Une image contenant texte, conception, poisson

Le contenu généré par l’IA peut être incorrect.

Table des matières

[I – Présentation du groupe 4](#_Toc197644746)

[II – Contexte 4](#_Toc197644747)

[III – Livrables 4](#_Toc197644748)

[IV – Réalisation 4](#_Toc197644749)

[V – Conclusion 22](#_Toc197644750)

[VI – Annexes 22](#_Toc197644751)

Table des illustrations

[Figure 1 - Diagramme Use Case du projet EasySave 6](#_Toc197644759)

[Figure 2 - Diagramme de séquences général du logiciel 7](#_Toc197644760)

[Figure 3 - Diagramme de séquences de "Créer un travail" 8](#_Toc197644761)

[Figure 4 - Diagramme de séquences de "Lister tous les travaux créés" 9](#_Toc197644762)

[Figure 5 - Diagramme de séquences de "Supprimer un travail" 10](#_Toc197644763)

[Figure 6 - Diagramme de séquences de "Modifier un travail" 11](#_Toc197644764)

[Figure 7 - Diagramme de séquences de "Lancer un travail" 12](#_Toc197644765)

[Figure 8 - Diagramme de séquences de "Changer la langue" 13](#_Toc197644766)

[Figure 9 - Diagramme d'activité général du logiciel 14](#_Toc197644767)

[Figure 10 - Diagramme d'activité de "Créer un travail" 15](#_Toc197644768)

[Figure 11 - Diagramme d'activité de "Lister tous les travaux créés" 16](#_Toc197644769)

[Figure 12 - Diagramme d'activité de "Supprimer un travail" 17](#_Toc197644770)

[Figure 13 - Diagramme d'activité de "Modifier un travail" 18](#_Toc197644771)

[Figure 14 - Diagramme d'activité de "Lancer un travail" 19](#_Toc197644772)

[Figure 15 - Diagramme d'activité de "Changer la langue" 20](#_Toc197644773)

[Figure 16 - Diagramme de classe du projet 21](#_Toc197644774)

# I – Présentation du groupe

Notre groupe de projet est composé de quatre membres :

* Isrâ BOUDEMAGH
* Clément EVANGELISTI
* Elrazi KOULA
* Alexis PEIFFER

# II – Contexte

Dans le cadre de notre intégration à ProSoft, nous allons concevoir EasySave, un programme de sauvegarde conçu spécialement pour les entreprises. L'objectif ici est de proposer une solution sécurisée, facile à configurer et fiable pour la sauvegarde des données sensibles.

EasySave devra incorporer des fonctionnalités telles que la possibilité de réaliser des sauvegardes totales ou différentielles, une gestion des journaux ou bien encore une interface conviviale.

# III – Livrables

Les différents livrables à réaliser pour ce projet sont :

- Une version 1.0 du logiciel EasySave avec les diagrammes UML qui lui correspondent

- Une version 2.0 et 1.1 du logiciel toujours avec les diagrammes qui le font évoluer

- Une version 3.0 de EasySave avec les diagrammes UML qui le font évoluer

# IV – Réalisation

Pour ce livrable, il fallait coder la première version du logiciel EasySave. Nonobstant, pour y parvenir, on avait besoin de réaliser quatre types de diagrammes :

* Un diagramme Use Case
* Plusieurs diagrammes de séquences
* Plusieurs diagrammes d’activité
* Un diagramme de classes

Pour réaliser nos diagrammes, nous avons imaginé un prototype de notre première version console. Celle-ci aurait un menu principal avec un choix par clavier numérique avec sept numéros attribués à sept fonctionnalités :

* Créer un travail
* Lister tous les travaux créés
* Supprimer un travail
* Modifier un travail
* Lancer un travail
* Changer la langue de l’application
* Quitter l’application

Pour la création de travail, on demandera à l’utilisateur de remplir à la suite le nom de son travail de sauvegarde, son répertoire source, son répertoire cible et son type. On listera ensuite les quatre entrées avant de demander confirmation.

Pour la liste des travaux, on affichera simplement tous les travaux qui ont pu être créés. Pour la suppression, on permettra à l’utilisateur de choisir un travail à supprimer avec le nom du travail à entrer puis on lui affichera celui-ci avant de demander confirmation pour la suppression.

Pour la modification, on fera comme la suppression : on affichera tous les travaux puis l’utilisateur en choisira un à modifier. Ensuite, l’utilisateur choisira un élément à modifier sur son travail puis on lui demandera confirmation pour l’élément à modifier.

Pour le lancement, on affichera tous les travaux avec leurs différentes caractéristiques. L’utilisateur choisira un travail qui sera inactif puis on l’affichera pour lui demander confirmation pour le lancement.

Au niveau du changement de langue, on affichera dans un premier temps la langue actuelle de l’application, l’anglais étant la langue par défaut, puis dans un second temps on le laissera choisir une langue parmi celles proposées. Si la langue est déjà sélectionnée, rien ne changera et on en informera l’utilisateur.

Le diagramme de cas d’utilisation (Use Case) permet de décrire les interactions entre les acteurs (utilisateurs ou systèmes externes) et le système étudié. Cela permet de visualiser les fonctionnalités du système du point de vue de l’utilisateur. Ici, l’acteur principal sera l’utilisateur du logiciel tandis que ce dernier est le système étudié.

Ci-dessous en Figure 1 le diagramme Use Case du projet :

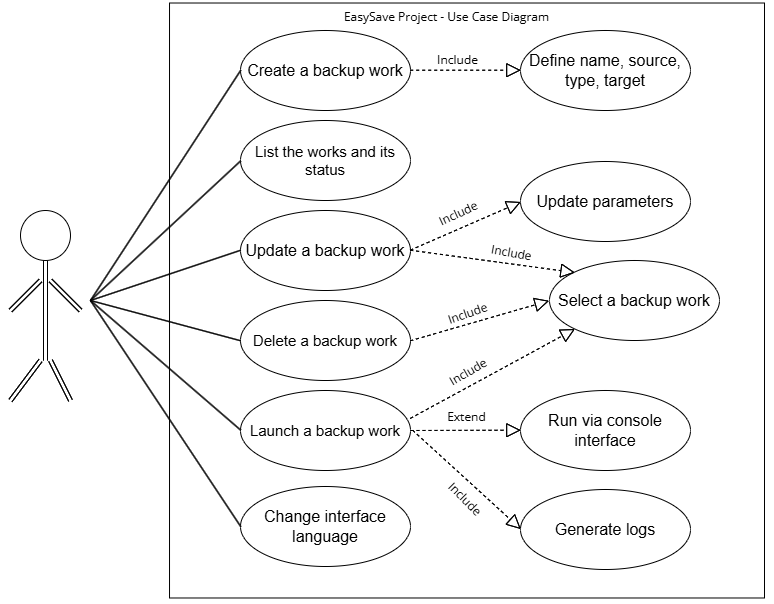


Figure - Diagramme Use Case du projet EasySave

Les diagrammes de séquences montrent l’interaction chronologique entre différents objets ou composants au sein d’un scénario particulier de cas d’utilisation, illustrant précisément les messages échangés et l’ordre temporel de ces échanges. Dans le cadre de notre projet, nous allons avoir sept diagrammes de séquences : un pour le général et un pour chaque fonctionnalité que nous avons présentée.

Voici en Figure 2 le diagramme de séquences qui présente le logiciel de manière générale :

Une image contenant texte, capture d’écran, diagramme, ligne

Le contenu généré par l’IA peut être incorrect.

Figure - Diagramme de séquences général du logiciel

Voici en Figure 3 le diagramme de séquences qui présente la fonctionnalité « Créer un travail » :

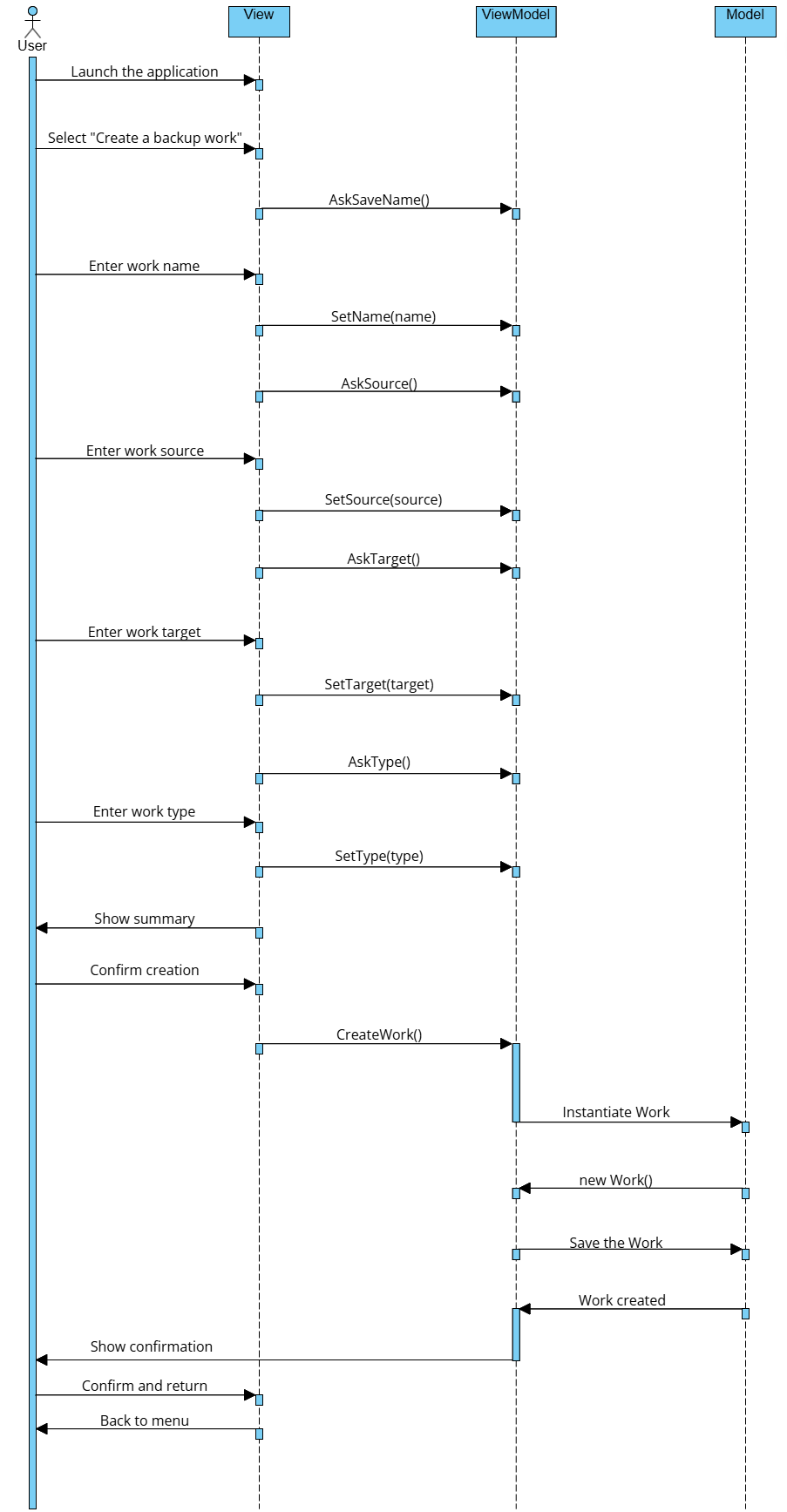


Figure - Diagramme de séquences de "Créer un travail"

Voici en Figure 4 le diagramme de séquences qui présente la fonctionnalité « Lister tous les travaux créés » :

Une image contenant texte, capture d’écran, Parallèle, diagramme

Le contenu généré par l’IA peut être incorrect.

Figure - Diagramme de séquences de "Lister tous les travaux créés"

Voici en Figure 5 le diagramme de séquences qui présente la fonctionnalité « Supprimer un travail » :

Une image contenant texte, capture d’écran, Parallèle, nombre

Le contenu généré par l’IA peut être incorrect.

Figure - Diagramme de séquences de "Supprimer un travail"

Voici en Figure 6 le diagramme de séquences qui présente la fonctionnalité « Modifier un travail » :

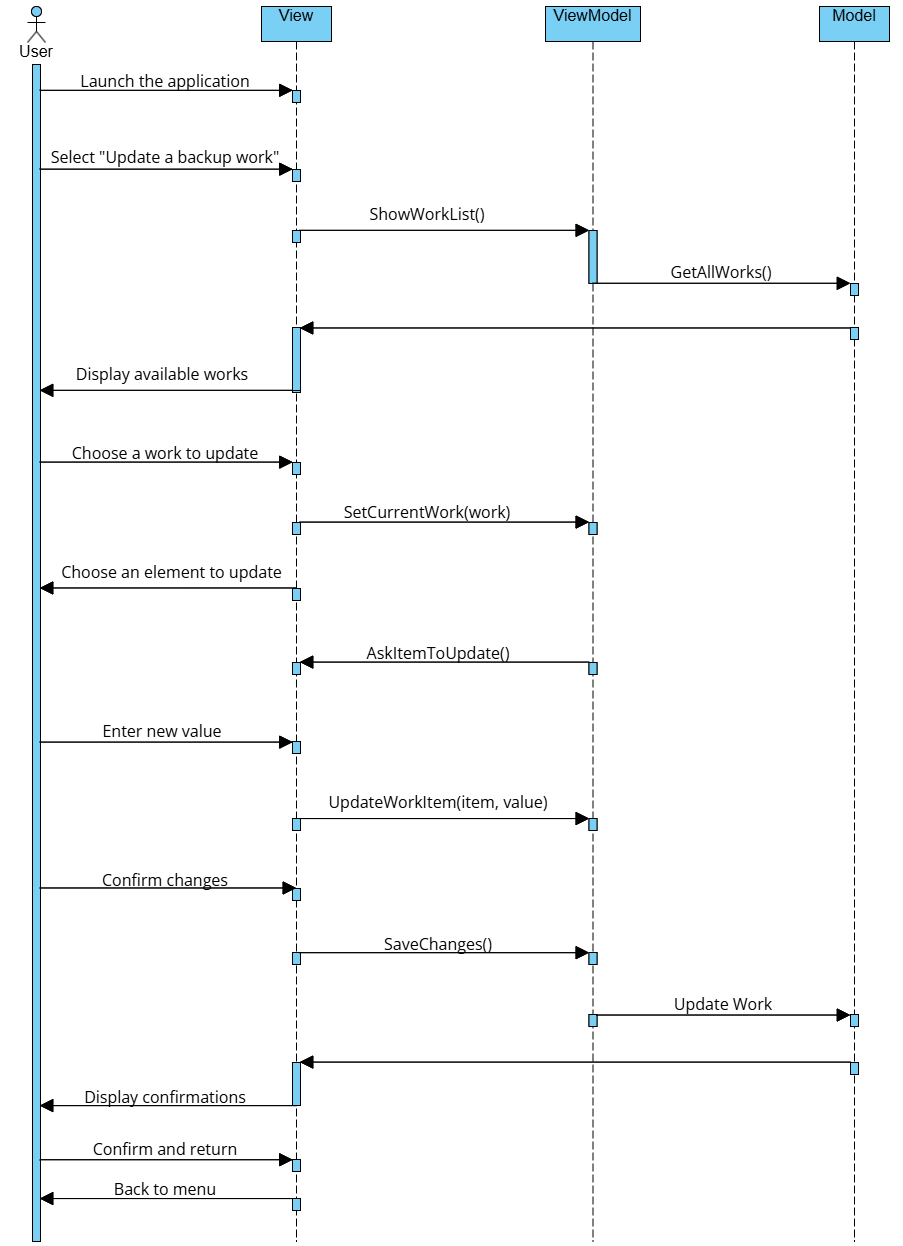


Figure - Diagramme de séquences de "Modifier un travail"

Voici en Figure 7 le diagramme de séquences qui présente la fonctionnalité « Lancer un travail » :

Une image contenant texte, capture d’écran, Parallèle, diagramme

Le contenu généré par l’IA peut être incorrect.

Figure - Diagramme de séquences de "Lancer un travail"

Voici en Figure 8 le diagramme de séquences qui présente la fonctionnalité « Changer la langue » :

Une image contenant texte, capture d’écran, Parallèle, nombre

Le contenu généré par l’IA peut être incorrect.

Figure - Diagramme de séquences de "Changer la langue"

Les diagrammes d’activité illustrent graphiquement le flux de contrôle d’activités ou d’opérations dans un processus ou scénario donné, mettant en évidence les enchaînements, les décisions et les embranchements possibles.

Voici en Figure 9 le diagramme d’activité qui présente le logiciel de manière générale :

Une image contenant diagramme, capture d’écran, ligne, conception

Le contenu généré par l’IA peut être incorrect.

Figure - Diagramme d'activité général du logiciel

Voici en Figure 10 le diagramme d’activité qui présente la fonctionnalité « Créer un travail » :

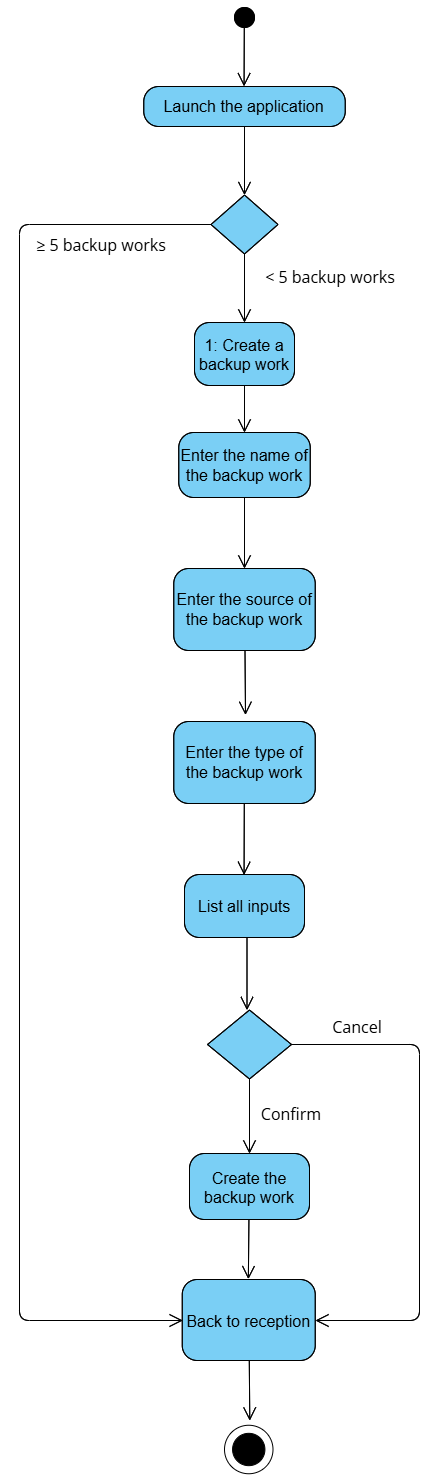


Figure - Diagramme d'activité de "Créer un travail"

Voici en Figure 11 le diagramme d’activité qui présente la fonctionnalité « Lister tous les travaux créés » :

Une image contenant texte, capture d’écran, diagramme, ligne

Le contenu généré par l’IA peut être incorrect.

Figure - Diagramme d'activité de "Lister tous les travaux créés"

Voici en Figure 12 le diagramme d’activité qui présente la fonctionnalité « Supprimer un travail » :

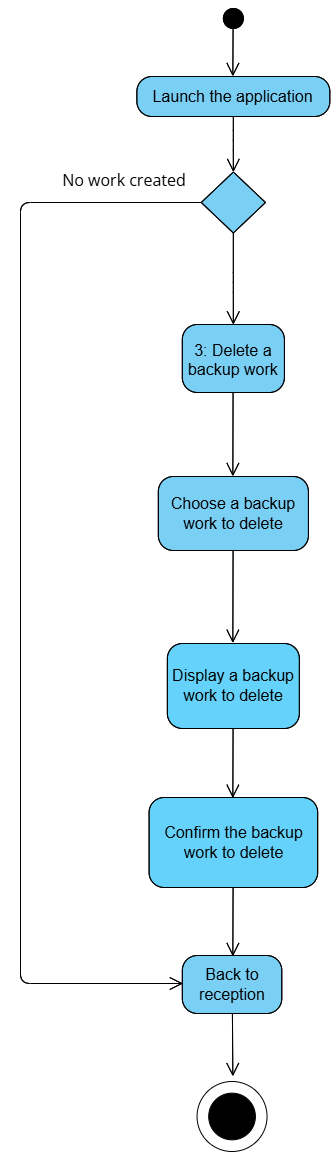


Figure - Diagramme d'activité de "Supprimer un travail"

Voici en Figure 13 le diagramme d’activité qui présente la fonctionnalité « Modifier un travail » :

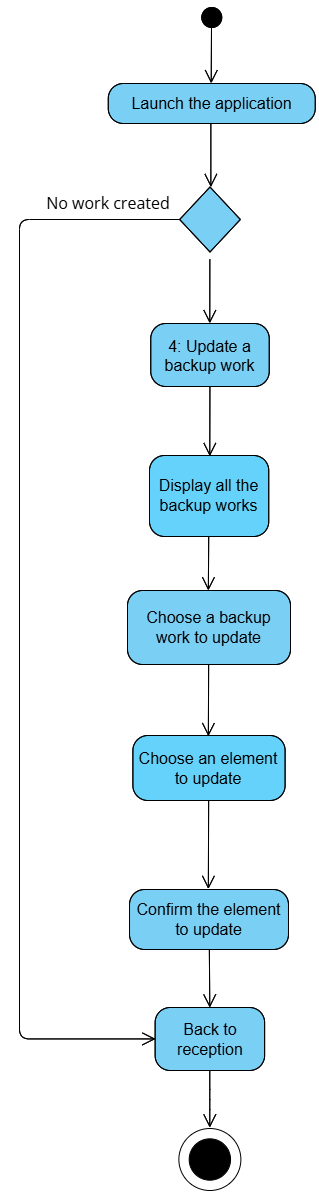


Figure - Diagramme d'activité de "Modifier un travail"

Voici en Figure 14 le diagramme d’activité qui présente la fonctionnalité « Lancer un travail » :

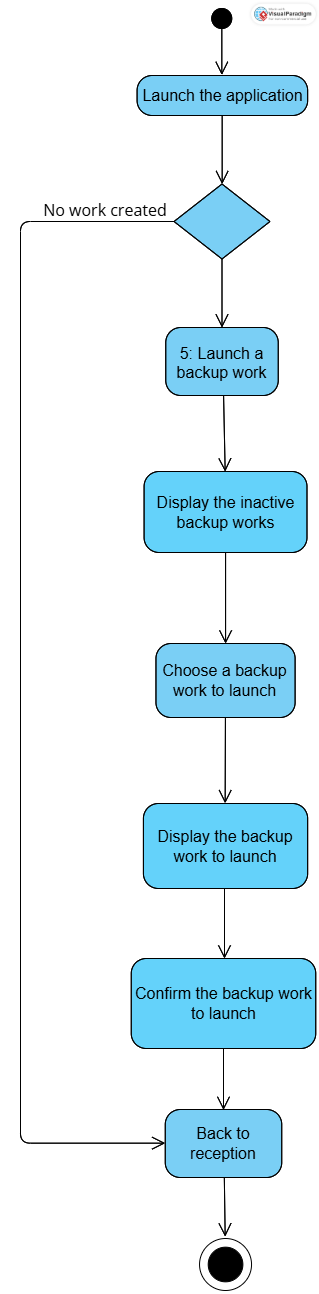


Figure - Diagramme d'activité de "Lancer un travail"

Voici en Figure 15 le diagramme d’activité qui présente la fonctionnalité « Changer la langue » :

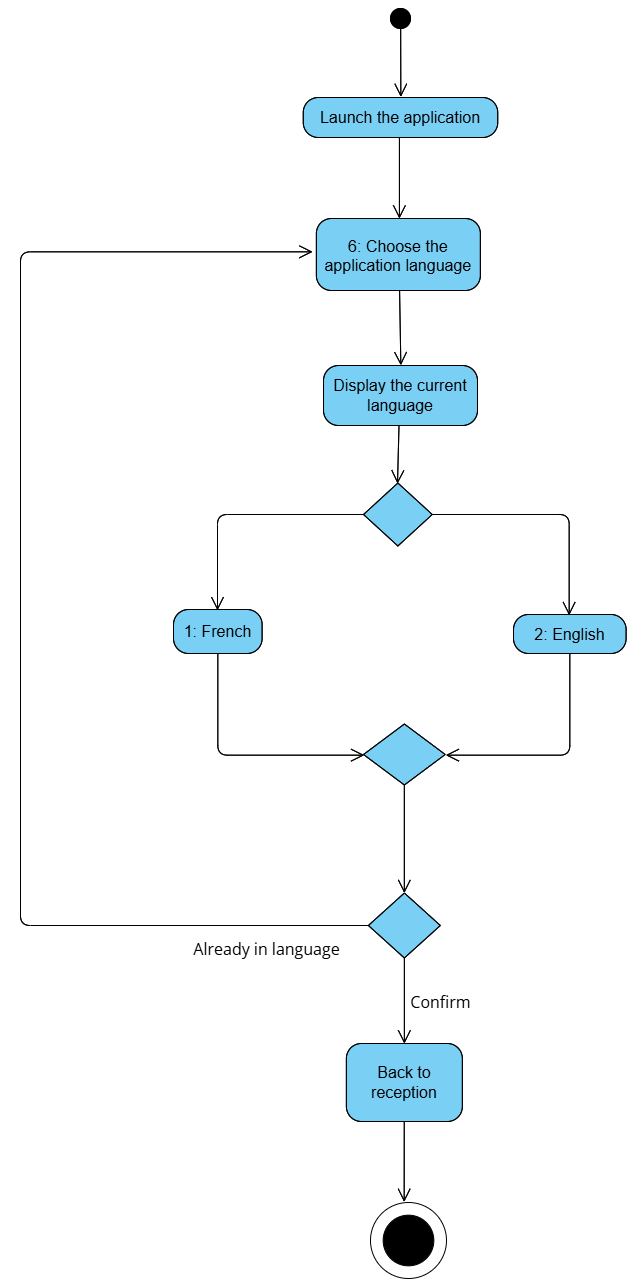


Figure - Diagramme d'activité de "Changer la langue"

Le diagramme de classe présente la structure statique du système en montrant les différentes classes, leurs attributs, leurs méthodes, ainsi que leurs relations (héritage, associations, dépendances, etc.) entre ces classes.

Ci-dessous en Figure 16 le diagramme de classes du projet :

Une image contenant texte, capture d’écran, Parallèle, Police

Le contenu généré par l’IA peut être incorrect.

Figure - Diagramme de classe du projet

# V – Conclusion

Dans ce premier livrable, on a pu créer une première version de notre logiciel de sauvegarde, avec des diagrammes Use Case, de séquences, d’activité et de classes.

Le code est disponible sur notre GitHub qui est présent dans la partie « Annexes ».

# VI – Annexes

Sujet : [*https://moodle.cesi.fr/pluginfile.php/160192/mod\_resource/content/3/co/Projet.html*](https://moodle.cesi.fr/pluginfile.php/160192/mod_resource/content/3/co/Projet.html)

Logiciel « Visual Paradigm Online » pour la réalisation des diagrammes : [*https://online.visual-paradigm.com*](https://online.visual-paradigm.com)

GitHub du projet : [*https://github.com/alexisP3011/EasySave-Group-1*](https://github.com/alexisP3011/EasySave-Group-1)