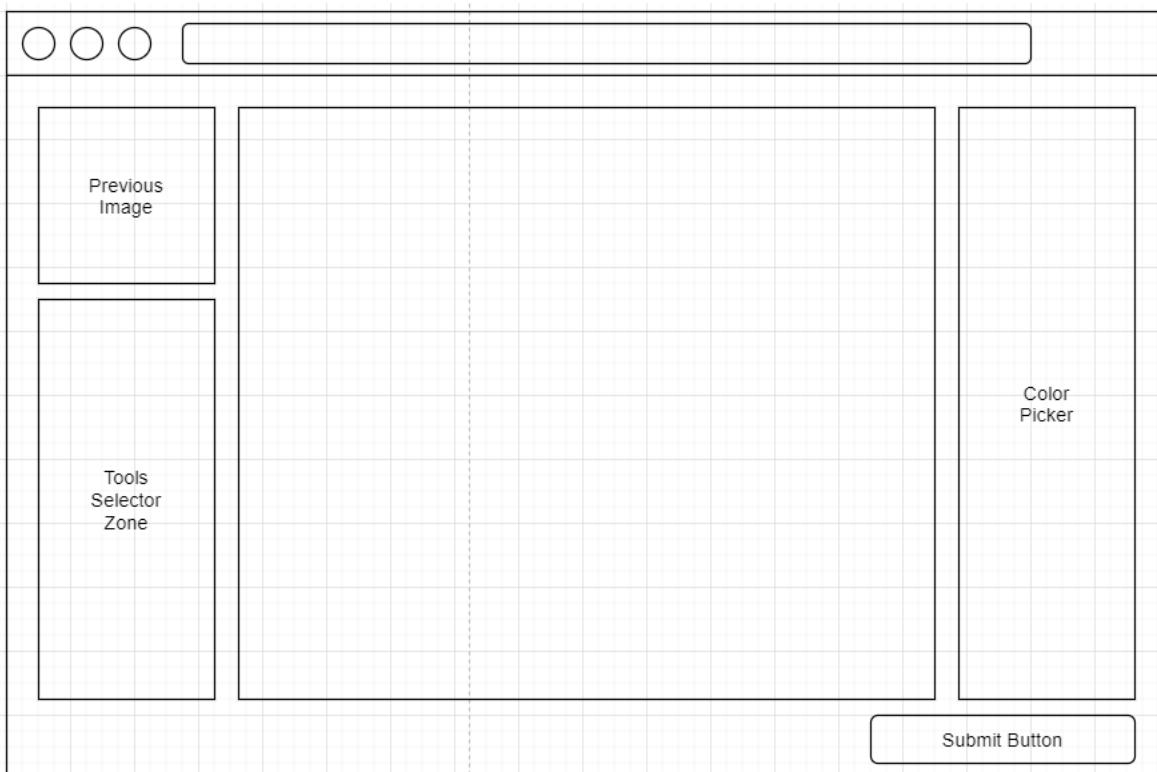


**Annexe 16 :** Première version maquette PC / Tablette - Navigation entre les planches en cours

Wireframe of a PC/Tablet screen showing a form for creating a board:

- Title**: Text input field.
- Text**: Text input field.
- Title / Theme**: Text input field.
- Game mode :**
  - Normal
  - Reverse
  - Intermediate
- Description / constraint :** Text input field.
- Template :** A row of seven "Template" buttons.
- Game type:**
  - Public
  - Private
- Submit button**: A large rectangular button at the bottom right.

**Annexe 17 :** Première version maquette PC / Tablette - Formulaire de création de planche

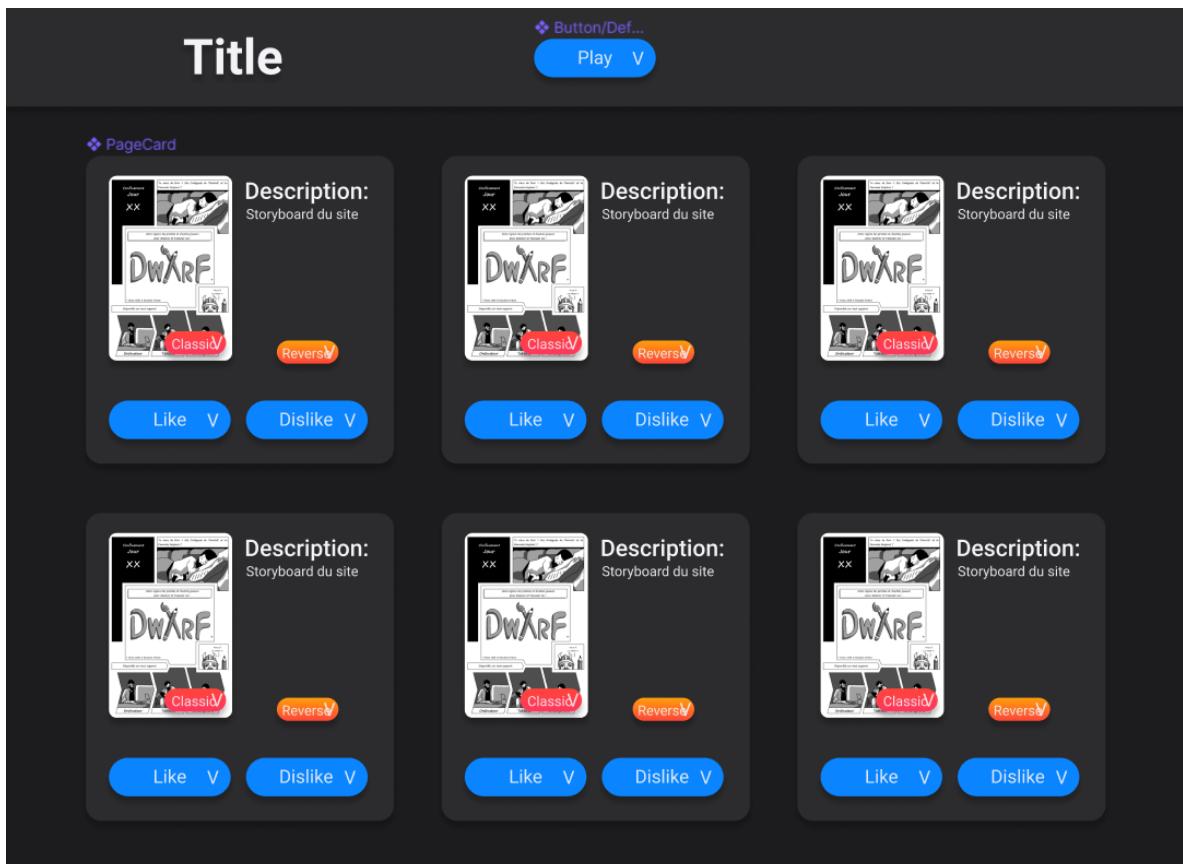


**Annexe 18 :** Première version maquette PC / Tablette - Interface de dessin

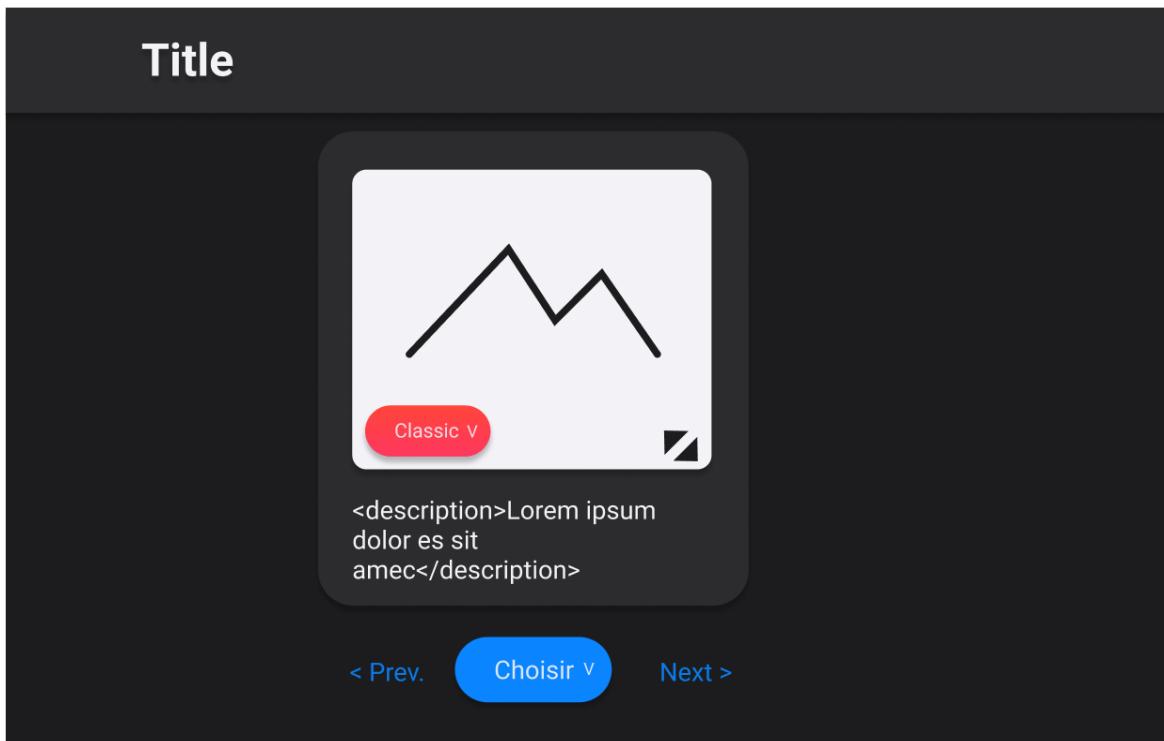
The wireframe illustrates a mobile application interface. At the top, there are three circular icons and a large rectangular input field labeled "Text". Below this, the screen is divided into two main sections by a vertical dashed line.

- Left Section (Log in):**
  - Section title: "Log in :"
  - Form fields:
    - "Mail" with "Mail input zone"
    - "Password" with "Password input zone"
  - A "Submit button" at the bottom.
- Right Section (Sign in):**
  - Section title: "Sign in :"
  - Form fields:
    - "Username" with "Username input zone"
    - "Age" with "Day", "Month", and "Year" inputs
  - A "Submit button" at the bottom.

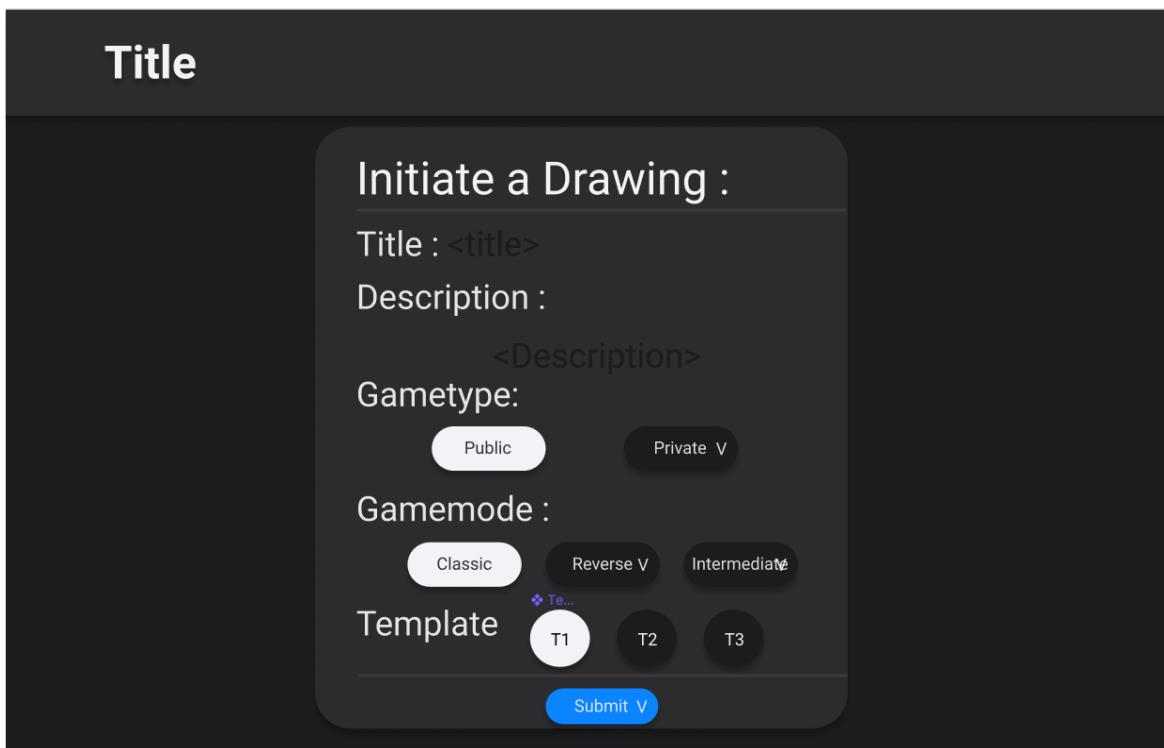
**Annexe 19 :** Première version maquette PC / Tablette - Interface de connexion / Inscription



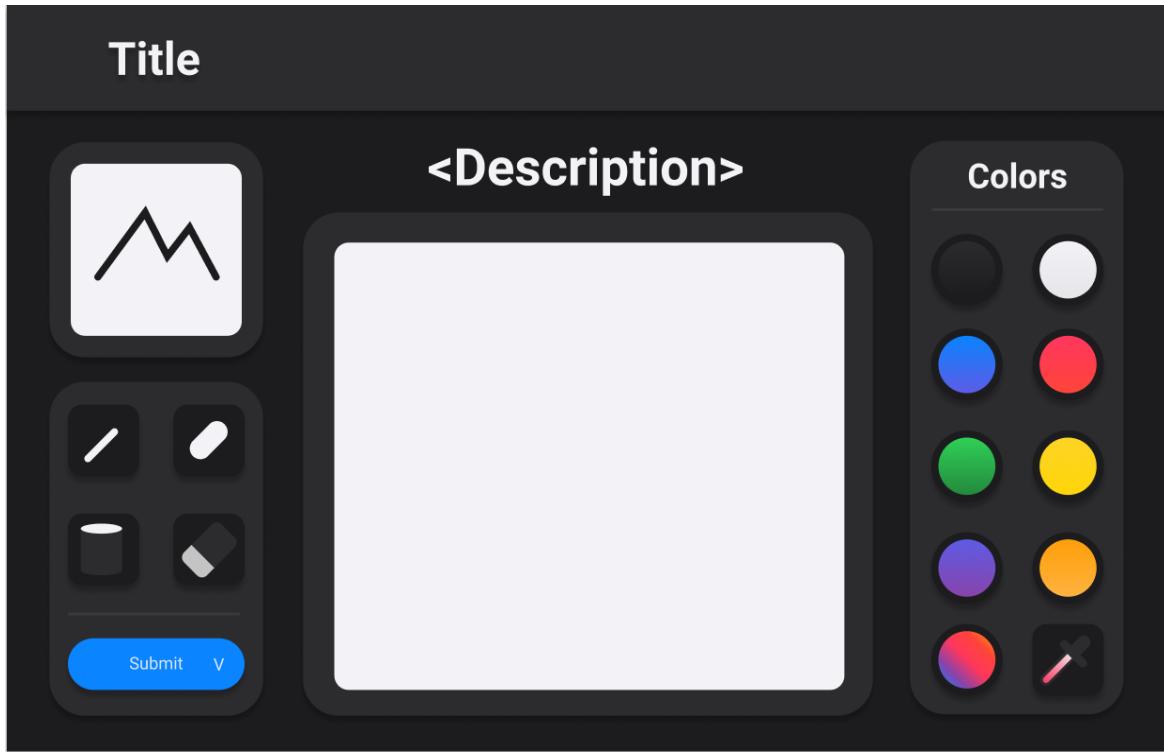
**Annexe 20 :** Seconde version maquette PC / Tablette - Page principale



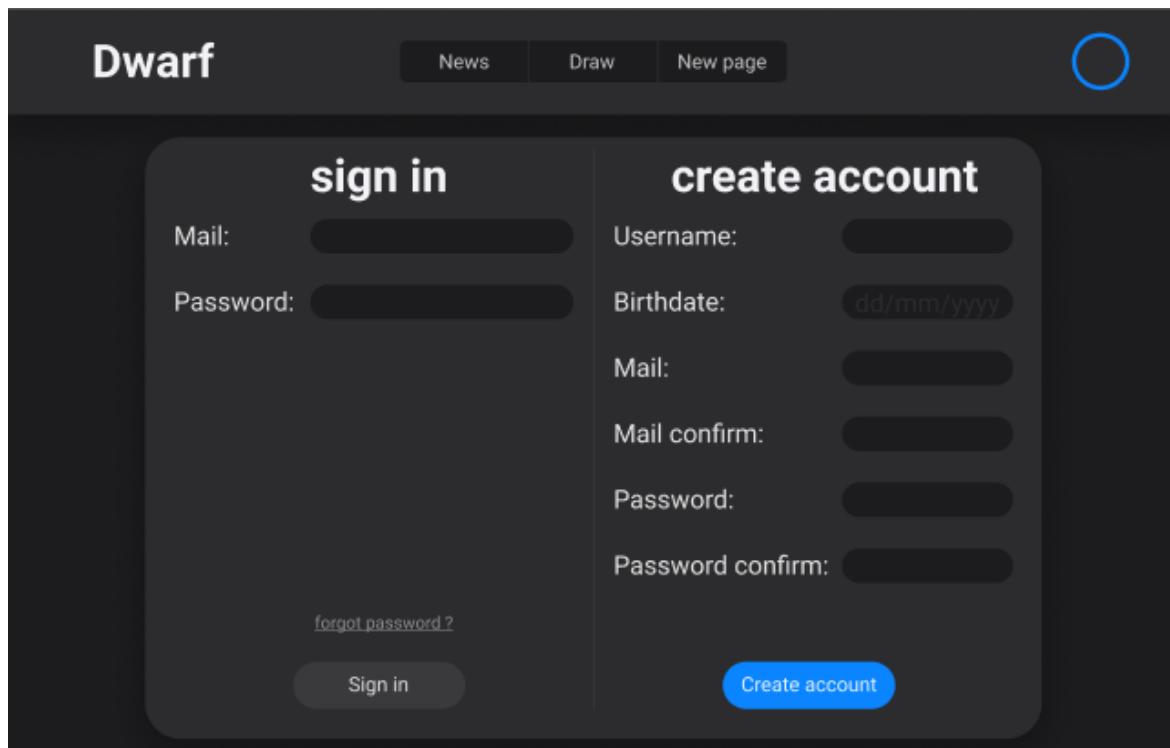
**Annexe 21 :** Seconde version maquette PC / Tablette - Navigation entre les planches en cours



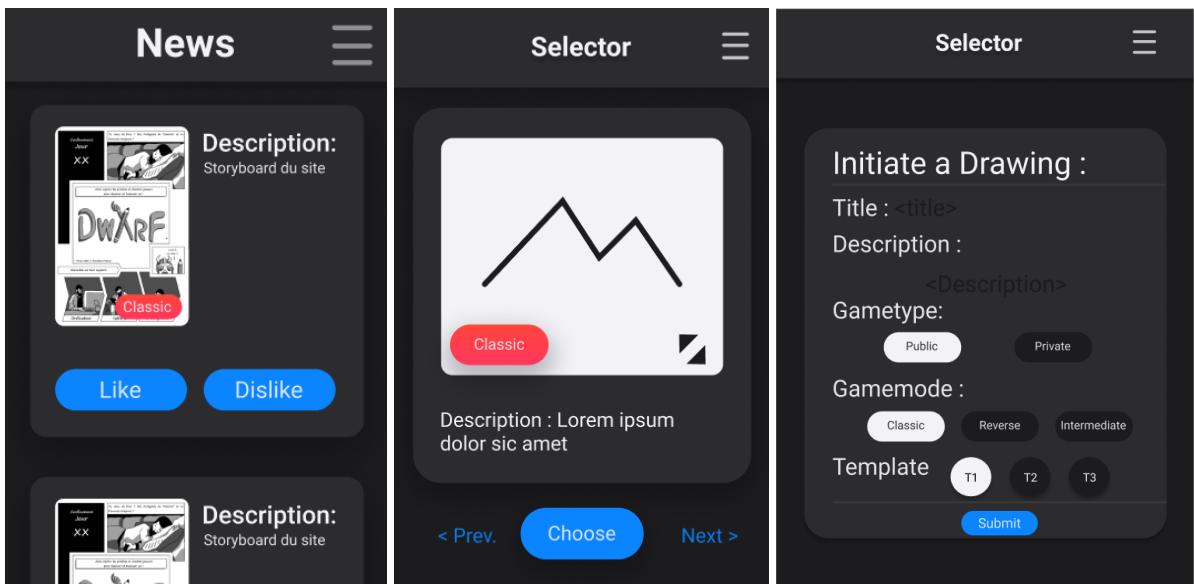
**Annexe 22 :** Seconde version maquette PC / Tablette - Formulaire de création de planche



**Annexe 23 :** Seconde version maquette PC / Tablette - Interface de dessin



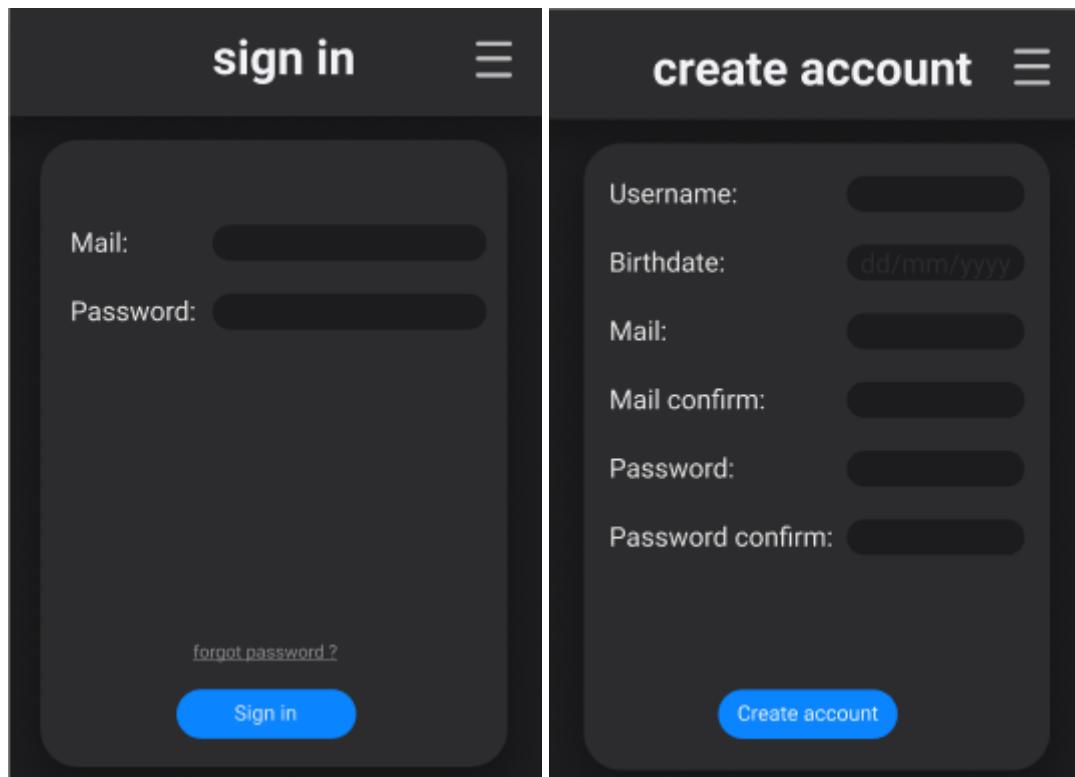
**Annexe 24 :** Seconde version maquette PC / Tablette - Interface de connexion / Inscription



**Annexe 25 :** Seconde version maquette smartphone - Page principale

**Annexe 26 (centre) :** Seconde version maquette smartphone - Navigation entre les planches en cours

**Annexe 27 (droite) :** Seconde version maquette smartphone - Formulaire de création de planche

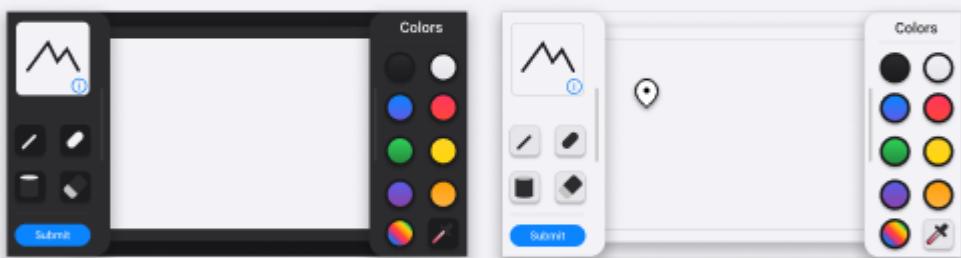


**Annexe 28 :** Seconde version maquette smartphone - Interface de connexion / Inscription

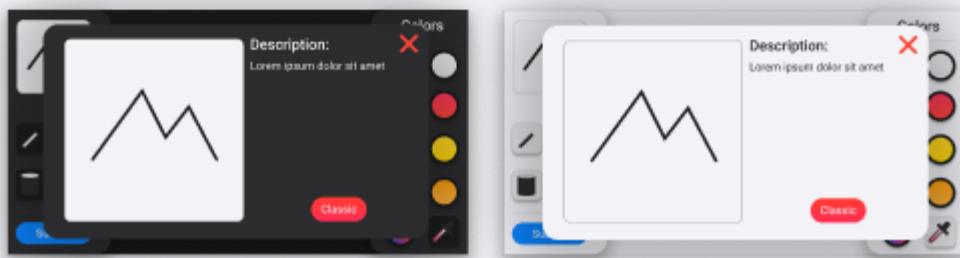
### Zone de dessin de base :



### Menu adjacent déplié :



### Menu du bouton (info) de l'image :



**Annexe 29 :** Seconde version maquette smartphone - Interface de dessin