

DESCRIPTIF DE PROJET

(3 pages recommandées)

CONCOURS EXTERNE D'INGÉNIEUR

DES SYSTÈMES D'INFORMATION ET DE COMMUNICATION

au titre de l'année 2025

NOM de famille (de naissance) : RUELLE

PRÉNOMS : Raphaël Jean Francis1

NOM d'usage :

Date de naissance : 22/09/1997

NOM et ADRESSE de l'entreprise :

EMERGENCE DESIGN

STATUT :

Entreprise SAS

DESCRIPTIF DE PROJET :

Introduction :

Dans le cadre de ma formation en 3ème année de licence ASR (administration système et réseau) à L'université Evry Val d'Essonne d'Évry-Courcouronnes, j'effectue un stage en télétravail du 02/05/2022 au 18/06/2022 au sein de l'entreprise « Emergence design ».

J'ai pu acquérir de nouvelles connaissances : ReactJS, en approfondir d'autre : HTML NodeJS utilisé pour la création d'une API permettant l'accès en lecture et/ou en écriture à la base de données.

Le but de cette licence est la formation de la maintenance informatique, la création de réseau, entretien d'un réseau, et/ou la gestion d'un parc de machine sur lesquels il faut créer et gérer les permissions des comptes en fonction des accès nécessaires pour les futurs utilisateurs. Ce cursus me permet d'acquérir, d'approfondir mes connaissances et les mettre en pratique lorsque je travaillerai.

Pour ce stage, j'ai été intégré dans le service informatique d'Emergence Design, où nous étions une équipe avec 1 chef de projet et 5 développeurs. Tout au long de ces semaines, nous avons été guidés par le président et le directeur général pour le traitement des tâches à réaliser. De plus, des réunions régulières permettaient de faire le point sur la progression du projet.

Au sein de l'entreprise, les différentes équipes sont managées par mon maître de stage dont la polyvalence est précieuse.

Grâce à Monsieur Pinto qui m'a accepté dans sa société, j'ai pu m'immerger dans le monde de l'entreprise, dans le métier de l'informatique et découvrir de nombreuses facettes du développement Web.

Les conditions de travail ont été très satisfaisantes. La communication, l'ambiance, ont été des éléments très importants pour fournir un travail le plus qualitatif possible.

Ma mission de stagiaire a été la construction du site du nouveau projet d'Emergence design, sobrement nommé « Projet Emergence », dans le but, à terme, d'être un réseau social afin de mettre en relation plusieurs personnes voulant créer un projet et/ou en rejoindre. L'autonomie et la liberté ont été les maîtres mots de ce projet. En cas de difficultés ou lorsque que l'un de nous ne comprenait pas une tâche demandée, nous n'hésitions pas à se concerter, l'entraide était régulière afin d'atteindre nos objectifs hebdomadaires. Je n'ai eu que peu de difficulté m'adapter au milieu professionnel qui est certes, vraiment différent du milieu scolaire, mais ayant réalisé des missions de longue durée ces dernières années, cela a été plus simple pour moi.

Tout au long de ce rapport, je parlerai de l'entreprise émergence design puis plus en détails du stage et ses différentes missions, j'exprimerai ensuite ses divers bénéfices et les difficultés rencontrées durant celui-ci. Je finirai sur une courte conclusion.

I – Présentation de l'entreprise « Emergence Design »

A- Présentation plus détaillée de l'entreprise « Emergence Design »

SIREN de l'entreprise : 912079068

Secteur d'activité de l'entreprise : Relations publiques et communication

Emergence Design est une entreprise récente qui ne demande qu'à se développer. Actuellement, elle est composée de 8 personnes à plein temps :

- M Idriss BEN SAID : Président et Business Développeur
- MME Ambre CARI : Directrice générale graphistes et designeuses
- MME Swannie CARI : Directrice générale graphistes et designeuses
- M Jonathan PINTO : Directeur général et consultant en marketing
- M Adilson GOMES GARCIA (notre tuteur) : Chef de projet

Les 3 derniers employés sont Louis, Denison et Mohamed qui exercent respectivement les fonctions de Photographe, Graphiste et d'Agent commercial.

L'entreprise propose plusieurs services à destination d'autres sociétés, principalement, créer une nouvelle image de marque, que ce soit par la création d'une nouvelle identité graphique, par l'élaboration de stratégies de communications, tout cela grâce

notamment à la création de nouveau logo, de site, de packaging sur mesure, ou encore par la réalisation de clip vidéo pour des campagnes marketing et publicitaire et de contenus pour les réseaux sociaux.

B- Historique de l'entreprise

Bien que l'entreprise soit encore très jeune : création le 06 février 2022, elle compte déjà plusieurs projets à son actif : Collaboration avec RedBull, ou encore « La Boutique du matelas » pour la création de leur nouveau site web, ou bien Foxy pour la création de leur site internet, du branding, de la conception de prints, de logo, d'images pour les réseaux, ou encore d'une vidéo de présentation. Emergence Design a aussi travaillé avec Paris barber shop pour la création d'une nouvelle identité graphique (charte graphique, nouveau design, nouveau logo) ainsi que la conception de design pour vêtements dans le but d'équiper les employés.

II – Le Stage

Durant ce stage, l'équipe et moi devions créer un site web type réseau social, complet et ergonomique. Les futurs utilisateurs pourront être mis en relation dans le but d'intégrer ou de créer différents projets innovants. Les « créateurs de projet » auront la possibilité à terme, de choisir des utilisateurs qui les intéressent en fonction de leurs compétences et expériences passées. Les utilisateurs, pourront aussi postuler à un projet, faire des postes sur un mur, et interagir avec ceux déjà créés par d'autres utilisateurs.

A- Le stage

Lors du premier jour, nous avons eu le droit à une réunion d'ouverture du Président expliquant les besoins et les technologies à utiliser. C'est à ce moment, que nous avons appris, que ReactJS était une technologie qui devait impérativement être utilisée pour ce projet. Ensuite, les jours suivants étaient surtout dédiés à la conception du cahier des charges et à la documentation sur les technologies que nous allions utiliser tout au long de ce stage notamment ReactJS, qui était tout nouveau pour moi.

A la suite de cette réunion, nous savions quelles tâches nous incombaient, qui serait notre chef de projet (un autre stagiaire s'étant donc attelé à la conception du cahier des charges et à la répartition du travail).

Lors de cette réunion, nous avons pris la connaissance de la méthode de management employée qui était la méthode SCRUM, il s'agit d'une méthode agile s'adaptant particulièrement bien aux projets informatiques et encore plus aux projets web. Nous travaillons notamment en sprint, ici, 2 sprints de 3 semaines soit 6 semaines au total ainsi qu'une semaine de recherches intensives. Cette dernière n'est évidemment pas incluse dans ces sprints.

Chaque jour nous participions à une succincte réunion d'ouverture, pour définir la répartition des tâches, et de fermeture dans laquelle nous faisions un compte rendu de nos tâches du jour, de leurs réussites ou non.

Les avancées étaient présentées à Idriss Ben Said, Jonathan Pinto ainsi qu'à mon tuteur, qui donnaient des indications sur la direction du projet et des choix à faire.

Pour y parvenir nous utilisons l'outil Trello qui était tout désigné pour l'organisation des tâches, pour indiquer lesquelles sont terminées ou non et leurs répartitions dans le groupe.

B- Nos différentes missions

Il y avait différentes missions pour chacun de nous :

- La conception des maquettes du site.
- La conception de l'API pour la communication entre la base de données et l'application web.
- La conception de l'application en elle-même et de son design.

Ma mission principale était de créer l'API permettant la communication avec la base de données, la mise en place de l'interface qui permettra l'utilisation de l'API sur l'application web.

Je ne parlerai que brièvement des missions qui ne m'étaient pas désignées

Mission 1 : Conception des maquettes du site

En fonction des différents besoins énoncés par le cahier des charges, nous avons créé les différentes maquettes représentant les schémas des futurs pages du site web, cette mission était impérative, car sans elle, nous ne pouvions commencer la conception du reste de l'application web et son design.

Tout cela a pu être réalisé à l'aide de l'outil « Adobe XD », un outil d'UI/UX design, permettant la conception de design d'interfaces utilisateur pour les applications mobiles et Web.

La maquette présente un formulaire intitulé 'Ajouter une formation' sur un fond gris clair. Le formulaire est structuré avec des sections distinctes : 'École' (avec un champ de texte pré-rempli 'Université Paris'), 'Diplôme' (avec un champ de texte pré-rempli 'Licence'), 'Domaine d'études' (avec un champ de texte pré-rempli 'Informatique'), 'Date de début' (avec deux menus déroulants pour 'Mois' et 'Année'), et 'Date de fin' (avec deux menus déroulants pour 'Mois' et 'Année'). En bas du formulaire, il y a deux boutons : 'Ajouter une formation' (en bleu) et 'Suivant' (en gris).

Figure 1 : Maquette pour la page « Ajouter d'une formation »

D'autres maquettes seront en annexes.

Mission 2 : Conception de la base de données

La base de données devait suivre plusieurs règles : l'inscription d'un utilisateur, y associer ses différentes expériences et ses formations ; pouvoir créer un projet, que l'utilisateur puisse suivre et/ou en intégrer un ; poster un article consultable par les autres utilisateurs, interagir avec, que ce soit avec un like, un commentaire ou encore, en le partageant.

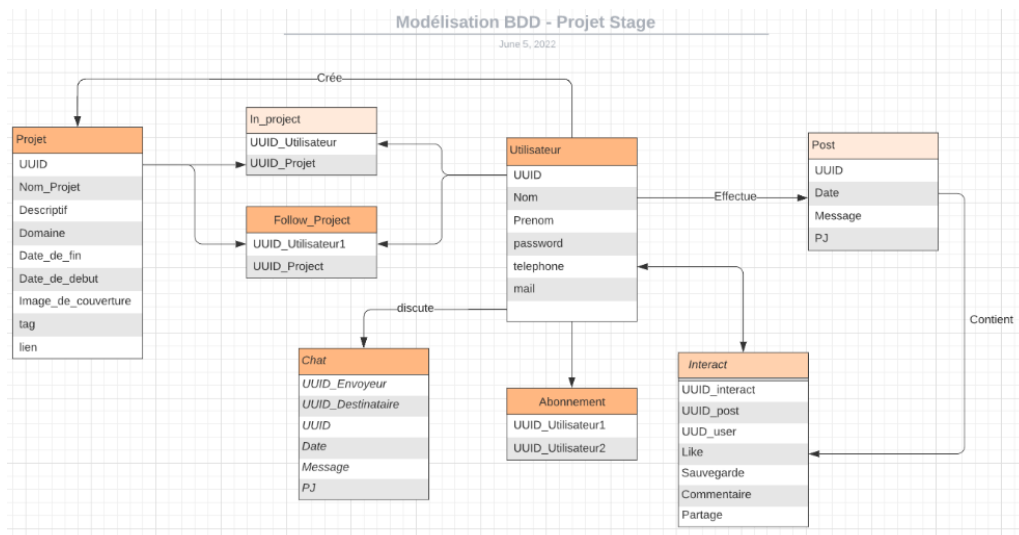


Figure : Modélisation initiale de la base de données

Pour la conception de la base de données, nous avons choisi la technologie « SQL » car aucune technologie ne nous était imposée, nous avons donc choisi une solution avec laquelle nous sommes à l'aise afin de gagner du temps. Pour y parvenir nous avons utilisé PostgreSQL, un système de gestion de base de données relationnelle et objet.

Le SQL pour « Structured Query Language » créé en 1974 est un langage informatique permettant d'exploiter les bases de données, il permet de rechercher, d'ajouter, de modifier ou de supprimer des données dans nos bases de données.

Mission 3 : API pour la communication avec la base de données

- Introduction :

Lors de nos recherches, nous avons remarqué qu'il était impératif d'utiliser une API pour accéder à la base de données depuis notre application ReactJS, car l'app web étant côté client, il nous fallait une interface qui réceptionnerait et enverrait les données nécessaires à son bon fonctionnement. C'est la mission qui m'a été donnée.

Une API pour « Application Programming Interface » est une interface de programmation d'applications qui permet la communication entre 2 applications. Dans notre cas, elle permet la communication entre notre base de données et notre application web ReactJS.

Pour réaliser notre API, nous avons besoin de Node.js et d'une librairie « expressjs ».

Node.js est une solution d'environnement libre, open-source qui permet la création d'applications simple et très évolutive du côté serveur. NodeJS utilise le langage « Javascript » comme indiqué dans son nom.

Express.js est un Framework qui permet la construction simple et rapide d'application web par son aspect minimaliste et utilisant Node.js. Il est très courant de l'utiliser pour la conception d'API web. Je l'ai choisi car très utilisé, il possède donc une documentation assez fournie.

Il nous faudra installer aussi la librairie « cors », en effet, cors est important car dans notre cas, il permet à un client « distant » d'accéder à l'api d'un serveur ayant une origine différente, par exemple, une application ayant l'URL <http://localhost:5000> ne pourrait pas communiquer avec une API ayant l'URL <http://localhost:3000> sans la librairie cors installée.

Non réservé, le port de l'app à 5000 sera défini à l'aide d'une constante ; si le changement du port de mon application est nécessaire, je ne modifierai que ma constante et ainsi le port sera modifié dans tous les endroits où ma constante est appelée.

Après cela, on doit indiquer à notre application que l'on va utiliser cors avec nos options choisies.

Dans notre application, on est capable d'envoyer des informations à notre API ou encore en recevoir, on le fait grâce à des requêtes POST, et grâce à des requêtes GET, la requête POST peut dans certains cas, renvoyer une réponse par exemple un message de confirmation. Il existe d'autres types de requêtes, mais ne les ayant pas utilisés dans ce projet, je préfère ne pas en parler.

Après avoir créé la base de l'application, on peut maintenant la lier à la base de données PostgreSQL en installant le module « pg » de NodeJS au même titre que cors ou ExpressJs.

- Inscription :

Pour la création de compte, nous avons besoin d'un identifiant unique pour chacun des utilisateurs inscrit dans la base de données et les rendre non prévisible, on appelle cela un UUID (Universally unique identifier). Pour y parvenir, nous utilisons donc une librairie du nom de « uuid » qui va réaliser cette tâche très simplement.

Dans le cas d'une inscription, notre but est de récupérer les données venant du formulaire d'inscription, que ce soit le nom, le prénom, l'adresse mail, le mot de passe et la date de naissance. Pour y parvenir, l'application WEB devra donc utiliser la méthode POST. La route pour s'inscrire qui a été choisie est « api/register ».

Pour valider une inscription, il ne faut pas oublier de :

- Vérifier que les données existent
- Vérifier qu'il n'y a pas de compte déjà existant dans la base de données
- Créer un UUID unique
- Enregistrer un mot de passe hasher dans la base de données
- Envoyer un message d'erreur en cas d'échecs des vérifications ou lorsque la base de données est inaccessible

Hasher un mot de passe revient à calculer son empreinte unique à partir des données fournies. Une fois les données hashées, on ne peut revenir aux données de base. C'est pour cela que nous l'utilisons pour stocker des mots de passes en base de données,

il faut donc préalablement hasher le mot de passe reçus lors d'une future connexion selon les mêmes contraintes dans le but de réaliser une comparaison avec celui stocker en base de données.

Dans le cadre de la création de compte, nous devons faire en sorte que l'utilisateur puisse ajouter une ou des expériences professionnelles, ainsi que leurs compétences, le stage n'étant pas encore terminé, cette fonctionnalité n'a pas encore été implémenté à l'API. J'espère pouvoir la terminer avant la fin du stage.

- Connexion :

Pour la connexion, on fait aussi une requête POST sur laquelle l'API va recevoir le mail et le mot de passe saisi par l'utilisateur sur l'application ReactJS. La route choisie pour la connexion est « api/login ».

Pour valider la connexion on doit :

- Vérifier que les données existent
- Vérifier qu'il y a un compte déjà existant dans la base de données
- Vérifier que le mot de passe reçu est identique au mot de passe enregistré en BDD
- Vérifier que l'association email/mot de passe est correcte
- Envoyer un message d'erreur en cas d'échecs des vérifications ou quand la base de données inaccessible

Pour plus de sécurité, nous avons utilisé un système de token qui respecte la norme du « JSON Web Token », un standard. Il permet l'échange sécurisé de Jetons (Tokens) entre au moins 2 applications. En effet, cette sécurité de l'échange est permise par vérification de l'intégrité et de l'authenticité des données reçues et envoyées.

Une fois la connexion validée, on envoie donc un « accessToken » et un message de confirmation de connexion à l'application reactJS

- Afficher les infos du profil :

Pour afficher les infos de l'utilisateur, on fait aussi une requête GET sur la route « /api/me ». En effet, l'app ReactJS place le token en entête de la requête que l'on récupère sur l'API, cela va nous permettre de vérifier que l'utilisateur est bien connecté. En effet, si le token est valide, L'API envoie les données liées au profil de l'utilisateur à l'application ReactJS, comme le nom, le prénom, le mail, et l'uid.

- Suppression d'un compte :

Pour supprimer un compte d'utilisateur, on fait aussi une requête quel qu'elle soit sur la route « /api/deleteaccount » car l'app react nous envoie encore une fois en en-tête de requête, le token qui va nous permettre de vérifier que l'utilisateur est bien connecté. En effet, si le token est valide, l'utilisateur est bien connecté, on envoie une requête à la base de données pour supprimer le compte. En cas d'échec de communication avec celle-ci, on envoie un message d'erreur à l'utilisateur.

En cas de réussite, on envoie un message de confirmation à l'utilisateur tout en supprimant le token. Dans le cas où on ne le supprimerait pas, on pourrait forcer la connexion sur un compte qui n'existe plus.

- Création d'un projet :

La création d'un projet se base sur le même principe que la création d'un compte, on envoie une requête POST sur la route « /api/createprojet » avec les informations du projet à envoyer. De plus, contrairement à la création de compte, il faut identifier la personne connectée à l'aide de son token. Une vérification de l'existence d'un projet du même nom est réalisée afin de pas faire de doublon. De plus un UUID unique lui est attribué, et pour plus de compréhension on y ajoute le préfix « projet », on a donc un UUID sous la forme « projet- XXXXXXXX-XXXX-XXXX-XXXX-XXXXXXXXXXXX ».

Mission 4 : L'application web et de son design

Pour l'application web, nous avons comme contrainte d'utiliser ReactJS et donc NodeJS.

J'ai dû créer son API qui permet l'accès en BDD, j'ai donc réalisé les différentes fonctions à l'aide de la librairie NodeJS « Axios » qui permet l'exploitation d'une API par une application ReactJS.

Pour autoriser l'accès à certaines routes, il fallait être connecté, j'ai dû donc créer un moyen qui permet de détecter que l'on est bien connecté. De plus si un utilisateur n'est pas connecté et qu'il a un accès hypothétique à une page non autorisé, l'api n'envoie pas les données, puisqu'elle n'aura pas reçu de token valide.

L'essentiel de l'application n'a pas été développé par moi-même, ce sont mes collègues qui étaient désignés pour cette tâche, il en est de même pour son design réalisé à l'aide du CSS (Cascading Style Sheets).

Comme pour la partie principale de l'application, je ne me suis pas occupé de son design mais elle devait suivre une charte graphique qui nous a été fournie le premier jour de stage.

Vous trouverez cette charte graphique en annexe.

III – Les bénéfices de ce stage

Ce stage a été bénéfique pour moi, car j'ai été plongé dans le monde de l'entreprise. J'ai mis en pratique mes connaissances et en ai acquis de nouvelles comme ReactJS ou encore la création d'API qui était tout nouveau pour moi. De plus, j'ai amélioré mes compétences déjà acquises en cours, ou bien en autodidacte.

IV – Les difficultés rencontrées lors du stage

Lors de ce stage, j'ai rencontré quelques difficultés, notamment avec NodeJS, bien qu'apparis en autodidacte n'étant donc pas une découverte. Cela était surtout dû à la librairie ExpressJS, étant toute nouvelle pour moi, j'ai eu du mal à détecter et à comprendre les différentes erreurs qui se présentaient.

L'utilisation de token a aussi été une épreuve pour moi, en effet, trouver une solution pour identifier un utilisateur n'était pas simple, aujourd'hui si on recharge la page web, on est automatiquement déconnecté. Je voudrais à terme, stocker le token dans les cookies afin de palier à ce problème.

J'ai eu aussi beaucoup de difficultés à mettre en application les fonctions asynchrones propre à javascript, elles m'ont fait donc perdre du temps, mais grâce à l'aide de mes recherches, mes collègues de stage et des différents employés qualifiés de l'entreprise j'ai pu les surmonter et j'espère pouvoir terminer le projet à temps.

IV – Conclusion

Ce stage a été très intéressant pour moi, car il m'a appris à connaître plus en profondeur le monde du travail, et m'a permis de le voir d'un œil nouveau.

Dans ce stage, j'ai pu aussi apprendre à beaucoup travailler en autonomie, car nous étions tous, plus ou moins livrés à nous-mêmes, cela m'a d'ailleurs permis de trouver de nouvelles méthodes de travail, et d'organisation dans un travail colossal qu'est la création d'un site web.

Malgré les difficultés que j'ai pu rencontrer, ce stage m'a conforté dans mon choix de carrière, l'informatique dans sa généralité est en effet quelque chose qui me passionne, concevoir quelque chose et le voir naître et évoluer progressivement à un effet très satisfaisant. Il est donc tout naturel pour moi que je veuille en faire un choix de carrière professionnel.

J'espère que l'entreprise développera dans son secteur d'activité, cela semble en bonne voie à la vue des réalisations actuelles de cette jeune entreprise.

Maquette de la page « Créer un projet ». Le formulaire est intitulé « Descriptif du projet » et est structuré en sections :

- Nom du projet** : Champ de texte avec la valeur « Emergence ».
- Domaine du projet** : Champ de texte avec la valeur « Informatique ».
- Date de début** : Deux champs de sélection (Mois et Année).
- Date de fin** : Deux champs de sélection (Mois et Année).
- Description du projet** : Champ de texte avec le placeholder « Décrivez votre projet... » et un bouton « Suivant ».
- Médias** : Bouton « +Ajouter une image de couverture ».
- Tag** : Bouton « +Ajouter un tag à votre projet ».
- Liens** : Bouton « + Ajouter un lien ».

Figure : Maquette de la page « Créer un projet ».

Maquette de la page « Ajouter une expérience ». Le formulaire est intitulé « Ajouter une expérience » et est structuré en sections :

- Intitulé de poste** : Champ de texte avec la valeur « Chef de projet informatique ».
- Type d'emploi** : Champ de sélection avec la valeur « Veuillez sélectionner ».
- Nom de l'entreprise** : Champ de texte avec la valeur « Microsoft ».
- Date de début** : Deux champs de sélection (Mois et Année).
- Date de fin** : Deux champs de sélection (Mois et Année).
- Lieu** : Champ de texte avec la valeur « Paris, France ».
- Secteur** : Champ de texte avec la valeur « informatique ».
- Descriptif** : Champ de texte vide.

À la fin du formulaire, il y a deux boutons : « Ajouter une expérience » et « Suivant ».

Figure : Maquette de la page « Ajouter une expérience ».

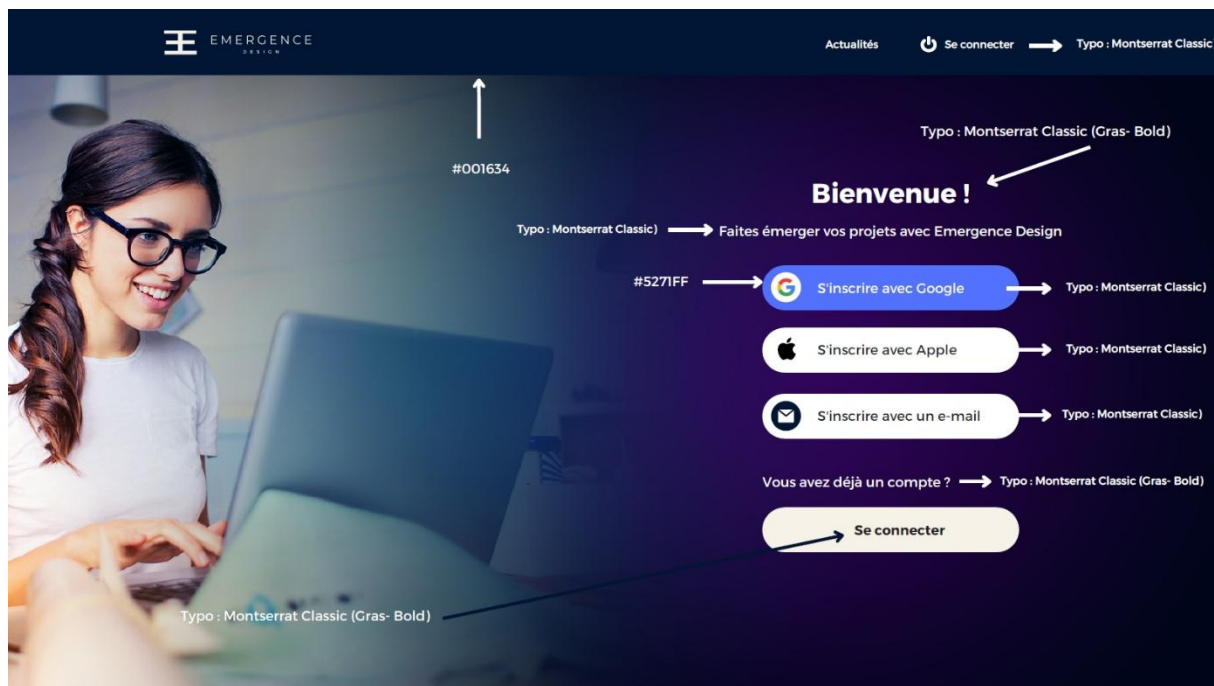


Figure : Charte graphique pour la page d'accueil s'appliquant pour la plupart des pages.

Bibliographie

Dave Gray. (2021). React JS Form Validation | Axios User Registration Form Submit | Beginners to Intermediate. Récupéré sur Youtube: <https://www.youtube.com/watch?v=brCHK3P6ChQ>

Documentation Axios. (s.d.). Récupéré sur <https://axios-http.com/>

Documentation Cors. (s.d.). Récupéré sur npmjs.com: <https://www.npmjs.com/package/cors>

Documentation ExpressJs. (s.d.). Récupéré sur expressjs.com: <https://expressjs.com/fr/starter/installing.html>

Documentation ReactJS. (s.d.). Récupéré sur reactjs.org: <https://fr.reactjs.org/docs/getting-started.html>

From Scratch - Développement Web. (2020). [Javascript Full-stack] Projet MERN | Créer un réseau social | Partie Back-end. Récupéré sur Youtube: <https://www.youtube.com/watch?v=SUPDFHuvhRc>

Göbel, D. (2021). Promise <pending>. Récupéré sur damaris-goebel.medium.com: <https://damaris-goebel.medium.com/promise-pending-60482132574d>

Gray, D. (2022). React User Login and Authentication with Axios. Récupéré sur Youtube: <https://www.youtube.com/watch?v=X3qyxo UTR4>

herdoycode. (2022). React Axios | get data, Update data and Delete data using Axios | next.js how to Working with API. Récupéré sur Youtube: <https://www.youtube.com/watch?v=O yUe8qEIYM>

Le Designer du Web. (2021). Tuto authentification avec React + Firebase. Récupéré sur Youtube: <https://www.youtube.com/watch?v=boZJtNzRCDQ>

Morgan Schiedt. (2020). Programmation asynchrone en JavaScript: Événements, Callbacks, Promises, async-await. Récupéré sur Youtube: <https://www.youtube.com/watch?v=nf0FvGwAQBq>

Morgan, J. (2020). How To Add Login Authentication to React Applications. Récupéré sur digitalocean.com: <https://www.digitalocean.com/community/tutorials/how-to-add-login-authentication-to-react-applications>

Sebastian, N. (2020). Getting started with Postgres in your React app. Récupéré sur logrocket.com: <https://blog.logrocket.com/getting-started-with-postgres-in-your-react-app/>

TheiPhoneRetro. (2017). API REST Node.js • Système d'authentification par tokens JWT (Json Web Tokens). Récupéré sur Youtube: <https://www.youtube.com/watch?v=NPJms-kq2F8>

Traversy Media. (2018). Node.js With Passport Authentication | Full Project. Récupéré sur Youtube: <https://www.youtube.com/watch?v=6FOq4cUdH8k>

Tyler Potts. (2021). Build a Login System in NodeJS with Passport.js Authentication | A NodeJS Tutorial. Récupéré sur Youtube: <https://www.youtube.com/watch?v=W5Tb1Mleg-l>

Wawa Sensei. (2021). Tuto Authentification/Refresh JSON Web Token en Nodejs avec Express. Récupéré sur Youtube: <https://www.youtube.com/watch?v=GXokEYwbOwA>