

Groupe n°14:

GENTIL Driss

CLERC Louis

HIGGINS Alec

DELHOMME Baptiste

KADIRI Saad

DAIKH Nassim

LINGO PEER

by lingo team



Sommaire

Sommaire	2
1) Cadrage	3
1.1) Définition du contexte	3
1.2) Définition des objectifs	3
1.3) Analyse du terrain (enquêtes) et de l'existant	4
1.4) Contraintes et risques (mitigation)	8
1.5) Suivi des risques	9
2) Expression des besoins	10
2.1) Identification et priorisation des besoins fonctionnels	10
2.2) Identification et priorisation des besoins non-fonctionnels	11
2.3) Spécification - Modélisation	11
2.4) Cohérence besoins - Modélisation	12
2.5) Mise en avant des choix de critères qualité logicielle	13
2.6) Mise en avant des critères ergonomiques	13
3) Solutions	17
3.1) Présentation et analyse des solutions techniques	17
3.2) Gestion du travail	18
3.3) Description d'un modèle de développement	18
Annexe:	19

1) Cadrage

1.1) Définition du contexte

Sujet du DeptInfoChallenge : *“Ré-inventons le distanciel”*

Suite aux événements récents qui nous ont affectés, le monde a pu voir à quel point internet est complet en termes d'applications pour toutes sortes de choses. Cependant, le fait de devoir rester enfermé chez soi a entravé la communication et la sociabilité entre tous. Et pour couronner le tout, il est très compliqué de voyager en ce moment. Nous en sommes venus à la conclusion que l'aspect international et sociable d'internet pouvait bénéficier d'approfondissement. Pour pallier le manque d'exotisme dans la vie sociale de chacun ou bien pour bien encadrer les échanges avec des personnes de tout endroit sur le globe, et surtout pour passer le temps de manière productive, nous allons créer une application d'échange dans toutes les langues.

Le but de cette nouvelle application est de permettre aux utilisateurs de pouvoir améliorer et conserver leur niveau en langues mais aussi de contrôler et guider les échanges. Cela devrait permettre de converser de manière non agressive/offensive avec des inconnus mais aussi d'avoir des sujets de discussion. De plus, il est possible de choisir la manière de discuter pour être plus à l'aise dans la discussion. C'est dans ce cadre là que le nom de “linguo peer” est apparu: ‘linguo’ pour la racine latine de langue et ‘peer’ pour souligner l'interaction entre les utilisateurs.

1.2) Définition des objectifs

Objectifs techniques:

Pour la réalisation de la web app, voici les caractéristiques attendues: En premier lieu, permettre des conversations entre utilisateurs de manière écrite, orale ou audiovisuelle. Ensuite, il est envisageable de pouvoir répartir les profils des utilisateurs dans des groupes uniques par le biais de leur niveau de langue, la ou les langue choisie(s) et les centres d'intérêts. Afin de créer les conversations les plus adaptées en fonction de l'affinité des participants, la mise en place d'une roulette est préférée. Par ailleurs, il n'est pas impossible de concevoir une gamification de l'application avec un système de missions en lien avec l'apprentissage des langues, des points gagnés au fur et à mesure pour fidéliser les utilisateurs et potentiellement un avatar pour l'utilisateur. Enfin il est primordial de posséder une interface homme-machine propre, épurée et agréable à l'œil.

En outre, il est préférable de permettre des connexions dites “Peer to peer” pour les conversations utilisateurs.

Finalement, il est souhaitable de stocker les données des utilisateurs dans une base de données afin de pouvoir s’en servir.

1.3) Analyse du terrain (enquêtes) et de l’existant

Dans cette partie nous allons parler des différentes analyses que nous avons réalisées afin de pouvoir mener à bien notre application web. Nous allons commencer par l’analyse du terrain, pour ensuite parler de l’analyse de l’existant.

L'analyse du terrain passe par des enquêtes, c’est pour cela que nous avons décidé de faire un sondage sur l’intérêt du public envers notre future application. Ce sondage nous a permis de déterminer que les utilisateurs cibles de notre application seraient les jeunes entre 15 et 30 ans comme le démontre la figure 1. Nous avons trouvé que ce groupe d’âge est ressorti car il contient beaucoup de lycéens/étudiants voulant maintenir leur niveau en langue mais qui sont également intéressés par des voyages linguistiques. Parmi ces étudiants, on trouve des natifs non francophones qui sont restés en France durant le confinement et qui souhaitent converser leur niveau dans leur langue natale ou dans une autre, cf la figure 2. Nous avons aussi constaté que les personnes appartenant à cette tranche d’âge attendent de notre service une vraie interaction sociale, voir figure 3.

Veuillez nous indiquer votre age:

26 réponses

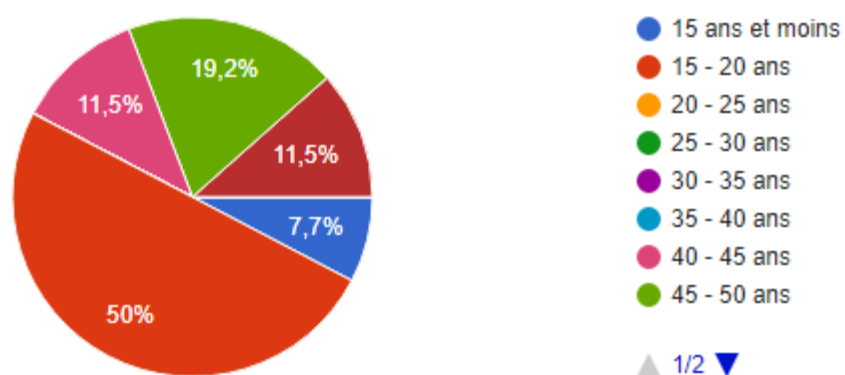


Figure 1: Age des participants

Utiliserez-vous un site d'échange dans d'autres langues, afin d'apprendre une nouvelle langue ou pour entretenir celle-ci ?

26 réponses

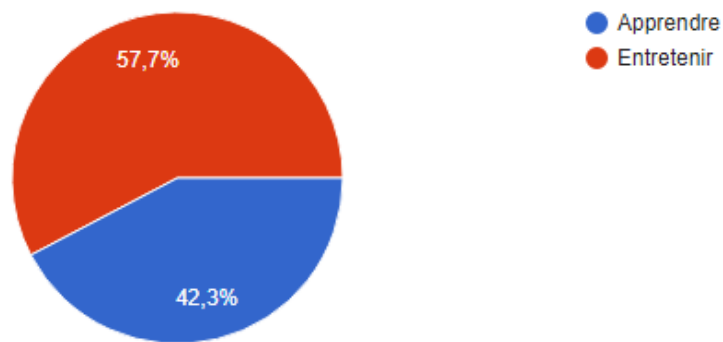


Figure 2: L'intérêt des participants

Cherchez-vous uniquement un moyen de communiquer dans une autre langue ou une interaction sociale ?

25 réponses

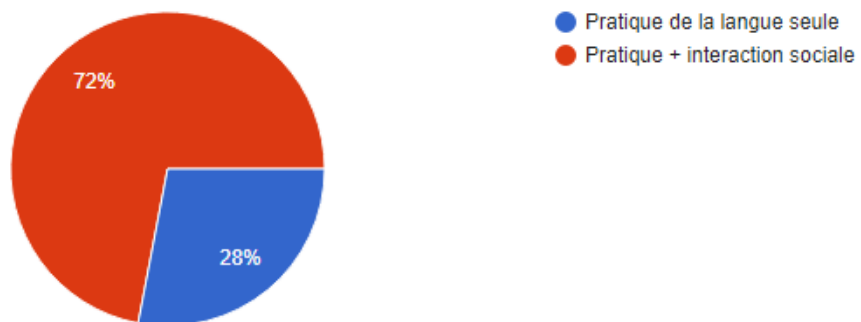


Figure 3: Les envies des participants

Une deuxième partie de l'analyse du terrain repose sur l'environnement de travail que nous devons utiliser. Après avoir étudié les avantages et les inconvénients de plusieurs éditeurs de texte comme Atom, Visual Studio Code ou bien NotePad++, nous avons décidé d'utiliser Visual Studio Code. Ce choix est fondé sur le fait que Visual Studio Code soit l'éditeur de texte le plus complet, en effet il propose plus d'outils de programmation que les deux autres. Pour la programmation en elle même, nous allons utiliser différentes librairies et framework JavaScript. Nous avons envisagé une multitude de frameworks et nous avons finalement choisis "react" et "WebRTC" car ils correspondent le plus à ce que nous souhaitons faire. Pour la gestion des données nous avons décidé d'utiliser soit mongoDb, soit postGres. Pour réaliser la partie serveur de notre application web, nous avons trouvé que nodeJs correspond le mieux à nos besoins.

Etant donné que nous réalisons un web app il n'y aura aucune contrainte lié aux plateformes. Lingo Peer sera accessible via internet donc de partout.

L'analyse de l'existant permet de se situer par rapport à la concurrence, les applications d'apprentissage de langues sont nombreuses. Les plus ressemblantes sont :

- *Verbling*, c'est une application web déjà bien ancrée dans les applications d'apprentissage de langue grâce à plusieurs avantages : l'association avec un professeur vérifié, une variété de formation différentes avec la possibilité de demander à son professeur un programme personnalisé et des outils d'échange efficaces comme des chatbox ou des appels vidéos. Disponible sur android et IOS en plus du navigateur web, beaucoup de langues sont disponibles (+50) et donc une communauté large et accueillante. Néanmoins le service est payant (environ 20\$ par cours), ce qui le limite à une catégorie d'utilisateurs définie. De plus, des imperfections sont à noter concernant le bon fonctionnement de tous les outils, notamment le chat box.

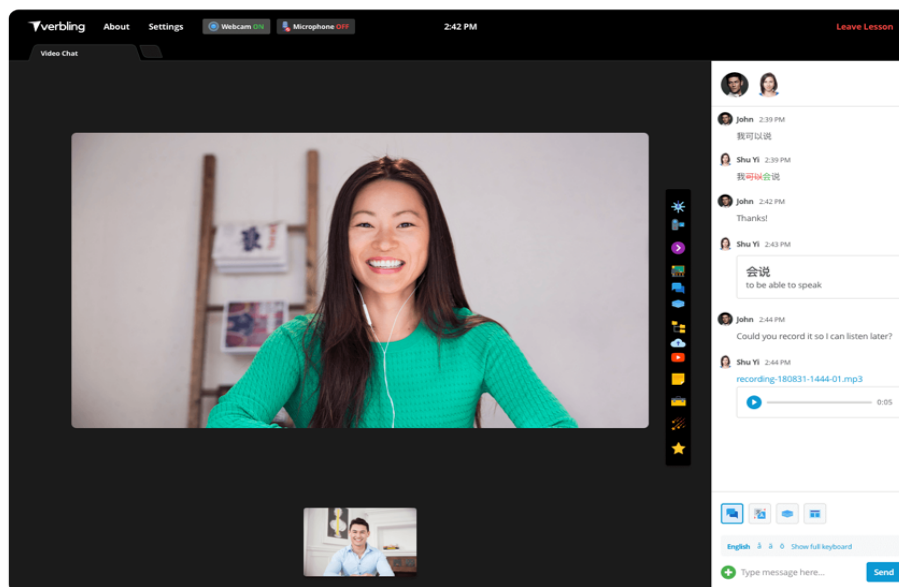


Figure 4 : *Verbling* - Appel vidéo (gauche) avec chatbox (droite)

- *Jaaser.com*, Cette application web n'est pas encore très connue mais elle a différents points forts : un système de chat efficace, en effet le but principal est d'avoir un correspondant ce qui permet d'apprendre d'une manière moins scolaire. Une communauté récente dont des personnes pratiquant l'espéranto (une langue sous représentée), enfin l'application est complètement gratuite et est disponible sur mobile. Mais pas d'appel vidéo ni audio disponible, seulement la correspondance écrite, de plus sur d'anciennes versions l'interface n'est pas ergonomique.

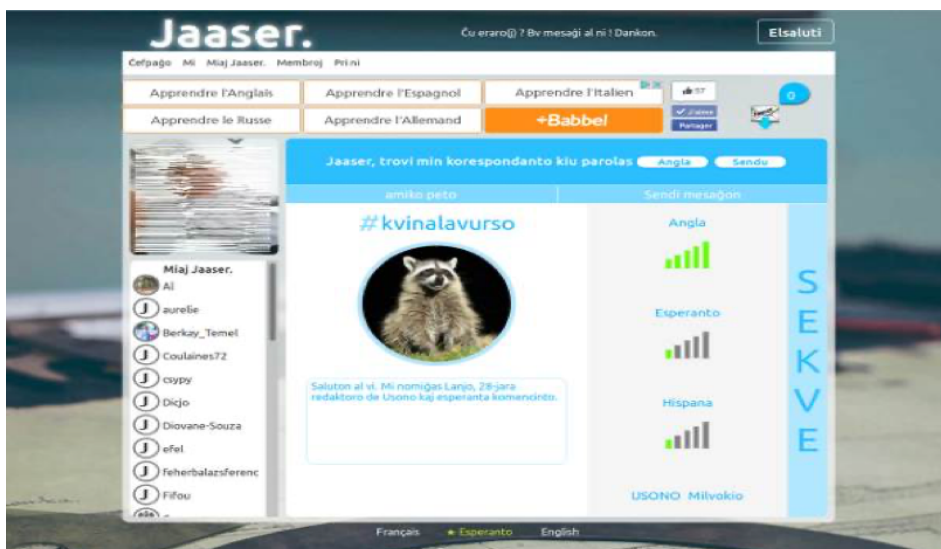


Figure 5: *Jaaser* - Page d'accueil

- *Babel*, c'est l'application référence en matière d'apprentissage linguistique, proposant ses services sur mobiles et sur navigateur web. Voici ces avantages : les cours sont créés par des professionnels en langue, les méthodes d'apprentissage sont efficaces (environ 2 mois pour parler une langue), les outils mis en place pour l'apprentissage comprennent des dialogues pratiques, des compréhensions orales, une reconnaissance vocale et un outil de révision. Interface très épurée et ergonomique. Malgré ça, différents inconvénients sont à noter : une approche trop scolaire, un entraînement à l'oral réalisable uniquement seul et un service payant (abonnement d'environ 6 €).

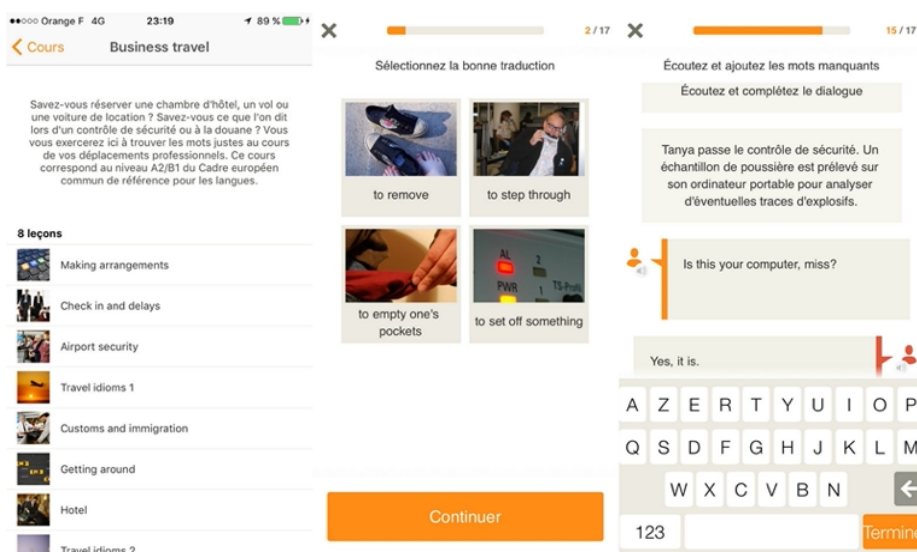


Figure 6 : *Babel* - Cours à base de texte (gauche), Exercices visuels (milieu) et Exercices écrits (droite)

Cette analyse de l'existant nous permet de saisir les points forts et faibles des différents sites qui existent déjà. Nous souhaitons donc rassembler un maximum de ces points forts sans les points faibles. Il paraît dès lors évident de posséder une interface propre, professionnelle et fonctionnelle tout en permettant d'échanger de la manière souhaitée avec son interlocuteur. Enfin il est souhaitable de créer une application gratuite qui regroupe plusieurs langues, dont des sous-représentées. Nous allons définir ceci plus en détail dans la partie suivante.

1.4) Contraintes et risques (mitigation)

Pour réussir un projet, il faut évaluer en amont les contraintes et les risques.

- De manière générale le respect de la vie privée des utilisateurs est primordial, par conséquent il est important d'évaluer la protection de toutes les données en contrainte à respecter. Ce qui est donc une contrainte légale.
- Une contrainte de temps nous est imposée dans ce projet, avec 3 itérations et un rendu attendu à la fin de chacune. On doit donc s'adapter à cette contrainte et respecter les consignes des itérations 1, 2 et 3:
 - n.1 Se positionner sur le sujet, trouver des idées de solutions et se renseigner sur les méthodes techniques utilisables.
 - n.2 Confirmer nos idées et choisir les méthodes techniques qui seront utilisées, faire la conception de l'application et commencer à la réaliser.
 - n.3 Finir de réaliser notre application dans le détail et la peaufiner, se débarrasser des erreurs et commenter le code.
- On est également confronté à une contrainte de moyens. Cela se traduit par les nouveaux outils et langages que nous allons utiliser dans ce projet, et que nous ne maîtrisons pas encore. Nous devons donc minimiser cette contrainte en nous entraînant à utiliser ces nouveaux outils et langages.

En ce qui concerne les risques nous en avons identifié 3 :

R1 = Un reconfinement ou un passage en distanciel d'où la mise en place dès maintenant de tout ce qui est nécessaire pour continuer le travail malgré la distance.

R2 = l'implémentation de la connexion vidéo. Au vu du temps et des technologies nécessaires pour mettre cela en place nous évaluons la non implémentation de la connexion vidéo comme un risque

R3 = l'interaction entre les frameworks. Nous ne sommes pas certains de la compatibilité entre les frameworks qu'on utilisera pendant le projet.

Pour schématiser les risques on fait une matrice de criticité :

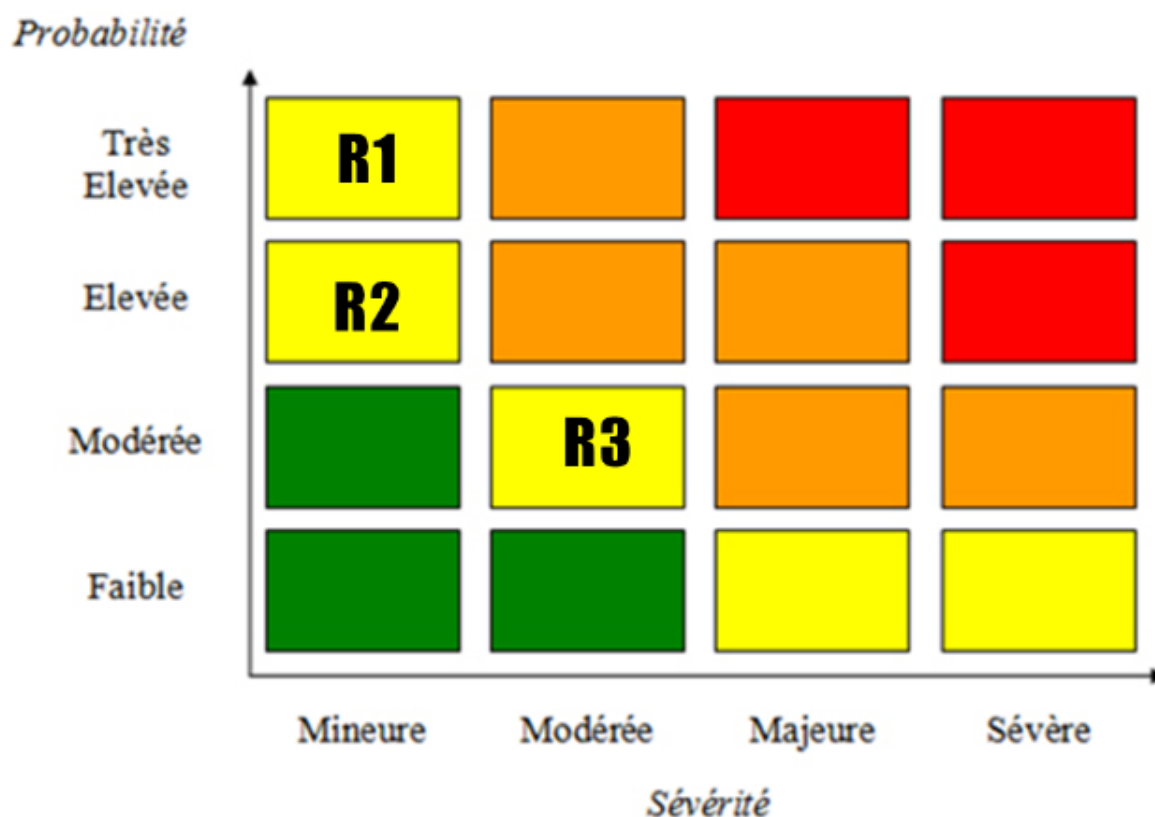


Figure 7: Matrice de criticité des risques

1.5) Suivi des risques

Pour la première itération, nous avons identifié 3 risques. Aujourd'hui un nouveau risque est à prendre en compte. C'est le risque que un ou plusieurs membres du groupe soit atteint du COVID-19. Ici on présente le suivi des risques présentés lors de la première itération :

R1 : Nous sommes maintenant confinés à l'impact que cela a sur le projet est mineur/modéré car toutes les dispositions ont été prises pour assurer la continuité du projet. Mais le confinement peut atteindre la motivation de certains et complique également un peu la communication entre l'ensemble du groupe et les professeurs.

R2 : à ce stade le risque de la non implémentation de la connexion vidéo n'est pas encore réglé. Mais nous pensons de plus en plus à l'exclure pour nous concentrer sur le chat écrit et vocal.

R3 : La question de la compatibilité entre les frameworks est réglée. on s'est informés et suite à nos tests nous avons vu qu'ils étaient tous compatibles donc ce n'est plus un risque.

R4: Pour ce nouveau risque, on a pour l'instant eu un seul cas dans le groupe qui n'a pas contaminé les autres heureusement. Mais pour la suite si on venait à avoir d'autres cas dans le groupe cela pourra impacter grandement l'avancement de notre projet.

2) Expression des besoins

2.1) Identification et priorisation des besoins fonctionnels

Priorisation des besoins fonctionnels	Nom des besoins fonctionnels	Explications des besoins fonctionnels
1	Permettre à l'utilisateur de saisir toutes les informations nécessaires et les stocker dans une base de données	Les informations seront : le nom, prénom, âge, adresse mail, langage parlé, langue que l'on souhaite apprendre, thèmes. Pour le nom, le prénom, et l'adresse mail on mettra en place des champs textes pour l'âge, langage parlé et langue que l'on souhaite apprendre et thèmes on mettra en place des listes déroulantes.
2	Mettre en relation deux personnes en faisant en sorte que leurs centres d'intérêts correspondent	Les personnes mises en relation devront avoir choisi entre apprendre, enseigner ou discuter. le binôme sera composé de : apprentissage/enseignement ou discuter/discuter.
3	Faire en sorte que les utilisateurs puissent communiquer en chat écrit	
4	Mettre en place un chat audio	Au préalable l'accord de l'utilisateur sera demandé.
5	Mettre en place une connexion vidéo	Au préalable l'accord de l'utilisateur sera demandé.
6	Permettre aux utilisateurs de s'ajouter en amis pour garder contact	Mise en place d'un bouton « Ajouter en ami » pendant une discussion. On pourra le retrouver dans une liste d'amis sur son profil.
7	Mettre en place une fonction qui permettra à l'utilisateur de changer rapidement de conversation	Mise en place d'un bouton « changer d'utilisateur » pendant une discussion.
8	Mettre en place un fil de partage	Un endroit où les utilisateurs peuvent partager des publications, mettre des j'aime et des commentaires.

9	Paramétrer son compte	Sur le profil de l'utilisateur avoir la possibilité de modifier son : avatar, mot de passe, adresse mail. Pouvoir aussi changer les langues et les centres d'intérêts.
---	-----------------------	---

2.2) Identification et priorisation des besoins non-fonctionnels

Priorisation des besoins non-fonctionnels	Nom des besoins non-fonctionnels	Explications des besoins non-fonctionnels
1	Fiabilité de l'application web.	Évaluer l'aptitude d'un système à maintenir son niveau de service : tolérance aux pannes, conditions de remise en service, maturité...
2	Facilité d'utilisation.	Caractériser l'effort nécessaire à un utilisateur potentiel pour utiliser le système : facilité de compréhension, d'apprentissage, d'exploitation ou encore pouvoir d'attraction.
3	Critère ergonomique.	Respecter les 8 critères ergonomiques de Bastien & Scapin (détaillé en 2.3)
4	Maintenabilité de l'application web.	Évaluer les possibilités de faire évoluer le système dans le cas de nouveaux besoins

2.3) Spécification - Modélisation

User

- *connexion*: se connecter à son compte.
- *déconnexion*: se déconnecter de son compte.
- *matchMaking*: trouver deux usagers qui possèdent la même langue à pratiquer et des thèmes en commun.

Inscrit

- *demandeAmi*: envoi une demande d'ami à un autre utilisateur inscrit.
- *envoyerMessage*: envoyer un message à un ami.
- *accéderProfil*: accès à son profil.

- *addLangue*: ajout d'une langue à la liste de langues que possède un utilisateur inscrit.

Invité

- *inscription*: se créer un compte Lingo Peer.

Message : un message entre des utilisateurs.

Paramètres : des paramètres de conversation.

Serveur : le serveur qui s'occupe des conversations.

- *connect*: se connecter au serveur.
- *afficherMessage*: affichage d'un message dans une conversation.

2.4) Cohérence besoins - Modélisation

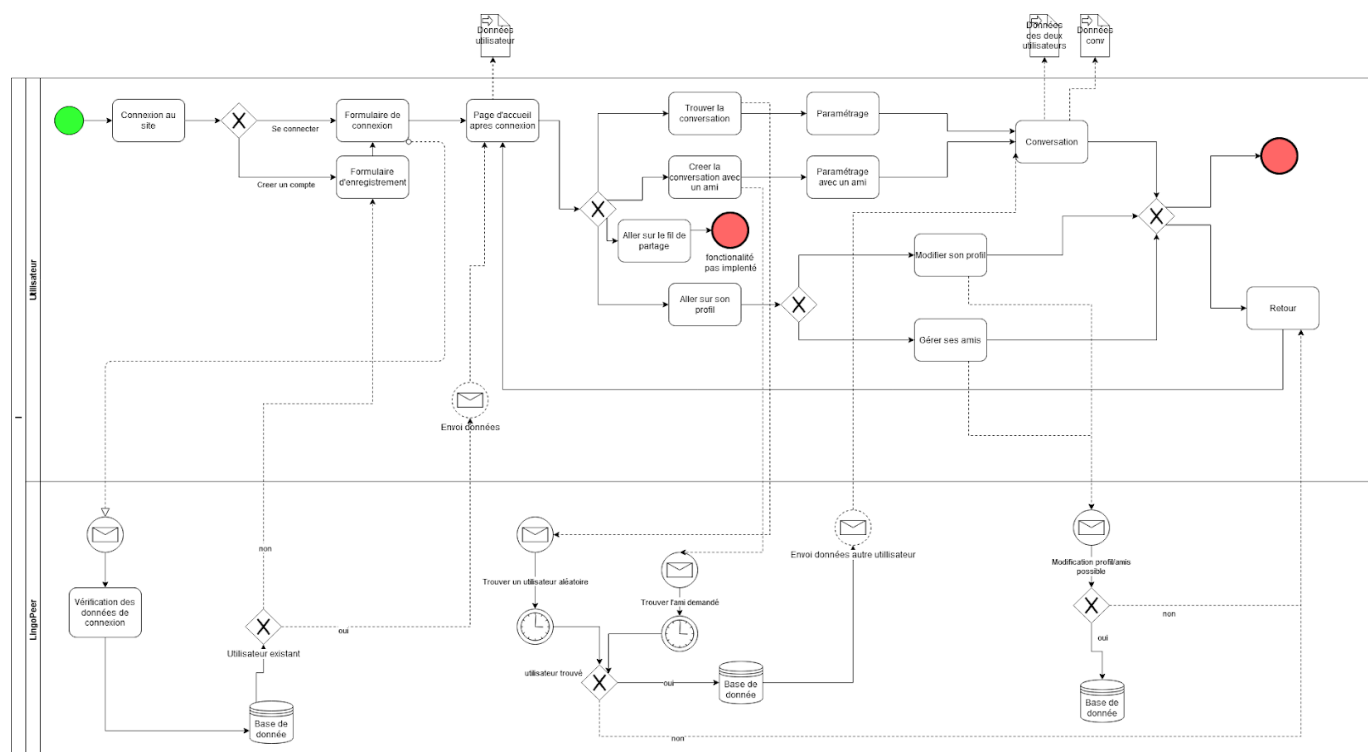
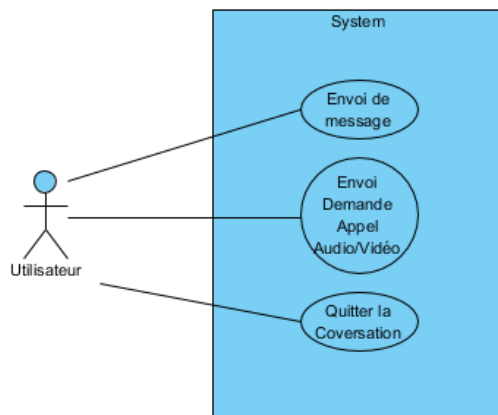


Figure 8 - BPMN lié au scénario

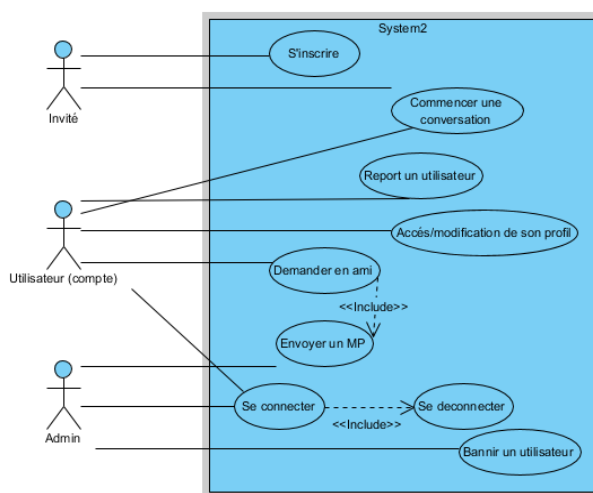
Ici nous allons voir de quelle manière nous modélisons les interactions entre les utilisateurs ainsi que l'application afin de répondre au mieux à nos besoins .



Tout d'abord voici les trois fonctions principales que devra pouvoir réaliser notre application ,

Il faut pouvoir envoyer des messages cela est primordiale au fonctionnement de l'application que nous avons déjà mis en place aussi s'ajoute la fonction d'appels vidéo que nous devons encore mettre en place à ce stade du développement et enfin pouvoir quitter la conversation car il est important à l'utilisateur est le contrôle totale de l'application.

Figure 9: Cas d'utilisation 1 (Conversation)



Ici nous voyons une liste plus détaillée de chaque besoins et on voit les fonctions que les besoins définis plus haut correspondent.

Figure 10: Cas d'utilisation 2 (Accueil App)

À des fins de lisibilité, le diagramme de classes est situé en annexe, et il tend à évoluer.

2.5) Mise en avant des choix de critères qualité logicielle

Il s'agit d'un ensemble d'attributs permettant de vérifier si le logiciel répond aux besoins fonctionnels exprimés. On y retrouve notamment les notions de pertinence, d'exactitude, d'interopérabilité, de sécurité et de conformité.

- Fiabilité : évaluer l'aptitude d'un système à maintenir son niveau de service : tolérance aux pannes, conditions de remise en service, maturité...
- Facilité d'utilisation : caractériser l'effort nécessaire à un utilisateur potentiel pour utiliser le système : facilité de compréhension, d'apprentissage, d'exploitation ou encore pouvoir d'attraction.
- Maintenabilité : évaluer les possibilités de faire évoluer le système dans le cas de nouveaux besoins

2.6) Mise en avant des critères ergonomiques

Notre projet devra respecter les 8 critères ergonomiques de Bastien & Scapin

Les critères ergonomiques précisés dans la section précédente doivent se refléter et suivre les règles des critères ergonomiques de Bastien et Scapin:

Tout d'abord on peut voir sur la page d'accueil le contrôle explicite, et respecter celui-ci permet à l'utilisateur de comprendre explicitement où il va être redirigé suivant ses actions. Ici le bouton se connecter permet simplement de comprendre qu'il va être redirigé sur une page de connexion (fig.11), de même avec le bouton créer un compte (fig. 11).

On voit aussi qu'il y a un guidage avec des grandes images des dispositions familières à l'utilisateur très imposantes et de manières égales et que la disposition est respectée entre les différentes pages.(fig.11)

Il y a aussi une gestion des erreurs en grisant les zones non accessibles pour certains utilisateurs afin d'éviter une génération d'erreurs au niveau du code.(fig.12)

Page d'accueil avant connexion :

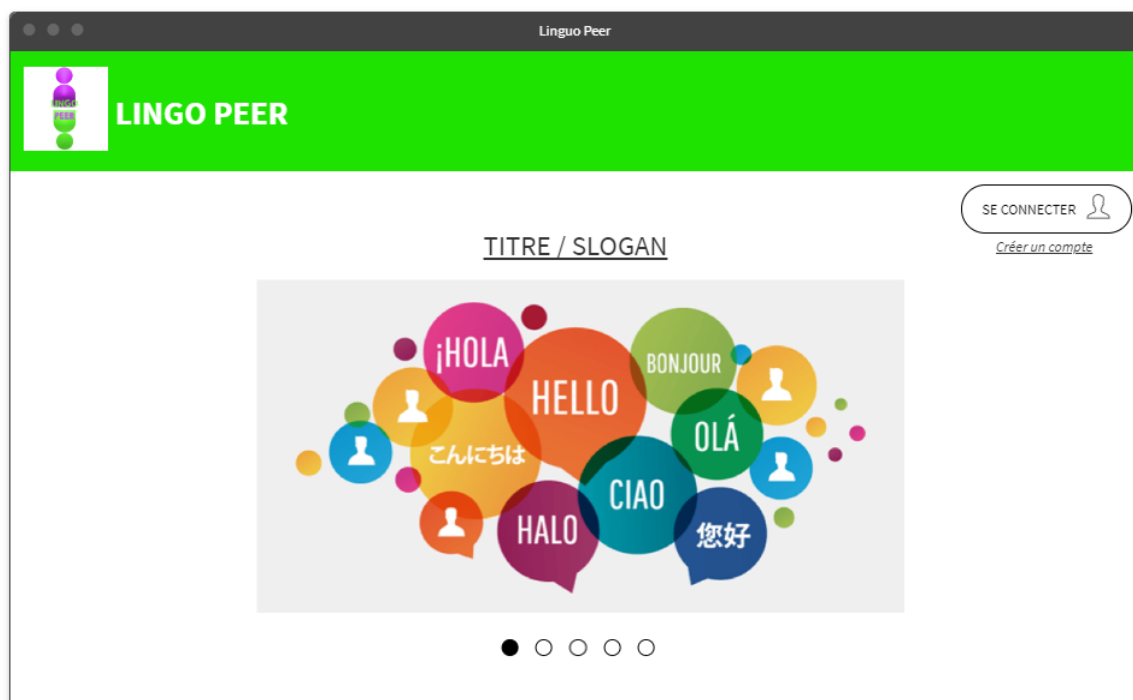


Figure 11

Page d'accueil après connexion avec fonctionnalité indisponible grisée :

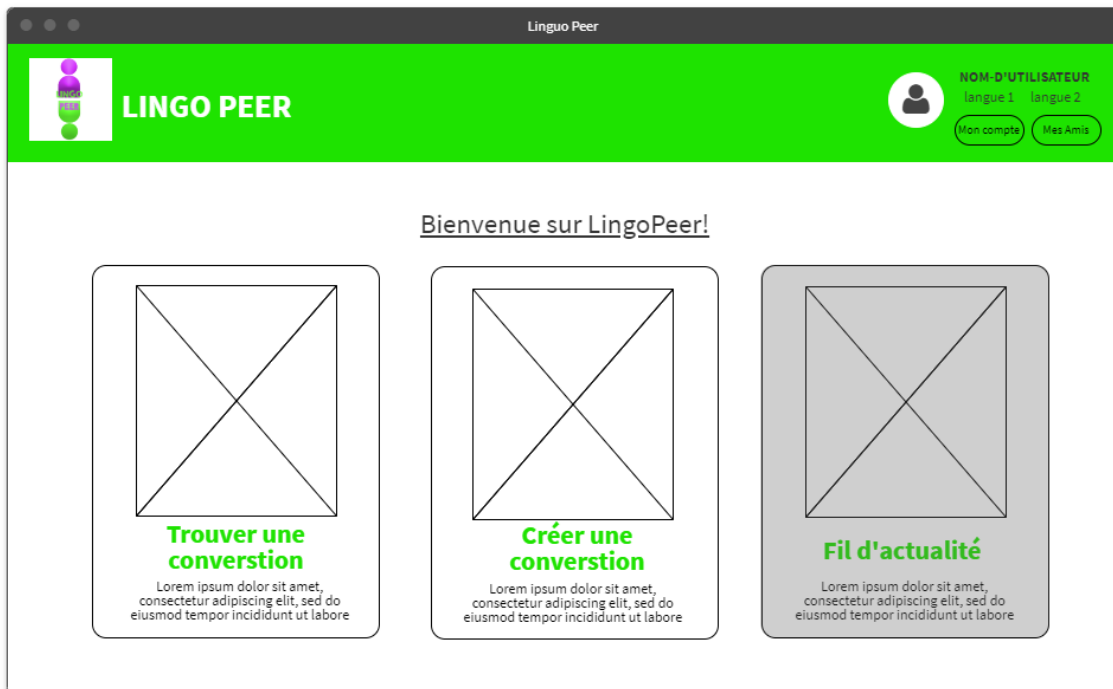


Figure 12

Les informations seront présentée de manière à ce que l'utilisateur passe le moins de temps possible à les lire comme le montre la seconde image réalisé à l'aide de wireframe il y a 3 élément un minimum de texte avec des icônes plutôt grosses.(fig.13)

Le site reste assez homogène en il suit une charte graphique ,par exemple l couleur verte reste inchangé ainsi que la disposition.(fig.13)

Le site s'adapte avec l'apport d'un thème sombre réglable au souhait de l'utilisateur en effet cette implémentation est de plus en plus apporté au sein même des OS .(fig.14)

Page d'accueil après connexion :

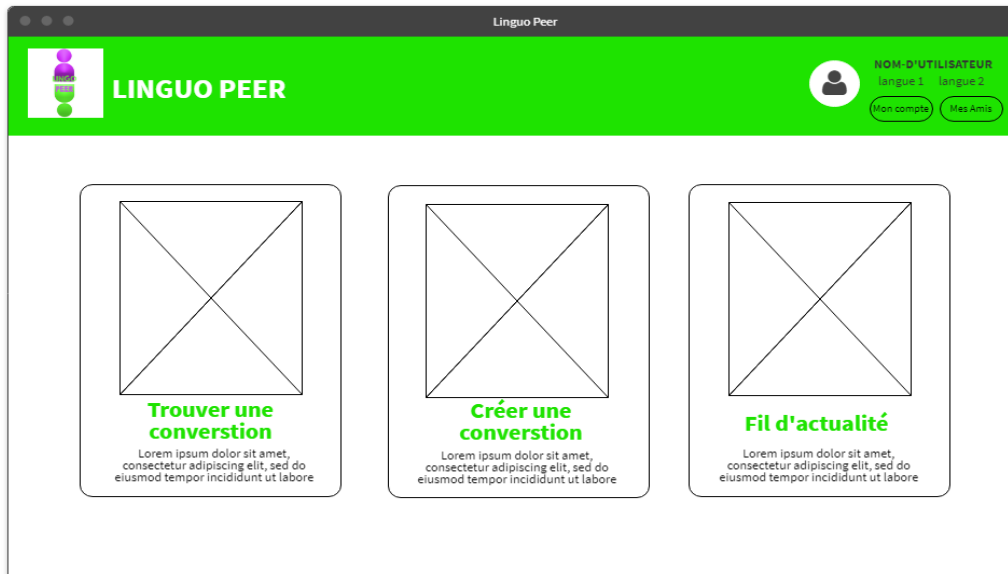


Figure 13

Page d'accueil après connexion (thème sombre):

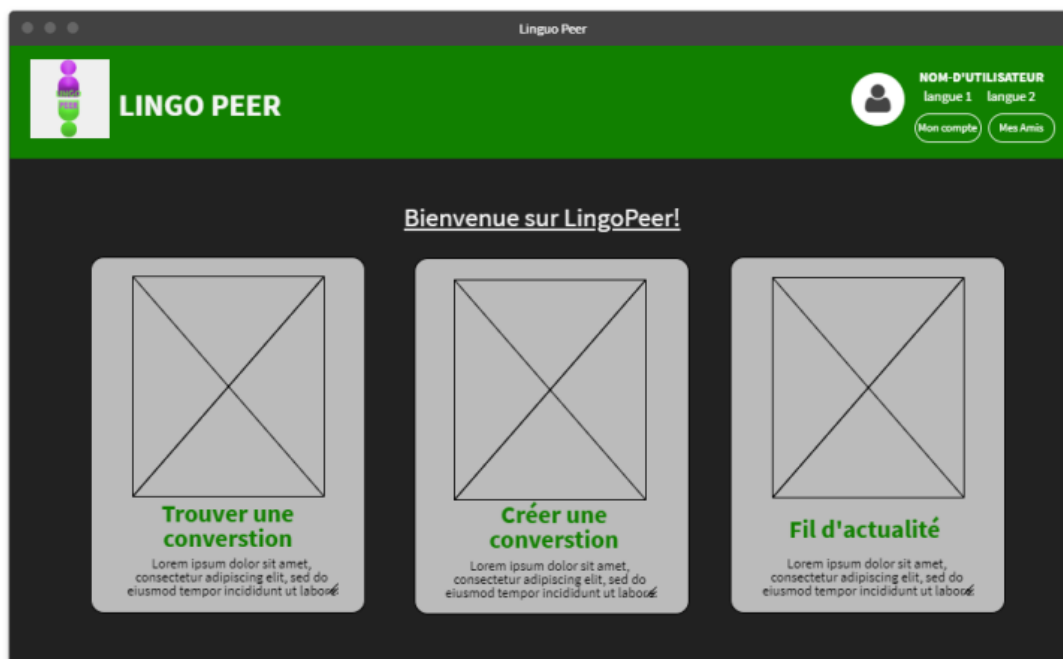


Figure 14

Le reste des différentes vues se situe en annexe .

3) Solutions

3.1) Présentation et analyse des solutions techniques

Dans cette partie, nous détaillerons l'utilisation de chaque outil dont nous nous servirons durant le projet.

Tout d'abord, nous avons choisis d'utiliser l'IDE Visual Studio Code car il est composé de beaucoup d'outils qui nous seront très utiles durant le projet, comme d'auto complétion assurée par IntelliSense, le débogage, le formatage, une navigation dans le code ainsi que la re factorisation et de nombreuses autres fonctionnalités. Visual Studio Code est également compatible avec Node.js, ce qui est intéressant dans le cadre de notre projet. Enfin, il est compatible avec Linux, Mac et Windows, ce qui lui permet d'être très flexible en matière d'utilisation dans les différents OS.

Notre projet est une web app, de ce fait, le choix du langage JavaScript était évident car il s'agit d'un langage :

- que nous allons apprendre facilement grâce à divers cours et de part sa difficulté relative.
- Très populaire. En effet, JavaScript était le langage le plus utilisé en 2019, de ce fait, nous allons facilement trouver de la documentation.
- essentiel, JavaScript un langage qui a été conçu pour réaliser des pages web dynamiques.
- qui possède de nombreux frameworks qui vont nous permettre de mettre en place différents aspects du projet.

Comme annoncé précédemment, nous utiliserons Node.js. Il s'agit d'une plateforme logicielle en JavaScript permettant d'utiliser ce langage côté serveur afin de concevoir des applications en réseau. Node.js est très similaire au JavaScript ce qui nous permettra de l'utiliser facilement, il est aussi très performant et très utilisé.

Notre application web sera composée d'une seule page qui sera modifiée en fonction des actions de l'utilisateur. De ce fait, nous utiliserons la bibliothèque React. React est conçu pour créer facilement des interfaces utilisateurs. Il met à jour les composants de la page qui doivent changer et ceux de façon optimale. Il peut aussi être utilisé côté serveur avec Node.

Afin de mettre en place un chat audio et peut-être vidéo, nous utiliserons l'API Web Real-Time Communication, abrégé en WebRTC. Il permettra de mettre en place des communications entre les utilisateurs, et ceux, en temps réel. WebRTC utilise le système pair à pair pour mettre en place ces communications, de ce fait, l'utilisateur n'a pas besoin d'installer de plugins ou de logiciels tiers pour communiquer.

Pour stocker les données des utilisateurs, nous avons choisis d'utiliser MongoDB, il s'agit d'un système de gestion de base de données écrit en C++, de ce fait, nous allons pouvoir le comprendre car nous étudions ce langage. Il permet des liaisons avec les principaux langages de programmation, notamment JavaScript. MongoDB est automatisé, sécurisé et fait régulièrement des sauvegardes, tous ces critères nous ont guidés dans ce choix. Il est aussi composé de beaucoup d'outils qui simplifient le travail des développeurs, il permet de visualiser des données grâce à des graphiques mis à jour en temps réel par exemple.

3.2) Gestion du travail

Afin de mener un projet dans les meilleures conditions ,il faut à tout prix une bonne dynamique de travail.

Il faut dans un premier temps,s'assurer que tout le monde est un rôle prédéfini,c'est donc le chef de projet qui a défini les rôles suivant :

GENTIL Driss	:	Gestion du peer to peer /back end
CLERC Louis	:	Gestion de la connexion vidéo /back end
HIGGINS Alec	:	Développeur UI /front end
DELHOMME Baptiste	:	Développeur UX /front end
KADIRI Saad	:	Gestion de base de donnée/back end
DAIKH Nassim	:	Interaction client serveur /back end

Cette répartition n'est pas anodine en effet elle permet de donner une dynamique de binôme ce choix a été mise en place en se basant sur le projet à réaliser en java en effet cette dynamique permet d'avoir une personne avec qui tacler un problème sans pour autant imputer à toute la progression du projet.

Mais il reste primordiale que chacun sache qui réalise quoi, c'est pourquoi avant de démarrer chaque session de travail collective ,il y a une réunion ou l'on échange sur ce que l'on à travailler .Cela permet de garder une bonne dynamique de groupe et assure une bonne communication .

Aussi dans l'objectif de savoir rapidement où en est l'avancement du projet nous avons décidé d'utiliser le logiciel trello qui est un diagramme de gantt interactif qui permet de voir l'avancement de chacun est d'attribuer des tâches spécifiques à un membre de l'équipe.

3.3) Description d'un modèle de développement

Le projet n'a pas seulement besoin d'une bonne gestion des charges de travail, il faut définir un modèle de développement en accord avec celui-ci .

Afin de rendre le projet plus digeste pour une équipe il est nécessaire de procéder à un découpage des de ce dernier afin d'en réduire la complexité .

Nous avons décidé d'adopter un découpage itératif car un effet toute l'équipe travaille à la réalisation de l'application et ce en simultanée il est donc important de d'utiliser un processus itératif. En clair ,on souhaite réaliser des prototypes fonctionnels de l'application et ajouter des fonctionnalités au fur et à mesure. Autrement dit, c'est une démarche Bottom-up car on va venir ajouter mais il se peut tout à fait que l'on repasse sur une démarche top down une fois toutes fonctionnalités implémentées.

Cette logique s'inscrit dans les méthodes agiles c'est-à-dire que la méthode est adaptative et prédictive.

Annexe:

Scénario d'utilisation, sur lequel s'est basé le BPMN:

Login:

L'utilisateur arrive sur la page d'accueil, il aura deux choix :

- Le bouton se connecter (en haut à droite) , cela le renverra sur une page où il va devoir rentrer son adresse mail et son mot de passe , si son compte existe il y aura un petit pop up lui disant qu'il est bien connecté ou non .

Si l'utilisateur n'a pas de compte il pourra cliquer sur "créer un compte", il devra remplir un formulaire lui demandant son nom, prénom, adresse mail, mot de passe, langue(s)

- Le bouton "rester déconnecter", il devra seulement entrer un prénom et des langues pour continuer.

Changer ses données personnelles:

L'utilisateur est sur la page d'accueil, il clique sur le bouton "Mon compte" (en haut à droite à côté de sa photo de profil). Il se retrouve sur la page de son profil. Il peut alors:

- Changer son niveau de langue : Il clique sur un menu déroulant qui lui propose les choix suivants: "débutant", "intermédiaire", "expérimenté", "expert". Après avoir choisi celui qui lui convient, le menu se ferme et son niveau de langue est changé.
- Ajouter une langue: Dans la partie "Mes Langues", il clique sur le bouton "Ajouter une langue". Une page pop-up sur son écran, sur cette page il y a une liste de langues ("anglais", "français", "espagnol", "italien", "Russe"), cette liste de langues ne comprend pas les langues déjà liées à son compte. L'utilisateur choisit alors la langue qu'il veut ajouter, ensuite la page pop-up disparaît et la langue choisie est ajoutée.
- Enlever une langue: Dans la partie "Mes Langues", il clique sur le bouton "Enlever une langue". Une page pop-up sur son écran, sur cette page il y a une liste de langues ("anglais", "français", "espagnol", "italien", "Russe"), cette liste de langues ne comprend pas les langues déjà liées à son compte. L'utilisateur choisit alors la langue qu'il veut enlever, ensuite la page pop-up disparaît et la langue choisie est enlevée.
- Ajouter/Changer sa photo de profil: Dans la partie "Ma photo", il clique sur le bouton "Changer ma photo de profil". Il choisit ensuite une photo sur sa machine (formats pdf, jpg, jpeg uniquement). Si il n'y a pas d'erreurs, le message "Changement réussi" apparaît sous le bouton "Changer ma photo de profil". Si il y a eu une erreur, le message correspondant apparaît en rouge sous le bouton "Changer ma photo de profil".

Trouver une conversation:

L'utilisateur est sur la page d'accueil, il y a trois gros boutons rectangulaire. Il clique sur le bouton : "trouver une conversation". Il arrive sur la page de paramétrage, il peut alors:

- Choisir la langue sur une liste comprenant "anglais", "français", "espagnol", "italien", "allemand".
- Choisir le thème parmi une liste (...)
- Faire tourner la roue en cliquant sur une icône d'une roue, cela lancera une petite animation, le prénom d'un utilisateur en ressortira .
- Lancer la conversation.

Créer une conversation avec un ami:

L'utilisateur est sur la page d'accueil, il y a trois gros boutons rectangulaires. Il clique sur le bouton : "Créer une conversation avec un ami". Il arrive sur la page de paramétrage, il peut alors:

- Choisir un ami de sa liste avec qui avoir une conversation
- Choisir la langue
- Choisir le thème

Son ami recevra une notification lui indiquant le nom de la personne voulant avoir une conversation avec lui, si oui ou non il veut rejoindre celle-ci, la langue sera indiqué, et le thème si il a été choisi.

- Lancer la conversation, seulement si il y a bien deux personnes.

Aller sur le fil de partage

(dépendra du temps qu'il nous reste : pas sur de le faire)

Gérer ses amis:

Envoyer une demande avec nom d'utilisateur:

L'utilisateur est sur la page d'accueil, il clique sur le bouton "Mes Amis" (en haut à droite à côté de "Mon compte". Il arrive sur la page d'amis, il y a la liste de ses amis au milieu de la page. A côté de la liste il y a un input de texte "Email", un input texte "Message" et un bouton "envoyer la demande". Il devra remplir les champs précédents (le message est optionnel) puis appuyer sur le bouton "envoyer la demande". Si il n'y a pas d'erreurs, le message "demande envoyée" apparaît sous le bouton, en cas d'erreur le message associé à celle-ci apparaît sous le bouton.

Demander en amis durant une conversation:

L'utilisateur est sur une conversation, si il le souhaite il peut envoyer une demande d'amis directement de cette page. Pour cela il clique sur le bouton "ajouter en ami", cela envoie automatiquement une demande à son interlocuteur.

Envoyer un message à un ami:

L'utilisateur est sur la page "Mes Amis", il a alors accès à sa liste d'amis. Il peut cliquer sur le bouton "Message" devant le nom de l'ami à qui il veut envoyer un message. Il arrive alors sur la page de message, il y a le nom de l'ami, sa photo de profil, un input de texte "Message" et un bouton "Envoyer". Il remplit alors le champ "Message" et clique sur le bouton "Envoyer". Si il n'y a pas d'erreurs, le message "Message envoyé" apparaît sous le bouton, en cas d'erreur le message associé à celle-ci apparaît sous le bouton.


Fonctionnalités de la page de conversation:

Page de chat écrit uniquement:

Lorsque l'utilisateur est sur cette page, il y a un chat au milieu de la page. Il peut envoyer et lire des messages en direct avec son interlocuteur. Il peut aussi envoyer une demande d'ami.

Page de chat audio/vidéo:

Lorsque l'utilisateur est sur cette page, il y a le retour vidéo au milieu gauche de son écran, et le chat écrit est au milieu gauche. Il a le choix de converser en chat audio ou en chat écrit. En bas de la page, il peut Activer/désactiver sa webcam, activer/désactiver son micro, mais aussi ajouter en ami son interlocuteur.



LINGO PEER

Connecte toi à LingoPeer

Se connecter

Adresse Email:

Mot de Passe:


Continuer sans se connecter

Mon Nom :

Langue 1 Langue 2

Figure 15

Page de création de compte :



LINGO PEER

Rejoins LingoPeer!

Données personnelles

Prénom:

Nom:

Adresse Email:

Mot de Passe:

Vérification Mot de passe:

Langue 1 Langue 2

Figure 16

Page de profil :

Linguo Peer

LINGO PEER

MON PROFILE

Mes données
Prénom:
Nom:
Adresse Email:
Modifier mes données

Mes langues
Langue 1: Langue 2:
Niveau de langue: Select
Ajouter une langue
Supprimer une langue

Ma photo
Changer ma photo de profil

Figure 17
Page d'amis :

Linguo Peer

LINGO PEER

Mes Amis

Friend n°1
Friend n°2
Friend n°3
Friend n°4

Ajouter un amis
Email:
Message
Envoyer la demande

Figure 18

Trouver une conversation :

The screenshot shows the 'Linguo Peer' web application. The header is green with the 'LINGO PEER' logo on the left and a user profile section on the right. The profile section includes a user icon, the text 'NOM-D'UTILISATEUR', and two language options: 'langue 1' and 'langue 2'. Below these are two buttons: 'Mon compte' and 'Mes Amis'. The main content area is titled 'Paramétrage de ta conversation'. It contains two side-by-side boxes. The left box is titled 'Choisis la langue !' and has a label 'langue :' followed by a dropdown menu with 'Select' and a downward arrow. The right box is titled 'Choisis un thème !' and has a label 'Thème :' followed by a dropdown menu with 'Select' and a downward arrow. Below these two boxes is a single button labeled 'Lancer la conversation'.

Figure 19
Créer une conversation :

The screenshot shows the 'Linguo Peer' web application. The header is green with the 'LINGO PEER' logo on the left and a user profile section on the right. The profile section includes a user icon, the text 'NOM-D'UTILISATEUR', and two language options: 'langue 1' and 'langue 2'. Below these are two buttons: 'Mon compte' and 'Mes Amis'. The main content area is titled 'Conversation avec un amis'. It contains three boxes. The left box is titled 'Choisis la langue !' and has a label 'langue :' followed by a dropdown menu with 'Select' and a downward arrow. The right box is titled 'Choisis un thème !' and has a label 'Thème :' followed by a dropdown menu with 'Select' and a downward arrow. Below these two boxes is a third box titled 'Invite ton ami!'. This box has a label 'Amis :' followed by a dropdown menu with 'Select' and a downward arrow. Below the 'Invite ton ami!' box is a single button labeled 'Lancer la conversation'.

Figure 20

Chat écrit :

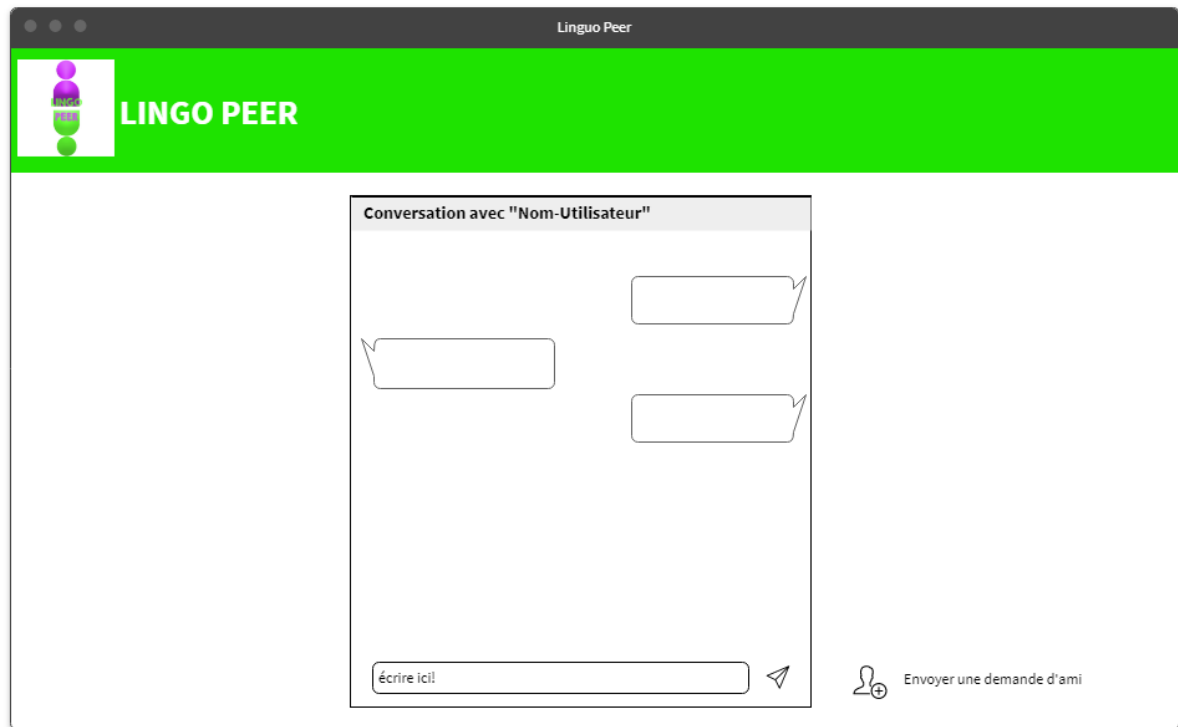


Figure 21

Chat video, oral et écrit :

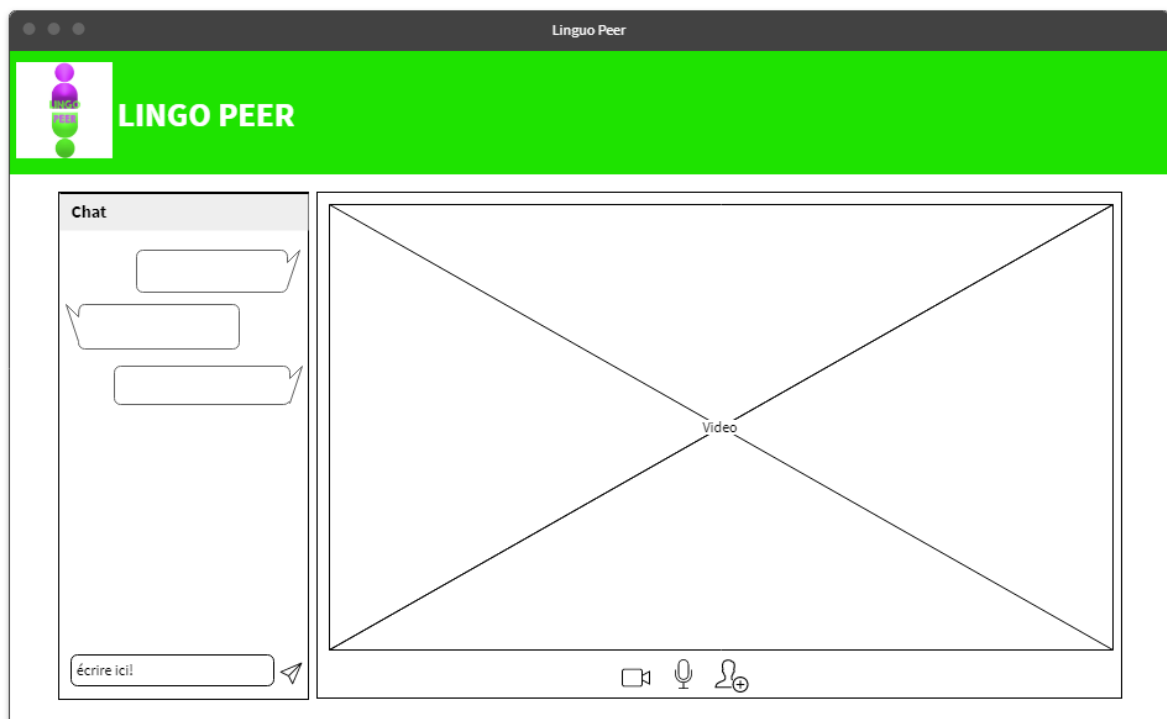


Figure 22

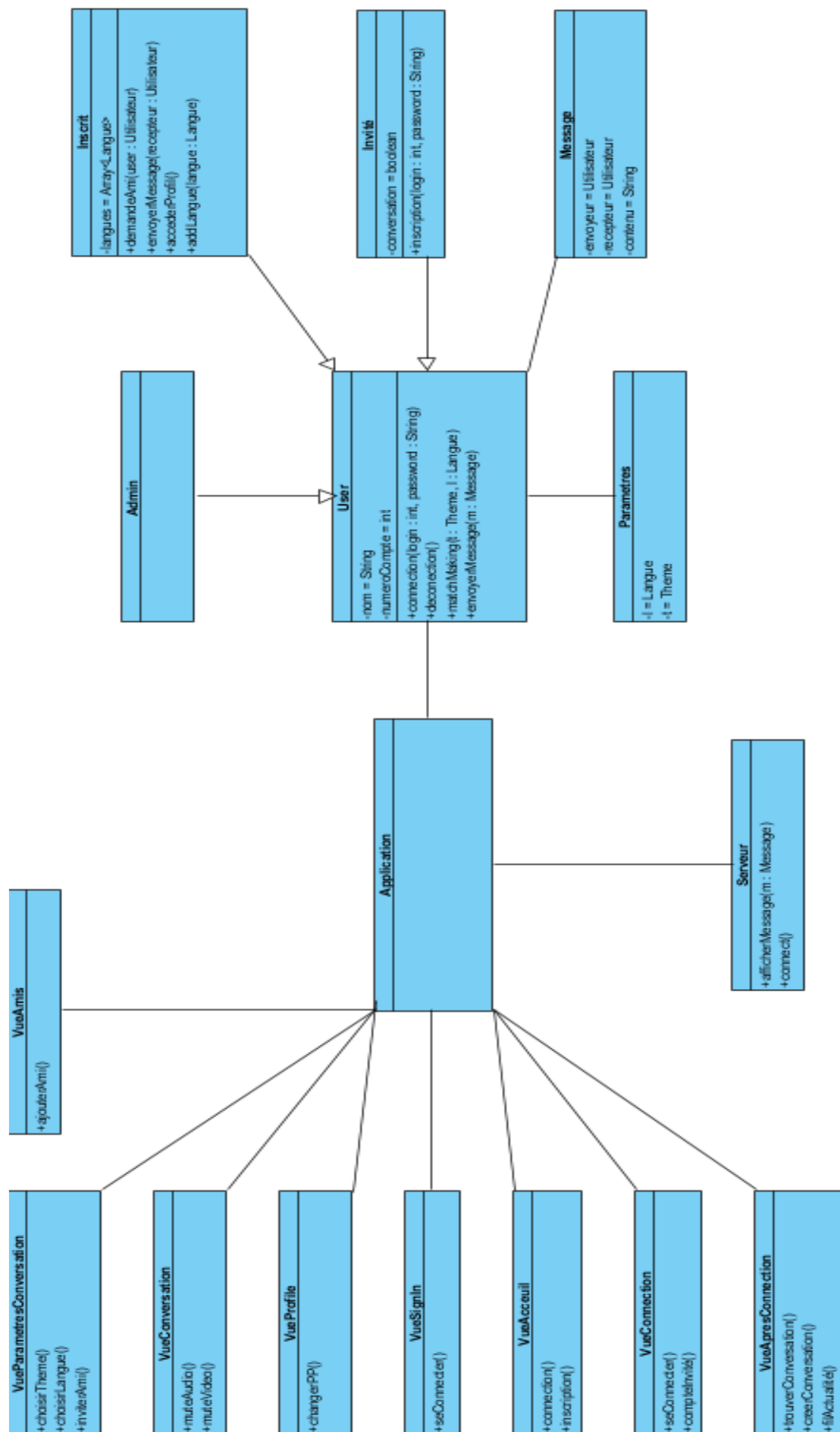


Figure 23

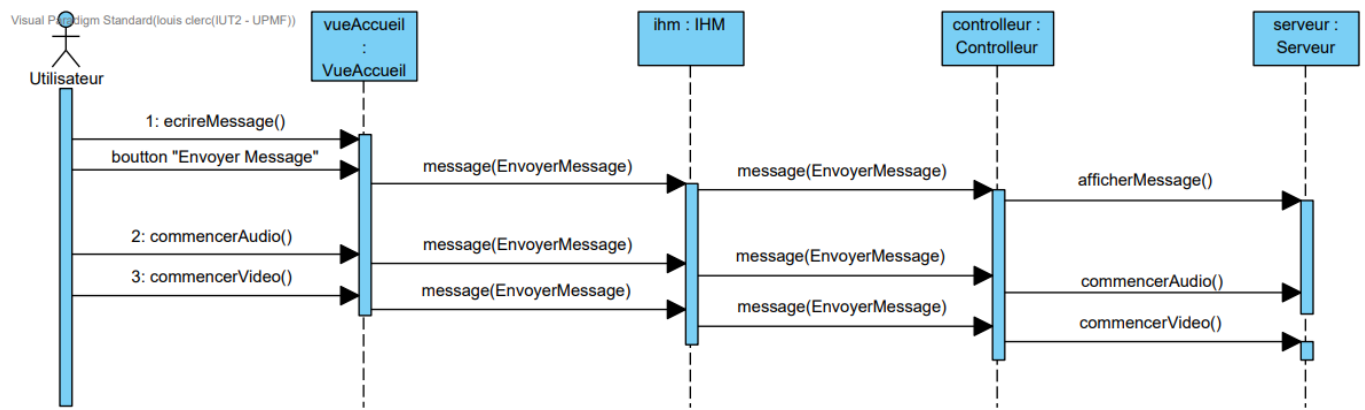


Figure 24