

Présentation Projet Titre RNCP37625

niveau 6

Concepteur Développeur d'Application Web et Mobile



ParentPalAI

Tom Fontaine – Batch 183 Bordeaux – 9 janvier 2025-  La Capsule

Table des matières

1. Conception et prototypage

- 1.1 description du projet
- 1.2 Storyboard
- 1.3 Test utilisateurs
- 1.4 Wireframes
- 1.5 UI/UX kit
- 1.6 Maquettes finales
- 1.7 User stories

2. Préparation du projet

- 2.1 Compétence appliquées pour la réalisation de l'application
- 2.2 Répartition des tâches
- 2.3 Schématisation de la base de données
- 2.4 API du backend

3. Pilotage du développement

- 3.1 Lien vers le code source du projet
- 3.2 Architecture globale du projet
- 3.3 Description de l'interface graphique
- 3.4 Composants et librairies utilisés
- 3.5 Contre-rendu des sprints

4. Mise en production

- 4.1 Mise en production du backend et du frontend
- 4.2 Schéma de l'environnement de déploiement
- 4.3 Système d'authentification

5. Conclusion

1 Conception et prototypage

1.1 Description du projet

ParentPal AI est une application mobile destinée à accompagner les jeunes parents et futurs parents dans leur parcours parental. L'application agit comme un assistant virtuel intelligent, offrant un accompagnement personnalisé, des conseils adaptés et une communauté d'entraide.

Problématiques identifiées :

1. **Besoin d'accompagnement personnalisé** - Les jeunes parents ont besoin de réponses rapides, fiables et adaptées à leur situation personnelle.
2. **Surcharge d'informations contradictoires** - Internet regorge d'informations sur la parentalité, souvent contradictoires ou inadaptées. ParentPal AI filtre et personnalise les conseils.
3. **Isolement parental** - De nombreux parents se sentent isolés. Notre forum communautaire permet d'échanger conseils et soutien.

Les solutions apportées par ParentPal IA sont :

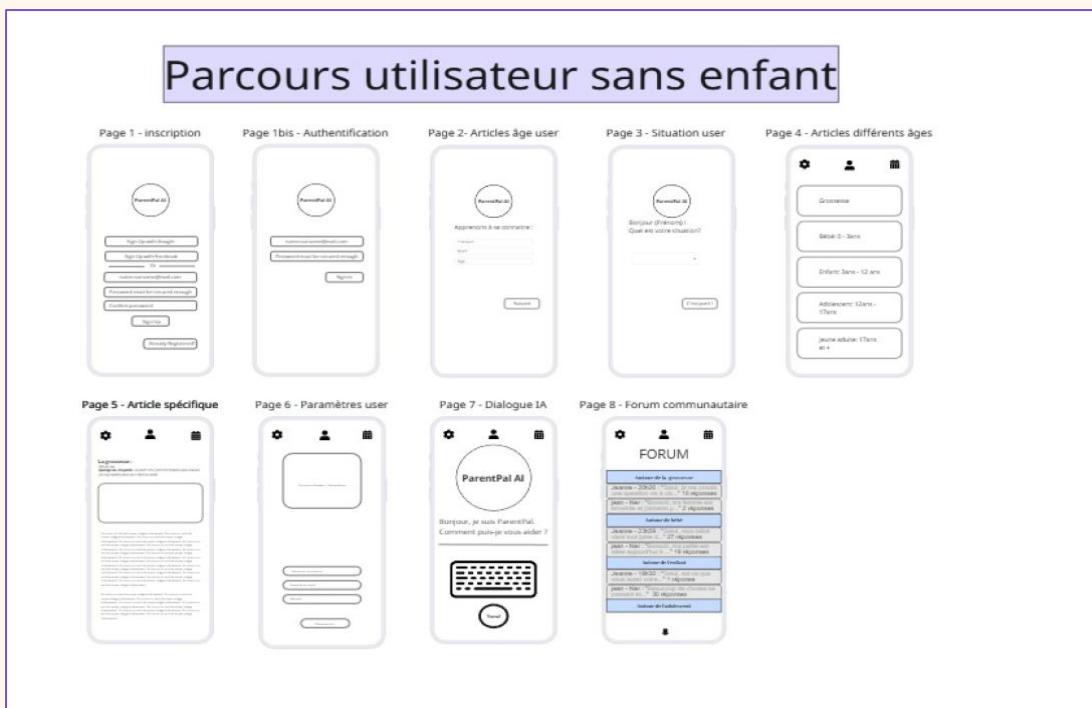
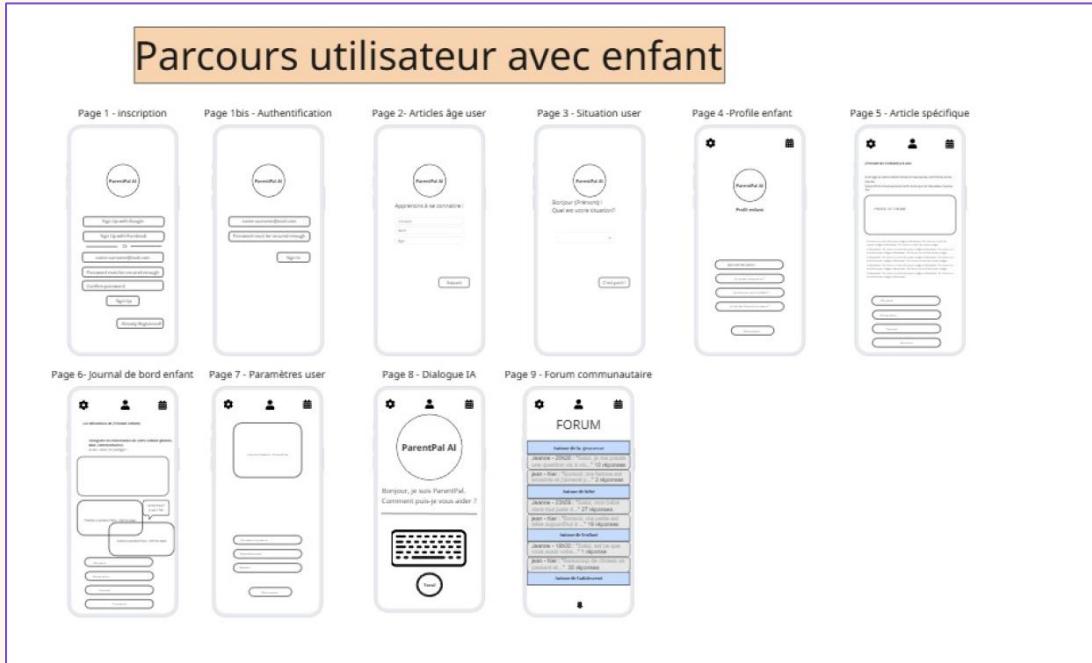
- Assistant IA disponible 24/7
- Articles personnalisés selon l'âge de l'enfant
- Forum communautaire bienveillant
- Suivi personnalisé du développement de chaque enfant

Contraintes du projet :

- **10 jours** de développement (8-19 décembre 2024)
- Équipe de **3 développeurs**
- Technologies: React Native (Expo) + Node.js/Express

1.2. Storyboard

Nous avons créé deux parcours utilisateurs distincts : "Avec enfant" pour les parents actuels et "Sans enfant" pour les futurs parents. Le but étant de voir s'il était viable d'avoir une application dont les fonctionnalités sont différentes en fonction de la situation.



Panel utilisateur :

Nous avons choisi deux personnes étant élèves aussi à La Capsule mais travaillant sur un autre projet. Un utilisateur devait avoir la vision de parent et l'autre d'une personne sans enfant.

Testeur	Age	Situation	Parcours
Thomas Gillaizeau	34 ans	1 enfant En couple	Avec enfant
Grégory Adolphe	22 ans	Etudiant Sans enfant	Sans enfant

1.3 Tests utilisateurs

Comme mentionné dans la partie storyboard (cf. 1.2. Storyboard), nous avons soumis chacun de nos schémas à deux testeurs de La Capsule qui ont eu pour rôle de tester les parcours et émettre des avis / recommandations. À la suite de ces échanges, chaque sous-groupe a créé un tableau récapitulatif des remarques /recommandations reçues.

Chaque tableau a ensuite fait l'objet d'une discussion lors de notre réunion de fin de sprint. Cette procédure nous a permis de choisir le scénario et le storyboard retenus.

D'un point de vue méthodologique, il est à noter que toutes les remarques faites par les utilisateurs ont été évaluées et prises en compte lors de notre développement.

Les deux tests utilisateurs sont présentés à la page suivante.

Il a été finalement retenu de faire une application générale laissant accès à toutes les données de l'application peu importe notre situation, une page

ou nous pouvons ajouter nos enfants sera la différence pour les personnes ayant des enfants.

1.3.1 Test Utilisateur avec enfant

Nom du test UTILISATION PARENTPAL IA - AVEC ENFANT				
Grandes étapes:				
Connexion à l'app Touche(s): Ecran de login / Création de compte	Découverte du profil parent Page d'accueil, onboarding	Création du profil enfant formulaire enfant	Interaction avec l'IA suggestions IA / assistant virtuel	Navigation contenus et fonctionnalités articles, forum, paramètres
Action du testeur: choisi de se connecter via address mail.	ajout de ses informations, de la situation (avec enfant), a défini le nombre d'enfants.	ajout des informations concernant son enfant (informations générales)	clic sur l'assistant virtuel pour poser des questions.	navigue sur les différents articles et forums concernant l'âge de son enfant, accès paramètres profile
PROFIL DU TESTEUR homme, 34 ans Bordeaux 1 enfant en couple	GOALS CONCLUSION DU TEST: fluidité sur la compréhension, un peu dubitatif sur l'utilisation de l'IA EXPERIENCE VECUE: bon ressenti, utile pour pour des questions de tous les jours.	FONCTIONNALITÉS PRÉSENTES : moyen de connexion / Authentification Numéro de la page du storyboard: page 1	FONCTIONNALITÉS PRÉSENTES : paramétrier son profil utilisateur Page 2 : Articles Age user Page 3 : Situation user	FONCTIONNALITÉS PRÉSENTES : paramétrier le profil enfant, article le concernant. Page 4 : Profil enfant Page 5 : Article spécifique Page 6 : Journal de bord enfant
RESSENCE: A l'aise pour la connexion, (depuis ce système sur d'autres applications mobile)	se sent guider mais n'a pas compris qu'à la page 3, en fonction de notre situation d'autres menus déroulants apparaissent, (ex : nombre d'enfant, place de l'enfant dans la fratrie, etc..)	facilité de l'interface, se demande ce qu'il se passait lors du clic sur un élément comme "quel est son poids?"	compréhension de son utilisation mais à peur de son utilisation si elle est générale sur l'application, moyennement à l'aise avec l'IA.	discussions avec la communauté, consultation d'articles en rapport ou non avec l'âge de son enfant, modification de son profile
Critical positive aspects of experience	designe sobre et efficace, ne déstabilise pas par rapport aux autres applications dans la façon d'accéder à notre compte.	bien qu'il y est que certaines infos et que ça ne demande pas trop d'informations personnelles excepté les essentielles.	outil simple et efficace pour des questions générales ou être renseigné sur l'application mobile si nous ne trouvons pas le contenu désiré.	bonne idée de pouvoir voir les commentaires des autres utilisateurs et voir que nous ne sommes pas la seule personne à se poser une question.
Critical points for improvement	le menu déroulant est esthétique mais cache les différents choix avant qu'on les sélectionnent, peut-être le menu avec croix et tout visible pourrait plus convenir pour savoir où nous allons?	ne pas donner trop d'informations sur l'enfant pour éviter que l'IA ne récupère tout.	ajouter une limite sur l'accès de l'IA pour les risques de sécurité des données surtout sur le médical	ajout d'une messagerie pour pouvoir dialoguer avec les autres membres pourrait être intéressant.
ANALYSE DU TEST (POSITIVE ET NEGATIVE)				

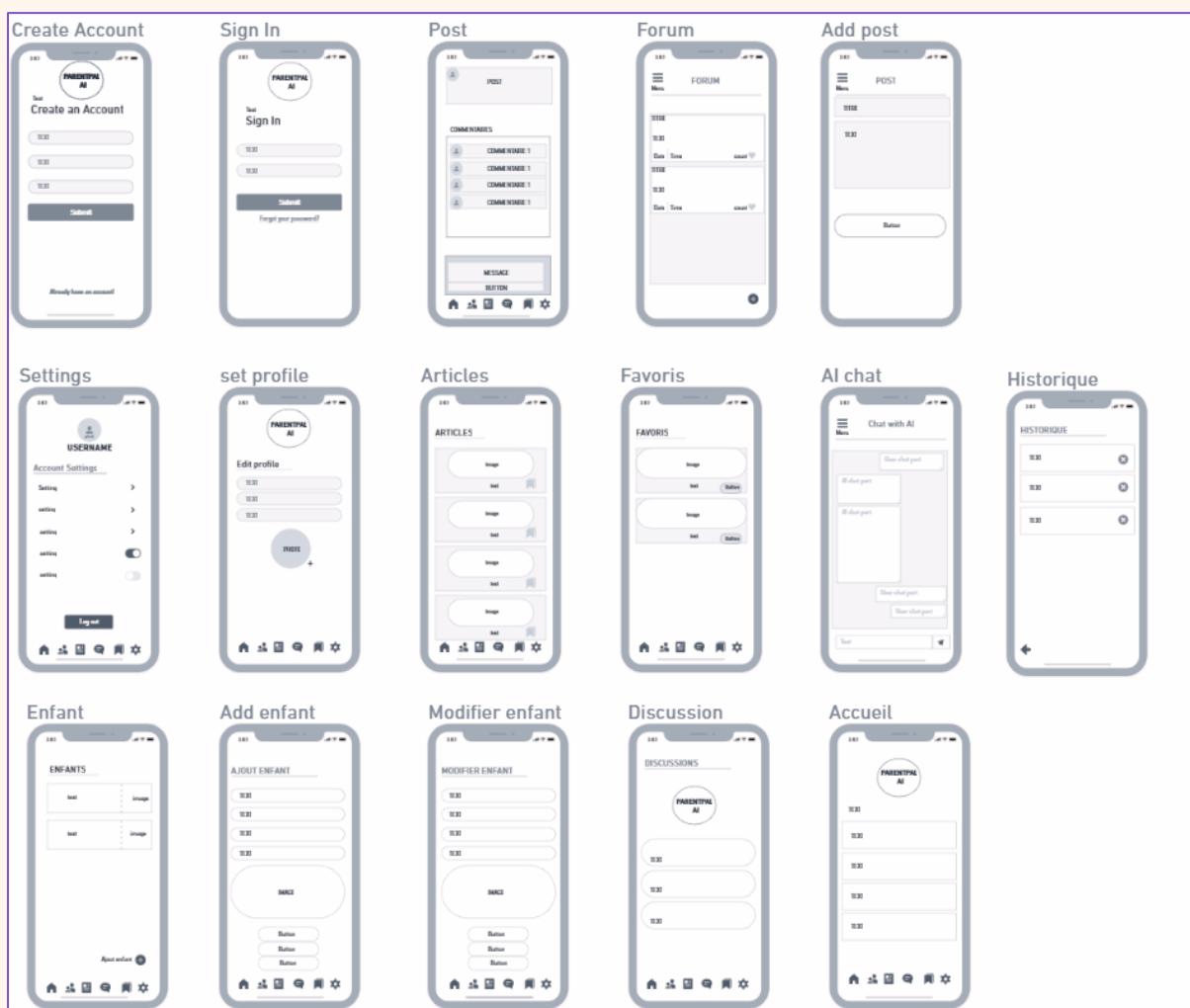
1.3.2 Test Utilisateur sans enfant

Nom du test				
Grandes étapes:				
Test de l'application Touche(s): Présentation de l'appli à un panel	Inscription/Connexion Inscription/Connexion	Création du profil enfant 1ers pas avec l'IA	Découverte du Forum	Articles généralistes Basés sur la tranche d'âge sélectionnée par l'utilisateur
Action du testeur: Clique sur le bouton "sign Up with Google"	Choisi la tranche d'âge correspondant à l'âge de l'enfant pour lequel il souhaite des infos	Renseigne profil enfant pour personnaliser l'expérience	Consultation des posts/commentaires Chatbot + rappels/calendrier	
PROFIL DU TESTEUR Géorgy Etudiant sans enfant Utilise l'appli pour la 1ère fois	GOALS CONCLUSION DU TEST: Obtenir des infos claires sur une tranche d'âge précise EXPERIENCE VECUE: Simple d'utilisation Accès facile à l'information Emet des réserves concernant l'IA	FONCTIONNALITÉS PRÉSENTES : Page 1 - Inscr... Page 3 - Article...	FONCTIONNALITÉS PRÉSENTES : Page 7 - Dialo... Page 5 - Journ...	FONCTIONNALITÉS PRÉSENTES : Page 8 - Foru... Page 4 - Article...
RESSENCE: Centralisation de l'information Echange avec les autres utilisateurs	PC Satisfait du sign up directement via Google.	Simple d'utilisation grâce au menu déroulant. UI claire	L'IA sera principalement utilisée pour les rappels	Le forum permettra de renforcer le côté humain des échanges.
Critical positive aspects of experience			Inquiétude quant à l'utilisation de l'IA et à la protection des données. La part de l'appli utilisant l'IA a par conséquent été réduite.	Le système de messagerie était jugé peu pertinant à la qualité des infos obtenues. Et peu pratique pour les partager au plus grand nombre.
Critical points for improvement				
ANALYSE DU TEST (POSITIVE ET NEGATIVE)				

1.4 Wireframes

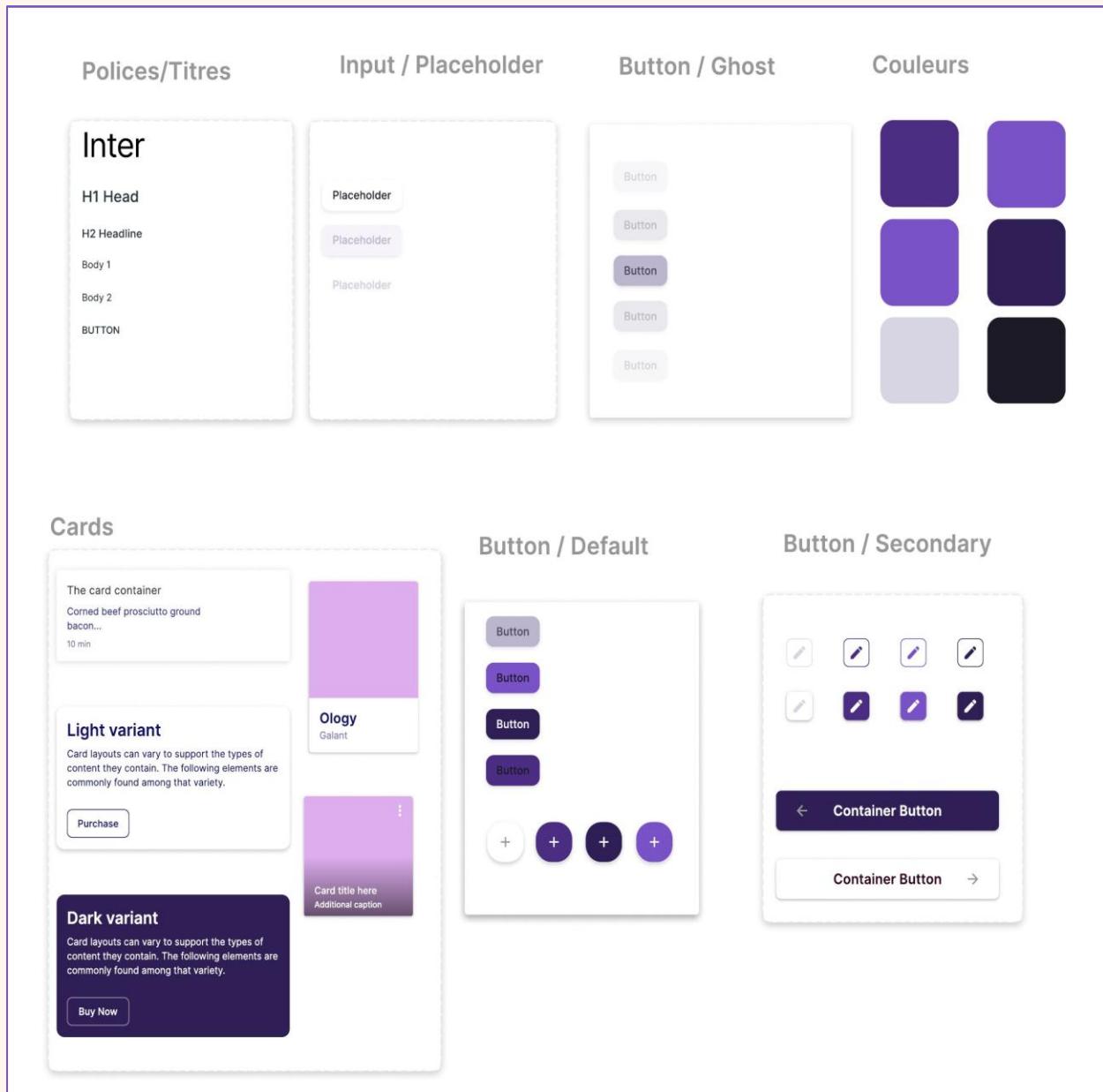
Ce sprint consistait à mettre au propre la version finale de notre application. Nous avons aussi dû repenser des écrans que nous n'avions pas pris en compte au début lorsque nous réfléchissions à comment structurer l'application. De légères modifications ont été faites lors de la production de celle-ci mais 95% de ce wireframe a été respecté.

Comme dit précédemment, l'aspect utilisateur avec ou sans enfant a été effacé, l'IA centralisée sur un Chatbot plutôt qu'avec un accès à l'application en général en rapport à la remarque des utilisateurs test qui étaient plutôt frileux vis à vis d'une utilisation de l'IA plus large.



1.5 UI/UX kit

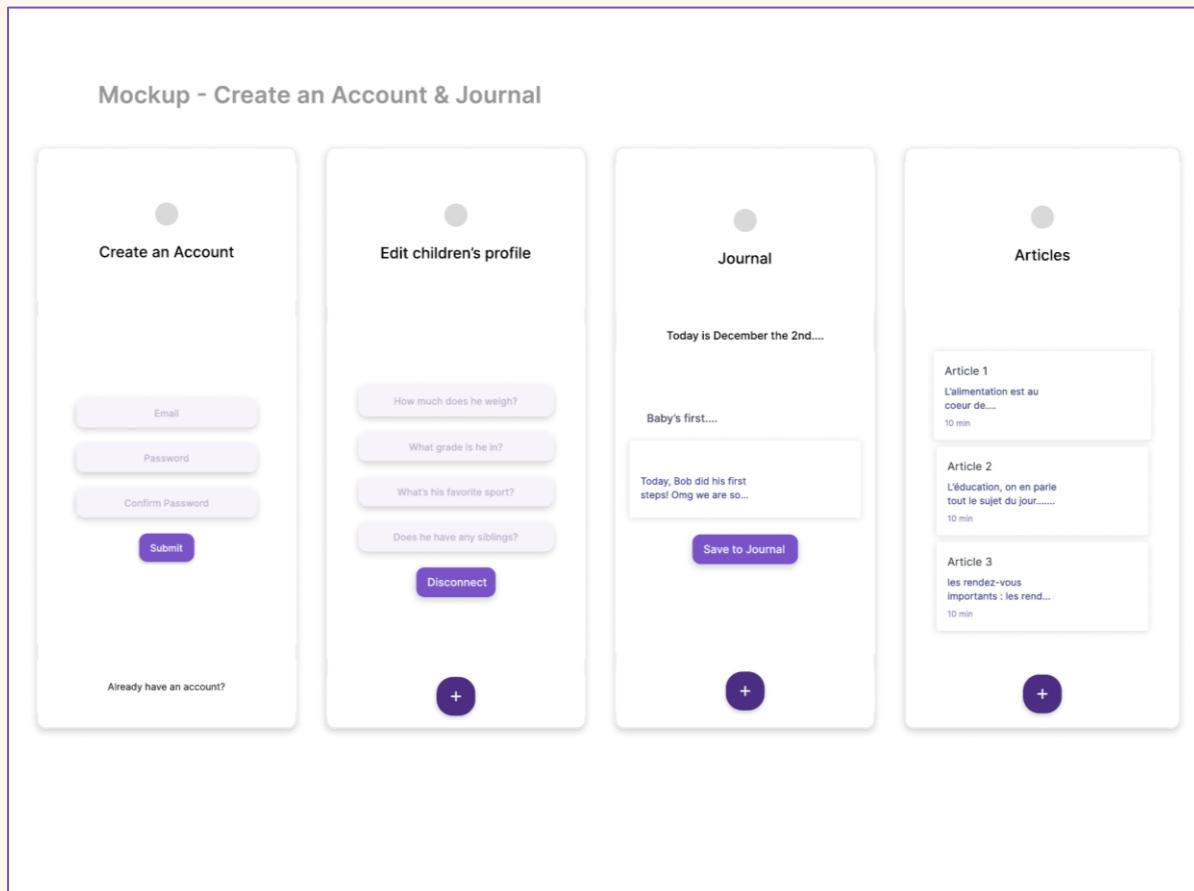
Pour faire l'UI kit, nous avons utilisé Figma . Celui-ci se présente sous cette forme : (lien Figma : [FIGMA PARENTPALIA UI KIT](#))



- Typographie : **Inter** (H1, H2, Body 1, Body 2, Button)
- Couleurs principales (hexadécimal) :
 - Primary : #7A52C7 (violet)
 - Primary Dark : #4B2E9F
 - Background : #F6F4FF
 - Card Background : #FFFFFF

1.6 Maquettes finales

Pour les maquettes finales, nous avons décidé d'appliquer un style sur 3 écrans afin d'avoir une idée générale du visuel qu'auront les écrans.



1.7 User stories

Ecran	User Story
Articles	En tant qu'utilisateur, je souhaite consulter une liste d'articles sur la parentalité. Je souhaite voir un aperçu de chaque article (image, titre, extrait). Je souhaite pouvoir sauvegarder des articles en favoris d'un simple clic. Je souhaite pouvoir cliquer sur un article pour lire le contenu complet.
Favoris	En tant qu'utilisateur, je souhaite retrouver facilement tous mes articles sauvegardés. Je souhaite voir un aperçu de chaque article favori. Je souhaite pouvoir retirer un article de mes favoris. Je souhaite accéder rapidement au contenu complet d'un article.
IA chat	En tant qu'utilisateur, je souhaite poser des questions à l'assistant IA sur la parentalité. Je souhaite que mes messages s'affichent à droite et les réponses de l'IA à gauche. Je souhaite pouvoir taper mon message dans le champ de saisie en bas. Je souhaite que l'IA connaisse le contexte de ma situation.
Historique	En tant qu'utilisateur, je souhaite retrouver toutes mes conversations passées avec l'IA. Je souhaite voir un aperçu de chaque conversation (titre, date). Je souhaite pouvoir supprimer une conversation. Je souhaite pouvoir reprendre une conversation en cliquant dessus.
Enfant	En tant que parent, je souhaite voir la liste de tous mes enfants. Je souhaite voir les informations principales de chaque enfant (nom, âge, taille, poids, photo). Je souhaite pouvoir ajouter un nouvel enfant facilement. Je souhaite pouvoir modifier les informations d'un enfant existant.
Add enfant	En tant que parent, je souhaite ajouter un nouvel enfant en renseignant son nom, sa date de naissance, sa taille et son poids. Je souhaite pouvoir ajouter une photo. Je souhaite valider l'ajout avec un bouton clair.
Modifier enfant	En tant que parent, je souhaite modifier les informations d'un enfant (nom, date de naissance, taille, poids). Je souhaite pouvoir changer sa photo. Je souhaite sauvegarder les modifications. Je souhaite pouvoir supprimer le profil de l'enfant si nécessaire.
Accueil	En tant qu'utilisateur connecté, je souhaite voir un message de bienvenue personnalisé. Je souhaite accéder rapidement aux fonctionnalités principales (Articles, Mes enfants, Discussions, Réglages). Je souhaite pouvoir accéder au forum directement.
Settings	En tant qu'utilisateur, je souhaite modifier mes paramètres de compte. Je souhaite pouvoir changer mon nom d'utilisateur. Je souhaite gérer mes préférences de l'application. Je souhaite pouvoir me déconnecter en toute sécurité.
Set profile	En tant qu'utilisateur, je souhaite modifier mon profil (nom, email, photo). Je souhaite ajouter ou changer ma photo de profil. Je souhaite sauvegarder mes modifications.

2 Préparation du projet

2.1 Compétences mobilisées

Afin de développer l'application ParentPal IA nous avons utilisé au maximum les compétences vues en cours ainsi que des recherches notamment ImagePicker pour la galerie photo et l'API google Gemini pour l'IA :

Frontend :

- React Native / Expo
- JavaScript
- React Navigation (Stack + Tabs)
- Redux Toolkit + Redux Persist
- Expo Camera + ImagePicker

Backend :

- Node.js / Express.js
- MongoDB / Mongoose
- Architecture MVC
- Système CRUD
- Cryptage bcrypt + uid2
- Cloudinary (images)
- Google Gemini API

Méthodologies :

- SCRUM (4 sprints)
- Jira/Atlassian
- Git/GitHub

2.2 Répartition des tâches

Pour le développement de l'application, nous avons opté pour 4 sprints répartis de cette façon :

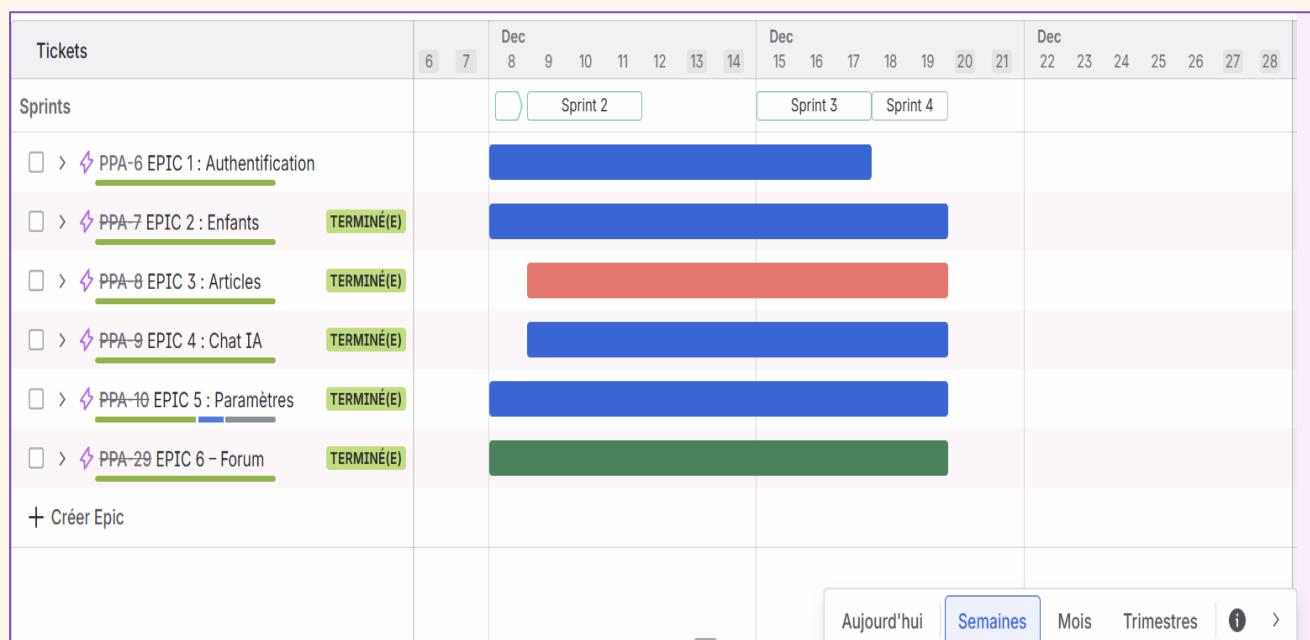
- Sprint 1 : 8 décembre au 9 décembre 2025
- Sprint 2 : 10 décembre au 12 décembre 2025
- Sprint 3 : 15 décembre au 17 décembre 2025
- Sprint 4 : 18 décembre au 19 décembre 2025
-

Pour déterminer les phases de développement, nous avons opté pour l'outil JIRA / Atlassian.

Voici le lien vers notre JIRA :

<https://parentpalai.atlassian.net/jira/software/projects/PPA/summary>

Après consultation des différentes pages à créer et en concertation avec les préférences de chacun, Léa a managé les tickets sur Jira pour les assigner aux différents membres du groupe.



Les sprints ont été répartis de cette façon :

- Tom :
 - Authentifications (page Login, sign up / Redux)
 - Forum complet (CRUD posts, commentaires, likes)
 - Chat IA (Gemini, conversations, rotation clés)
 - Upload photos (Camera + Galerie → Cloudinary)
- Léa :
 - UI/UX Design (Figma)
 - Articles (backend + frontend + NewsAPI)
 - Favoris (localStorage + backend)
 - Profils enfants (frontend)
- Félix :
 - Backend profils enfants
 - Navigation globale
 - Paramètres utilisateur
 - Tests et débogage

Planning des sprints :

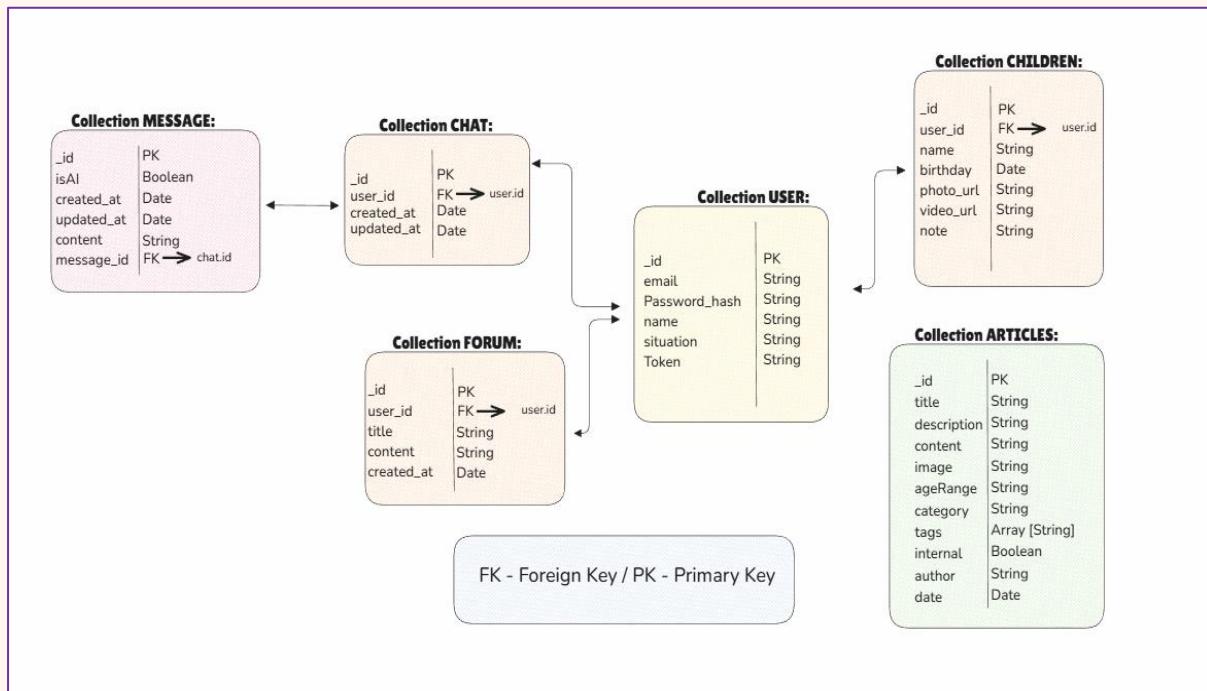
Sprint	Dates	Objectifs
1	8-10 déc	Fondations (auth, navigation, premiers écrans)
2	10-12 déc	Enfants, Articles, Forum, Backend
3	15-17 déc	Chat IA, Commentaires, Likes, Photos
4	17-19 déc	Débogage, Optimisations, Déploiement

2.3 Schématisation de la Base de Données

La base de données a été répartie en 6 collections :

- Collections Users pour stocker les informations des utilisateurs
- Collection Articles pour stocker les articles internes et externes sur la parentalité.
- Collection Messages pour stocker l'historique des conversations entre utilisateurs et l'IA.

- Collection Forum pour stocker les posts du forum communautaire avec leurs commentaires associés.
- Collection Children pour stocker les informations sur les enfants des utilisateurs pour le suivi personnalisé.
- Collection Chat pour gérer la conversation du user et de l'IA.



2.4 API du backend

2.4.1 Section Utilisateur

Route	Extension	Type	Rôle	Sécurité
/users	/signup	POST	Créer utilisateur	Bcrypt + uid2
	/signin	POST	Connecter utilisateur	Bcrypt.compareSync
	/me	GET	Récupérer le profil	Bearer token
	/me	PUT	Modifier le profil	Bearer token

2.4.2 Section Enfants

Route	Extension	Type	Rôle
/children	/	POST	Ajouter un enfant
	/me	GET	Récupérer les enfants du user
	:childId	PUT	Modifier un enfant
	:childId	DELETE	Supprimer un enfant

2.4.3 Section Forum

Route	Extension	Type	Rôle
/forum	/	POST	Créer un post
	/	GET	Récupérer tous les posts (populate userId)
	:postId	GET	Récupérer un post (populate comments.userId)
	:postId	DELETE	Supprimer un post (vérification propriétaire)
	:postId/like	POST	Toggle like

	/:postId/comment	POST	Ajouter un commentaire
	/:postId/comment/:commentId	DELETE	Supprimer un commentaire

2.4.4 Section Chat IA

Route	Extension	Type	Rôle
/chat	/send	POST	Envoyer message + réponse IA
	/history/:userId/:conversationId	GET	Historique de conversation
	/list/:userId	GET	Toutes les conversations
	/:userId/:conversationId	DELETE	Supprimer une conversation
	/rename	PUT	Renommer une conversation

2.4.5 Section Articles

Route	Extension	Type	Rôle
/articles	/	GET	Tous les articles de la BDD
	/:id	GET	Un article

2.4.6 Section Upload

Route	Extension	Type	Service	Rôle
/upload	/upload	POST	Cloudinary	Photo de profil du user
	/uploadChild	POST	Cloudinary	Photo enfant

2.4.7 Section Favoris

Route	Extension	Type	Rôle
/favorites	/toggle	POST	Ajouter/Retirer un favori
	/:token	GET	Récupérer favoris du user

3 Pilotage du développement

3.1 Liens vers le code source du projet

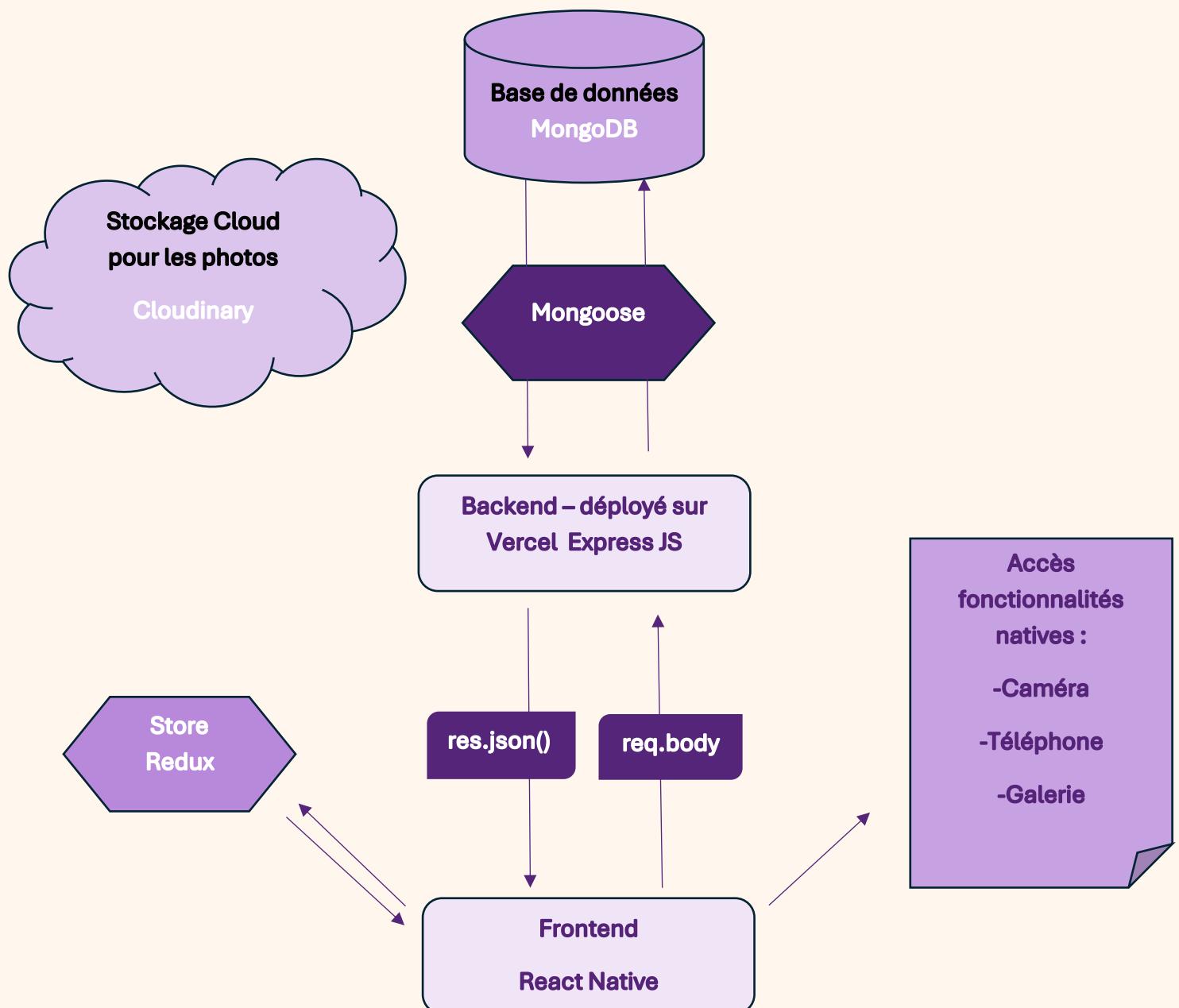
Pour réaliser ce projet, nous avons déployer notre code dans deux répertoires GitHub distincts, un pour le backend et un pour le frontend. Etant que trois personnes sur le projet, nous avons décidé de rester sur la même branche principale.

Voici les liens vers ces répertoires :

- ParentPal-AI frontend : <https://github.com/FelixHls/ParentPalAI-frontend>

- ParentPal-AI backend :<https://github.com/FelixHls/ParentPalAI-backend>

3.2 Schéma de l'architecture globale de l'application



3.3 Description de l'interface graphique en React

ParentPal AI est une application de soutien parental. L'interface privilégie une approche douce et rassurante, avec une palette de couleurs apaisantes dominée par des tons bleu pastel et vert menthe, évoquant la bienveillance et la croissance.

Dès la connexion, l'utilisateur accède à un dashboard personnalisé qui centralise les fonctionnalités clés : un assistant IA mis en avant, des articles recommandés selon l'âge de ses enfants, et un aperçu de l'activité communautaire. Cette page d'accueil sert de hub intelligent qui s'adapte aux besoins spécifiques de chaque parent.

L'application utilise une navigation par barre inférieure fixe avec cinq onglets principaux : Accueil, Chat IA, Articles, Forum et Profil. Cette structure permet une navigation fluide et intuitive entre les différentes sections sans perdre le contexte.

L'écran de chat avec l'assistant IA adopte une interface conversationnelle familière, similaire aux applications de messagerie, avec des bulles de discussion différencierées et des suggestions de questions pour guider les parents. Le design épuré met l'accent sur la clarté des échanges et la facilité d'utilisation.

Les articles sont présentés sous forme de cartes visuelles avec filtres par catégorie et âge, permettant aux parents de trouver rapidement des ressources pertinentes. Le forum communautaire suit une structure classique mais optimisée pour mobile, favorisant les échanges entre parents.

L'ensemble de l'interface privilégie la lisibilité avec des polices généreuses, des espaces blancs confortables et des animations subtiles qui renforcent le sentiment de fluidité sans distraire de l'objectif principal : accompagner les parents dans leur quotidien.

3.4 Composants et librairies utilisés

Composants / screens	Dépendances Frontend	Dépendances Backend
<pre>└ components └ AddButton.js └ ArticleCard.js └ ChildCard.js └ FavoriteCard.js └ SearchBar.js └ screens └ AddChildScreen.js └ ArticleDetailScreen.js └ ArticlesScreen.js └ ChatScreen.js └ ChildrenScreen.js └ CreatePostScreen.js └ DiscussionsScreen.js └ EditChildScreen.js └ FavoritesScreen.js └ ForumScreen.js └ HistoriqueMessagesScreen.js └ HomeScreen.js └ LoginScreen.js └ MessageScreen.js └ PostDetailsScreen.js └ SettingsScreen.js └ SignUpScreen.js └ UserProfileScreen.js └ UserScreen.js</pre>	<pre>"dependencies": { "@react-native-async-storage/async-storage": "^2.2.0", "@react-native-community/datetimepicker": "8.5.1", "@react-navigation/bottom-tabs": "7.0.0", "@react-navigation/native": "7.1.25", "@react-navigation/native-stack": "7.8.6", "@reduxjs/toolkit": "2.1.1", "expo": "~54.0.27", "expo-camera": "17.0.10", "expo-constants": "18.0.11", "expo-file-system": "19.0.21", "expo-image-picker": "17.0.10", "expo-linear-gradient": "15.0.8", "expo-status-bar": "3.0.9", "expo-system-ui": "6.0.9", "expo-updates": "29.0.15", "react": "19.1.0", "react-dom": "19.1.0", "react-native": "0.81.5", "react-native-safe-area-context": "5.6.0", "react-native-screens": "4.16.0", "react-native-web": "0.21.0", "react-native-webview": "13.15.0", "react-redux": "9.2.0", "redux-persist": "6.0.0" }, "devDependencies": { "@babel/core": "7.12.9", "jest-expo": "54.0.10" }, "private": true</pre>	<pre>"dependencies": { "-": "0.0.1", "@google/generative-ai": "0.24.1", "bcrypt": "6.0.0", "cloudinary": "2.8.0", "cookie-parser": "1.4.4", "debug": "2.6.9", "dotenv": "17.2.3", "express": "4.16.1", "express-fileupload": "1.5.2", "mongoose": "9.0.0", "morgan": "1.9.1", "node-fetch": "2", "nodemon": "3.1.11", "rss-parser": "3.13.0", "uid2": "1.0.0", "unid": "5.4.0" }, "devDependencies": { "jest": "30.2.0", "supertest": "7.1.4" }</pre>

3.5 contre-rendu des sprints

3.5.1 Réunion 1 (08/12/2025) :

Sujet : Lancement du projet + Sprint 1

Remarque(s) :

- Répartition des différentes tâches
- Objectif de finir les tâches avant la fin du sprint

3.5.2 Réunion 2 (10/12/2025)

Sujet : Débrief du sprint 1 et début Sprint 2

Remarques(s) :

- Le mot de passe oublié ainsi que la validation de l'email seront traités post MVP, le temps étant trop court, nous voulons nous focaliser sur les fonctions principales.

- Navigation faite par une Stack générale et TabNavigator, finalement une Stack nommée avec chaque élément a été choisie pour la compréhension des routes.
- Store Redux créé.
- Sprint 2 focus sur les fonctionnalités Enfants, Articles, Forum et Backend général

3.5.3 Réunion 3 (12/12/2025)

Sujet : Débrief du Sprint 2 et début du Sprint 3

Remarque(s) :

- Léa était Focus sur la partie Enfants et Articles, Félix était sur la partie backend générale et moi sur le Forum et le match backend Forum
- Un peu de retard a été pris sur la partie Articles car aucun site proposant une API matchait notre application. Finalement des articles générés en interne ont été simulé.
- J'ai dû finir plus tard les deux soirs car lors des merges sur GitHub, beaucoup de conflit ont été accepté et je voulais débuguer pour avoir une base saine le lendemain matin.

3.5.4 Réunion 4 (15/12/2025)

Sujet : Sprint 3 + prépa semaine

Remarque(s) :

- Mis en place du ChatBot avec Gémini. J'ai rencontré beaucoup de soucis dont une refonte complète des imports sur les routes, models et module. J'ai dû migrer tout les imports vers ES Modules car l'import GenAI n'accepte pas le require("").
- Gemini ayant une limite de 20 requêtes par jours, j'ai du augmenter la limite en ajoutant une rotation des clés API.
- Prompt pour que l'IA ai seulement une conversation correspondant au sujet de l'application.
- Résolution d'un souci entryPoint (v1Beta vs V1) qui m'empêchait d'utiliser l'API Gemini.
- Commentaires ajoutés dans la section Forum.

- Style mis en place par Léa et test du mode sombre sur l'application pour Félix mais finalement abandonné sur choix de Félix.
- Modification des routes par Félix pour éviter les erreurs/bugs du type 'TimeOut' en remplaçant tous les .then() en Async/Await.

3.5.5 Réunion 5 (17/12/2025)

Sujet : Sprint final, validation de tous les éléments, mise en place du style complet + débogage et peaufinage détails

Remarques :

- Déploiement sur Vercel du Backend et sur Expo pour le Frontend.
- Ajout de l'image sur la page profile du user et ajout d'une image pour les enfants.
- Intégration Caméra via Expo ainsi que de l'autorisation Galerie.
- Problème de compatibilité avec Android, étant sur Iphone je n'étais pas impacté mais j'ai essayé de créer une compatibilité en ajoutant le type de fichiers créés en fonction du support utilisé.
- (Félix) Correction du calendrier afin qu'il ne se ferme pas à chaque interaction, sur iOS. Il fonctionne déjà correctement sur Android. Suite à la correction, il est possible de sélectionner l'intégralité de la date avant de valider la fermeture du calendrier, sur iPhone.
- Ajout de la page Home et du Logo ParentPal IA sur celle-ci.
- (Félix) Suppression du champ "nombre d'enfants" sur la page UserProfileScreen Ce champ se remplissant automatiquement depuis la page ChildrenScreen, qui regroupe le nombre d'enfants enregistrés par l'utilisateur.
- Ajout des boutons manquants pour le retour aux pages précédentes et ajustement des détails
- Vérification complète des fonctionnalités de l'application, validation du style finale, derniers push et merges GitHub pour

une mise en commun, validation des bugs possibles et tests supplémentaires de l'application.

4 Mise en production

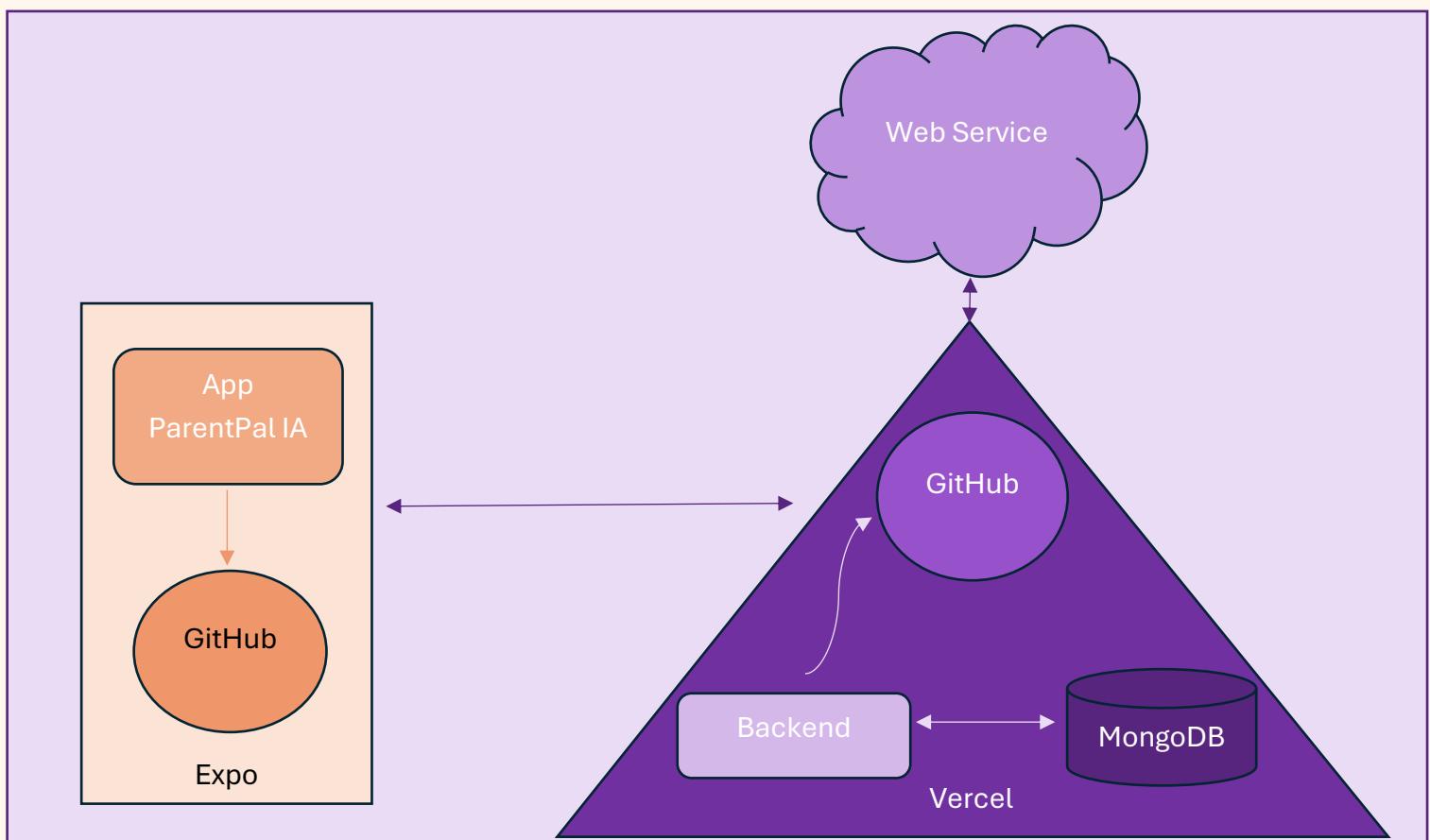
4.1 Mise en production du backend et du frontend

Dans le cadre du déploiement du projet, le backend et le frontend ont été déployés via deux canaux séparés. Le backend a été déployé sur Vercel et le frontend via Expo.

- Backend : [Vercel-link](#)
- Frontend : [Expo-link](#)

4.2 Schéma de l'environnement de déploiement

Le déploiement est illustré dans le schéma ci-dessous :



4.3 Système d'identification

Le système d'authentification permet aux utilisateurs de s'inscrire en fournissant leurs informations, sécurise les mots de passe avec bcrypt, génère des tokens uniques avec uid2 qui sont stockés côté client dans Redux, puis utilise ces tokens dans les en-têtes des requêtes pour vérifier l'identité des utilisateurs lors de chaque interaction avec l'API.

Voici l'ordre lors de l'inscription :

1. User fournit email, username, password
2. Password hashé avec bcrypt (10 rounds)
3. Token généré avec uid2 (32 caractères)
4. Stockage en BDD
5. Token renvoyé au frontend
6. Token persisté dans Redux Connexion :
 - 1. User fournit email et password
 - 2. Backend vérifie avec bcrypt.compareSync()
 - 3. Si OK, token renvoyé 4. Redux setUser() + persistence Requêtes authentifiées :
 - * Token envoyé dans Authorization: Bearer \${token}
 - * Backend vérifie existence en BDD
 - * Si valide → requête traitée
 - * Sinon → 401 Unauthorized

5 Conclusion

Avec ce projet lors de la fin de ma formation, a été l'occasion pour moi de pratiquer la majorité des sujets évoqués durant ma formation à La Capsule. Ce projet a été l'occasion d'évoluer en autonomie, d'appréhender les problématiques du travail d'équipe et de visualiser les différentes tâches qui incombent au chef de projet.

En plus du développement, le débogage et l'esprit d'équipe ont été très formateurs pour la suite. J'ai pu voir les problématiques lié au travail à plusieurs sur un projet, faire preuve d'autonomie et de recherche pour débuguer des situations pas forcément vu en cours, connaitre les différents outils mis à disposition sur internet comme StackOverFlow lors de questions techniques ou le site de React dev par exemple.

Une expérience des plus formatrices donnant l'envie de faire de ce métier mon quotidien.