

A thick dark blue vertical bar runs along the left edge of the page. A blue arrow-shaped banner points to the right, containing the date. In the bottom left corner, there are several thin, curved, overlapping lines in shades of blue and grey.

01/02/2021

Location de matériel

Gestion de Stocks

Paola Costa

CENTRE PROFESSIONNEL DU NORD VAUDOIS

Table des matières

1	Analyse préliminaire	3
1.1	Introduction	3
1.2	Objectifs.....	3
1.3	Planification initiale	4
2	Analyse / Conception.....	5
2.1	Concept	5
2.1.1	Fonctionnalités.....	5
2.1.2	Modèles de données	6
2.1.2.1	Modèle de données conceptuel.....	6
2.1.2.2	Modèle de données logique.....	7
2.2	Maquettes.....	8
2.3	Use Cases & Scénarii.....	9
2.3.1	Use Cases	9
2.3.2	Scénarii.....	10
2.3.2.1	Ajouter un objet.....	10
2.3.2.2	Afficher les objets dont le délai est dépassé	10
2.3.2.3	Générer un PDF	10
2.3.2.4	Louer un objet.....	11
2.3.2.5	Retourner un objet loué	12
2.3.2.6	Rechercher par mot-clé	12
2.4	Diagrammes	13
2.4.1	Diagramme de classe	13
2.4.2	Diagramme de flux.....	14
2.5	Stratégie de test.....	15
2.6	Risques techniques	15
2.7	Planification définitive	16
2.8	Infrastructure	17
2.8.1	Matériel hardware et système d'exploitation	17
2.8.2	Outils logiciels.....	17
3	Réalisation.....	18
3.1	Problèmes rencontrés et résolution	18
3.2	Dossier de réalisation	18
3.2.1	Répertoires et fichiers du projet.....	18
3.2.2	Produit fini.....	19
3.2.3	Programmation et scripts.....	19
3.3	Description des tests effectués	19
3.4	Erreurs restantes	19
3.5	Liste des documents fournis	19
4	Conclusions	20
5	Annexes.....	21
5.1	Résumé du rapport du TPI / version succincte de la documentation.....	21
5.2	Sources – Bibliographie.....	21

5.2.1	Livres utilisés	21
5.2.2	Pages internet consultées	21
5.2.3	Personnes consultées	21
5.3	Journal de travail	22
5.4	Manuel d'Installation	24
5.5	Manuel d'Utilisation.....	24
5.6	Archives du projet.....	24
5.7	Table des illustrations	24

1 Analyse préliminaire

1.1 Introduction

Avec les avancées actuelles de la technologie, il n'a jamais été aussi simple de gérer des stocks d'une entreprise. Cependant, tout le monde n'est pas encore équipé pareillement. Dans le but de préparer mon TPI, je vais donc réaliser une application de gestion de stocks pour mon pré-TPI. Celle-ci est créée dans le cadre d'une PME, spécialisée dans la location de matériel d'extérieur. Elle permettra à une personne, même novice en informatique, d'insérer ou sortir des pièces du stock (selon les arrivages ou les locations), de créer un PDF de l'inventaire d'un simple clic ainsi que de rechercher de plusieurs manières dans les données. L'intégralité des données propres à l'application seront stockées dans une base de données.

Ce projet servira tout d'abord à reprendre en main des technologies qui ont été abordées précédemment dans différents modules. Il s'agit ici du C# et du MySQL. Les mêmes technologies serviront à réaliser le TPI. Ensuite, le fait de compléter un projet individuel dans des conditions semblables à celles du TPI permet également de se projeter et de se rendre compte de comment le projet final (TPI) va se dérouler.

Enfin, aucun travail n'a été effectué eu préalable pour préparer le pré-TPI. L'intégralité du développement se fera lors du module et du temps mis à disposition.

1.2 Objectifs

Afin de mener à bien ce projet, de nombreux objectifs sont à compléter. La validation de ceux-ci permettra de déterminer le degré de complétion du projet. L'élément principal de ce projet consiste à créer une application « clé en main ». Cela signifie que l'application sera fonctionnelle sans investissements ultérieurs. De plus, l'application doit être accessible à des novices en informatique. Un mode d'emploi sera donc mis à disposition, afin de faciliter l'utilisation.

De plus, l'application contiendra plusieurs fonctionnalités. Il s'agit de celles qui sont citées ci-dessous :

- L'application devra permettre d'ajouter des objets aux stocks.
- L'application devra permettre de louer et retourner une location.
- Il sera possible d'effectuer une recherche sur l'application.
- Les données de l'application seront stockées dans une base de données distantes.
- L'application permettra de générer des rapports au format PDF, afin de réaliser des inventaires.
- L'application permettra d'effectuer un tri dans les données.
- L'application devra permettre d'afficher l'historique d'une pièce spécifique.

Enfin, afin de faciliter l'accessibilité à l'application, l'application sera accompagnée d'une procédure d'installation et de mise en service.

[illegible]

2 Analyse / Conception

2.1 Concept

2.1.1 Fonctionnalités

Afin de mener à bien ce projet, de nombreuses fonctionnalités doivent être mises en place. L'élément principal de ce projet consiste à créer une application « clé en main ». Cela signifie que l'application sera fonctionnelle sans investissements ultérieurs. De plus, l'application doit être accessible à des novices en informatique. Un mode d'emploi sera donc mis à disposition, afin de faciliter l'utilisation.

L'application devra également répondre à plusieurs points techniques spécifiques. Ce sont les suivants :

- Les données internes à l'application sont stockées dans une base de données.
- Au lancement de l'application, une connexion à une base de données distante est établie.
- Chaque pièce présente dans le stock est unique. Cela s'exprime grâce à son ID, qui est un code numérique unique.
- Il est possible d'ajouter une pièce dans le stock grâce à son ID. Celui-ci peut être déjà existant ou pas. S'il est nouveau, les informations propres à la pièce sont requises avant l'ajout.
- Il est possible de sortir une pièce du stock grâce à son ID. Il faut la durée de la location et les informations sur le loueur pour que cela s'effectue.
- Il est possible de générer deux types de rapports PDF différents, afin de réaliser un inventaire. Le premier regroupe le contenu actuellement en stock, le deuxième le matériel loué.
- Une fonction de recherche par mot-clé est disponible sur l'application.
- Une fonction de tri permet de voir les pièces dont la date de retour de location est dépassée.
- Il est possible de consulter l'historique de location d'une pièce.

Enfin, afin de rendre l'application accessible, une procédure d'installation et de mise en service seront également fournies.

2.1.2 Modèles de données

2.1.2.1 Modèle de données conceptuel

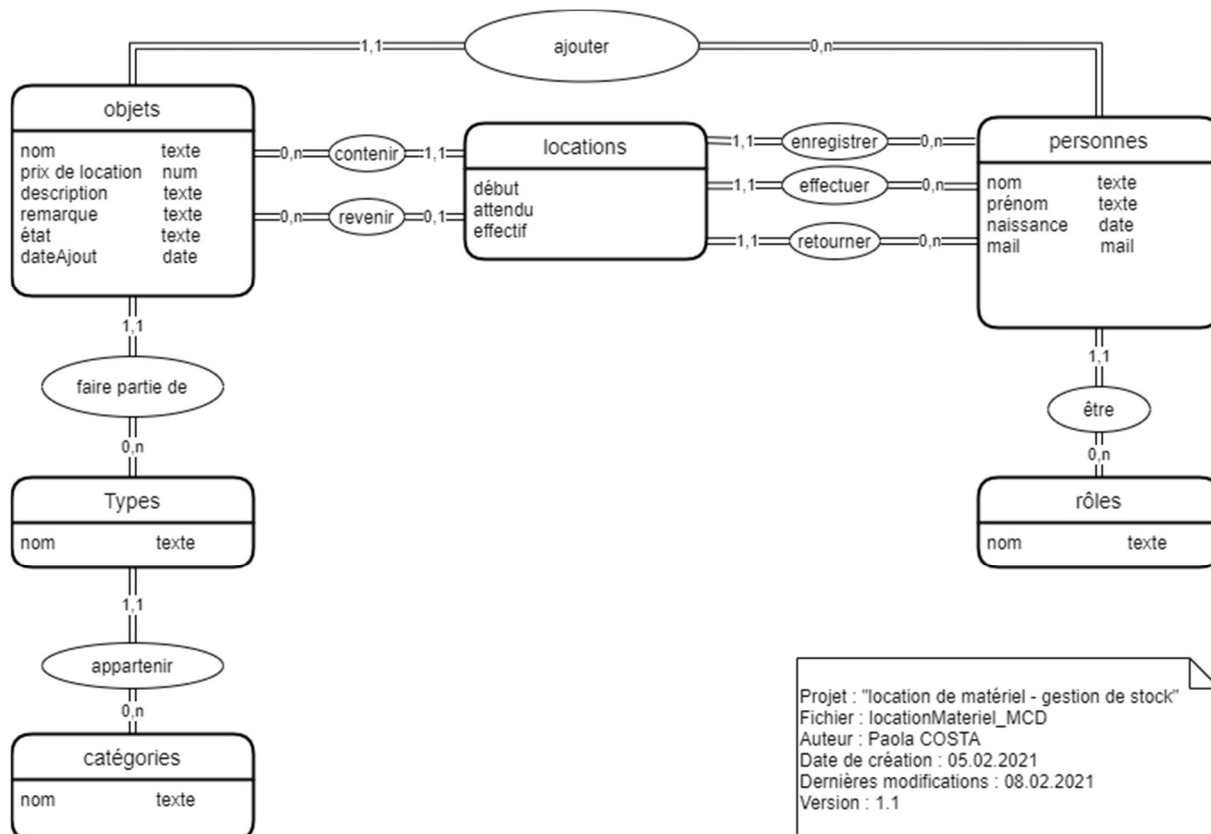


Figure 1 : modèle conceptuel de données

2.1.2.2 Modèle de données logique

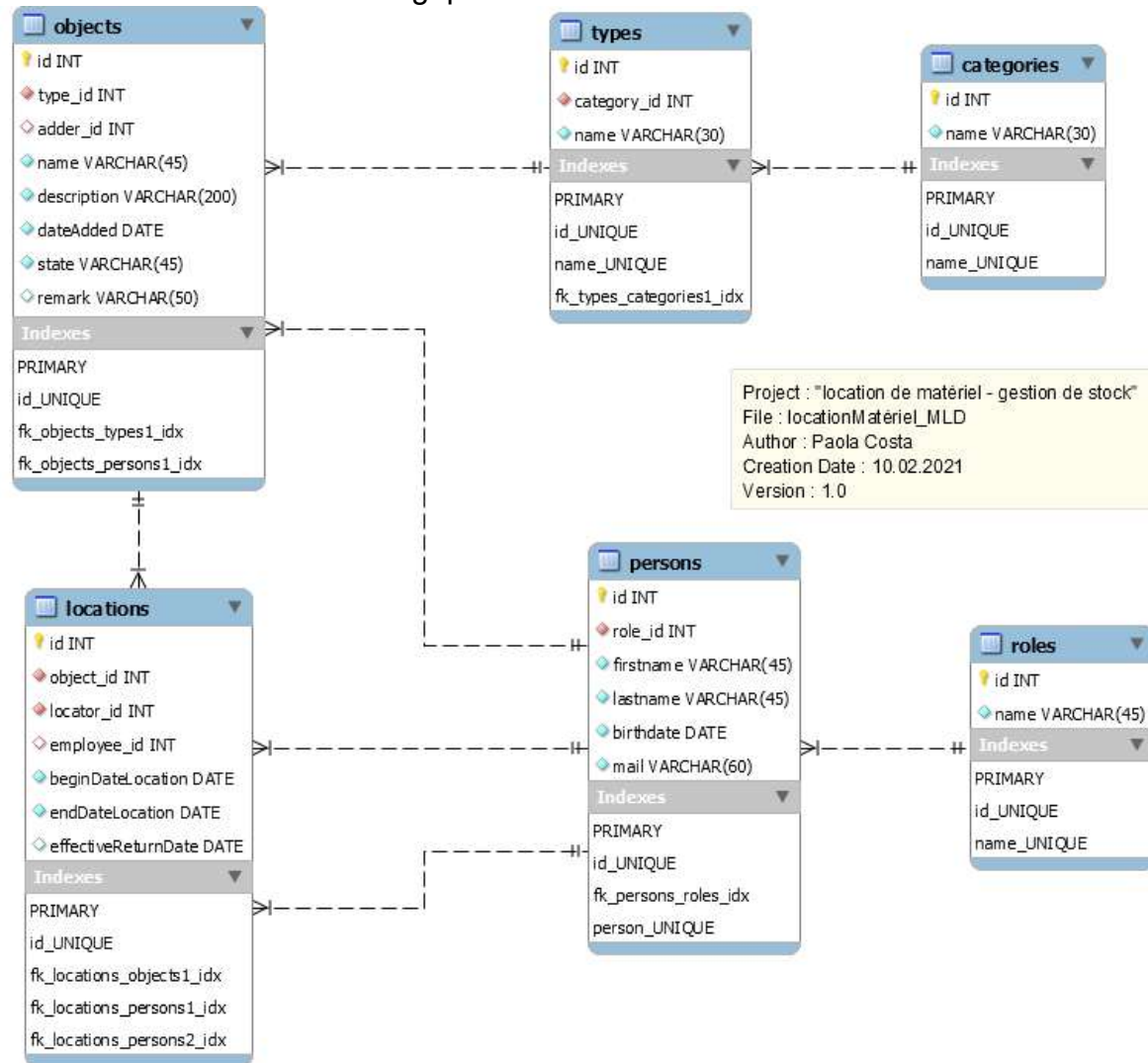


Figure 2 : modèle logique de données

Dans la table « locations », il y a trois clés étrangères :

- object_id représente l'objet qui est loué
- locator_id représente la personne qui loue l'objet
- employee_id représente la personne de l'entreprise qui enregistre la location. Cette information est présente afin de pouvoir créer des logs plus complets.

La table « roles » a pour but de classer les personnes en trois groupes : utilisateur, administrateur et client. Les clients n'ont pas accès à l'application. Les administrateurs gèrent les utilisateurs.

La fonctionnalité de gestion n'est pas encore mise en place, la présence de cette table a pour but de permettre une évolution future de l'application.

Dans la table « objects », la colonne adder_id sert à identifier la personne qui a ajouté l'objet dans le stock. Encore une fois, il s'agit de pouvoir créer un log complet.

La raison pour laquelle adder_id (table « objects ») et employee_id (table « locations ») peut être nul est pour permettre la suppression d'un employé (rôle utilisateur) sans être impacté parce qu'il aurait procédé à une location non finie. Cela aura pour conséquence d'afficher « NULL » à l'emplacement de son nom si on récupère les logs.

2.2 Maquettes

L'intégralité des maquettes se rapportant à l'application seront présentées ici. Celles-ci permettent de visualiser l'apparence du produit fini.

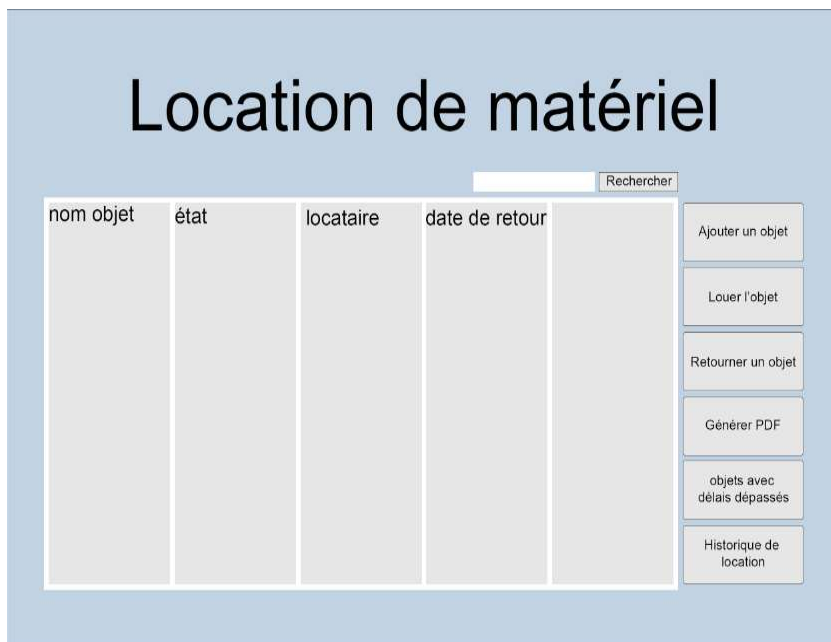


Figure 3 : maquette – vision globale de l'application

Cette maquette représente le visuel global de l'application, avec l'intégralité des fonctionnalités.

Les boutons à droite servent à effectuer les actions sur l'application. Celles-ci sont variables.

Certains des boutons permettent d'ouvrir des formulaires à remplir, afin de renseigner des détails supplémentaires.

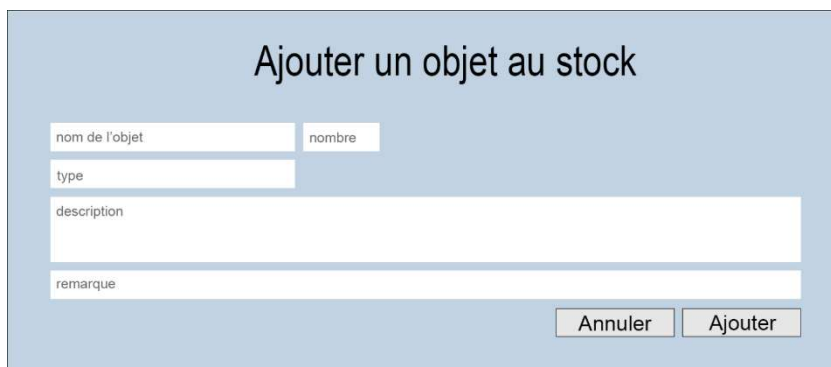


Figure 4 : maquette – formulaire d'ajout d'objet

Cette maquette montre l'apparence d'un formulaire. Il s'agit ici de celui d'ajout d'un objet au stock.

Les textes en gris clair représentent des placeholders. Ils sont visibles tant que le curseur n'est pas dans le champ texte.

Tout le projet va fonctionner sur le principe de ces deux maquettes. Dès qu'un formulaire doit être utilisé, le visuel sera semblable à celui de la Figure 2. Un pop-up de confirmation valide chaque clic sur un bouton, la sélection par défaut de la confirmation est « Valider ».

2.3 Use Cases & Scénarii

2.3.1 Use Cases

Comme montré par le schéma, l'intégralité des actions se font en tant qu'employé. Ce diagramme permet de montrer les cas d'utilisation de l'application.

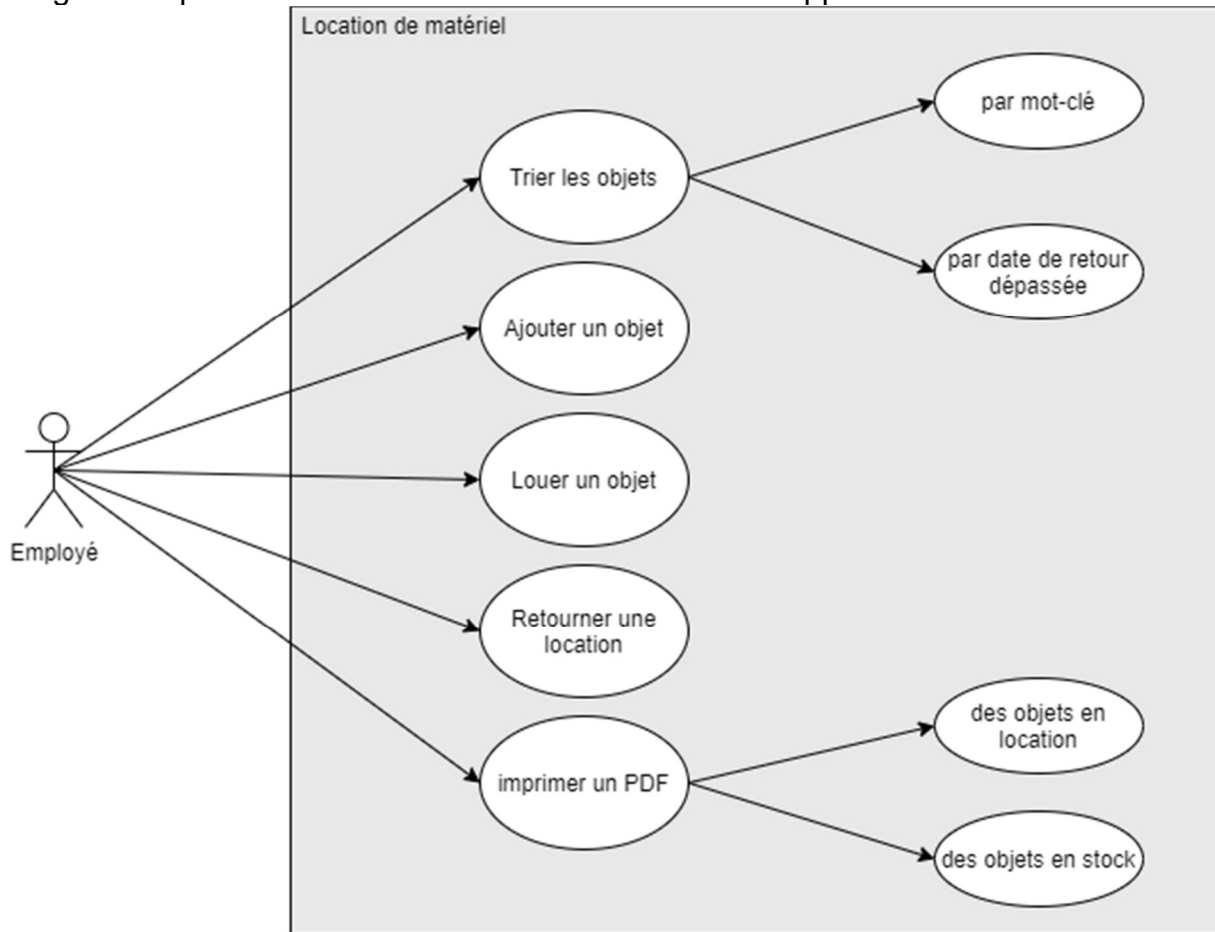


Figure 5 : use cases

2.3.2 Scénarii

2.3.2.1 Ajouter un objet

Action	Condition	Réaction
L'employé clique sur « ajouter un objet »		
L'employé donne un nom générique à l'objet (clé à molette)	Le nom n'existe pas du tout	Un formulaire s'ouvre, il faut remplir les spécificités de l'objet
	Le nom existe déjà (clé à molette 2)	Le système demande une confirmation qu'il s'agit du même objet
Validation de la pop-up	il s'agit du même objet	L'objet est ajouté avec un nouveau numéro aux objets en stock
	Il ne s'agit pas du même objet	Un formulaire s'ouvre, il faut remplir les spécificités de l'objet
L'employé entre les spécificités et valide et ajoute l'objet	Toutes les données sont entrées	L'objet est ajouté au stock, message comme quoi ça a bien fonctionné, retour à l'accueil
	Des données sont manquantes, de type erroné	Un message d'erreur apparaît, retour à la page d'accueil de l'application

2.3.2.2 Afficher les objets dont le délai est dépassé

Action	Condition	Réaction
L'employé clique sur « afficher les objets avec délai dépassé »		Seul les objets actuellement en location et dont la date de retour prévue est antérieure à la date actuelle sont affichés à l'écran

2.3.2.3 Générer un PDF

Action	Condition	Réaction
L'employé clique sur « générer un PDF »		Une pop-up avec 3 options s'affiche : « annuler », « location », « en stock »
L'employé sélectionne « location »		Un PDF avec tous les objets loués est généré
L'employé sélectionne « en stock »		Un PDF avec tous les objets en stock est généré
		Retour à la page d'accueil
L'employé sélectionne « annuler »		Retour à la page d'accueil

2.3.2.4 Louer un objet

Action	Condition	Réaction
L'employé clique sur « louer un objet »		
L'employé tape le nom ou l'id de l'objet	Le nom ou l'id existe dans la base de données et l'objet n'est pas loué	Une pop-up de confirmation apparaît, demandant si c'est le bon objet à louer
	Le nom ou l'id sont incorrects l'objet avec ce nom est déjà loué	Un message d'erreur apparaît, retour à la page d'accueil
	Le nom ou l'id sont incorrects l'objet avec cet ID est déjà loué	
L'employé invalide la confirmation de l'objet sélectionné		Retour à la page d'accueil
L'employé valide la confirmation		Un formulaire de location s'ouvre, pour spécifier les données du locataire
L'employé remplit le champ n° de client et la date de retour de location. Le reste du formulaire est grisé	Le client existe	Une pop-up de confirmation apparaît
	Le client n'existe pas	Retour à l'accueil
L'employé remplit tout le formulaire, sauf le champ n° de client	Un client identique existe déjà	Une pop-up pour savoir si le client voulu est celui affiché apparaît
	Le client n'existe pas	Une pop-up de confirmation de la location apparaît
Le client est bien celui indiqué		Une pop-up de confirmation de la location apparaît
Le client n'est pas le même		Un message d'erreur apparaît, retour à l'accueil
Validation de la confirmation de location	Le client existait déjà	Message de réussite, retour à l'accueil
	Le client n'existait pas	Message de réussite, message d'ajout du client à la base de données, retour à l'accueil

2.3.2.5 Retourner un objet loué

Action	Condition	Réaction
L'employé clique sur « retourner un objet »		Un formulaire s'affiche
L'employé entre le n° ou le nom de l'objet et valide	Le nom ou l'id est valide	La liste des éléments correspondants loués apparaît sur le formulaire
	Le nom et/ou l'id sont invalides	Message d'erreur. Retour à la page d'accueil
L'employé sélectionne l'objet qu'il doit retourner et valide		Une pop-up s'affiche pour valider
L'employé valide		Message de succès et retour à la page d'accueil
L'employé annule.		Retour à la page d'accueil

2.3.2.6 Rechercher par mot-clé

Action	Condition	Réaction
L'employé tape du texte dans le champ de recherche et appuie sur le bouton « Rechercher »	Le champ de texte est vide	Rien ne se passe
	Le champ de texte contient du texte	Un tri s'effectue. Seules les données avec une correspondance partielle s'affichent
	Il n'y a aucune correspondance entre les données et le texte de recherche	Aucune entrée n'est présente dans le tableau
	Toutes les données correspondent à la recherche	Toutes les données s'affichent dans le tableau
L'employé tape un autre texte et appuie sur « Rechercher »		La recherche repart de zéro, la liste affichée à l'écran n'impacte pas le résultat

2.4 Diagrammes

2.4.1 Diagramme de classe

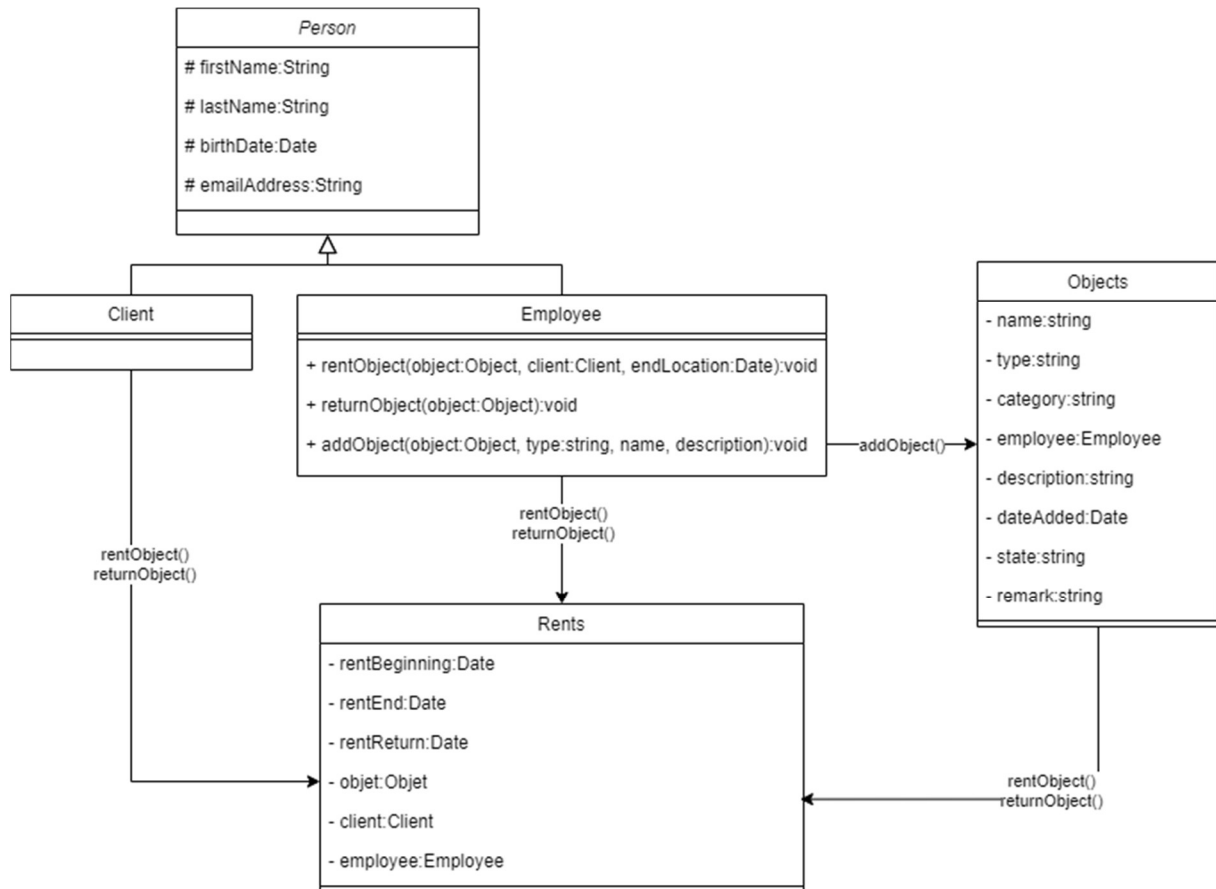


Figure 6 : diagramme de classe

2.4.2 Diagramme de flux

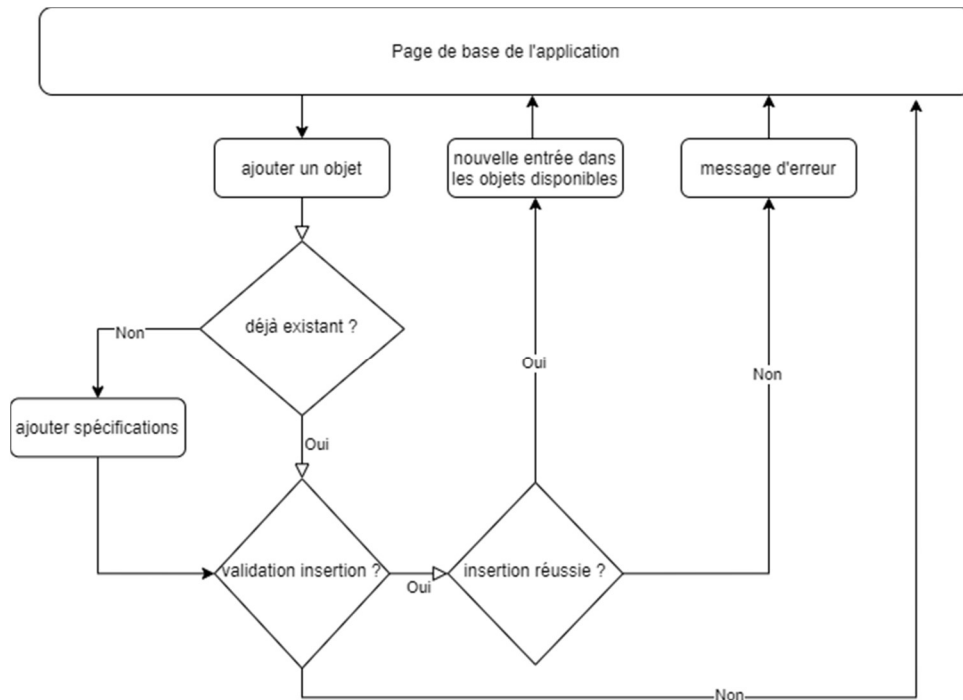


Figure 7 : diagramme de flux – ajouter un objet

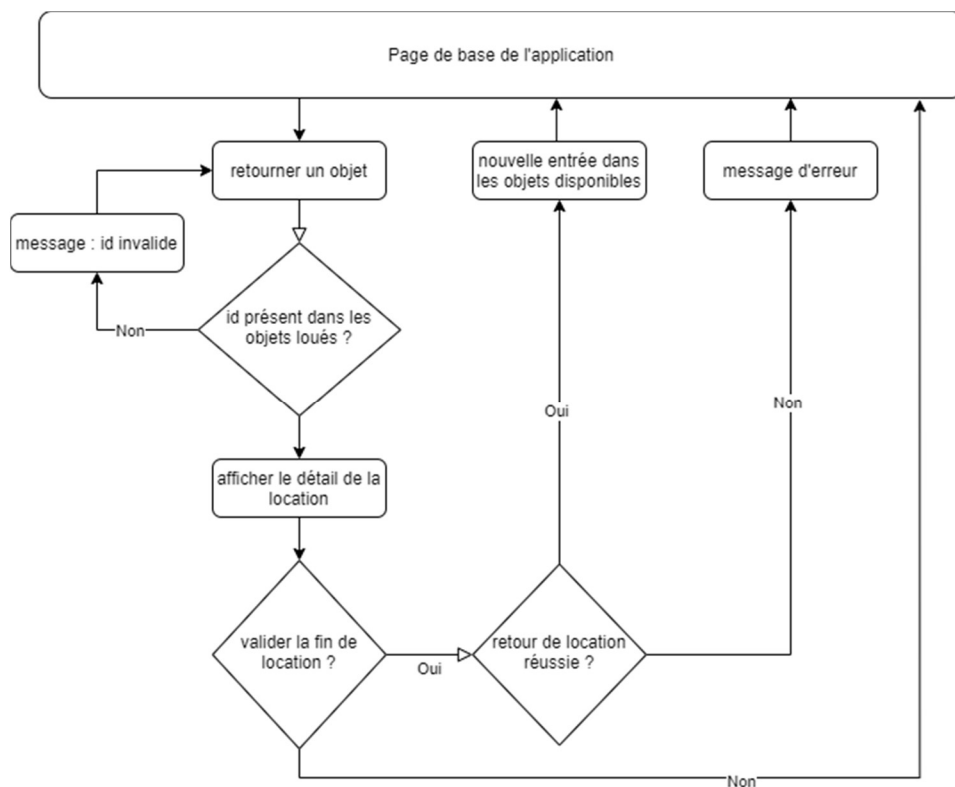


Figure 8 : diagramme de flux – retourner un objet

2.5 Stratégie de test

Afin de tester l'application, trois types de tests vont être effectués : les tests unitaires, les tests d'intégration et les tests fonctionnels. Ils sont réalisés de manière automatique grâce à des classes de tests spécifiques sur Visual Studio 2019. Les tests d'intégration s'effectuent entre différentes unités et valide le bon fonctionnement de ces unités entre elles. Les tests fonctionnels permettent de valider que l'application fonctionne comme elle doit fonctionner et fait ce qui est attendu.

Pour pouvoir détecter les problèmes au plus vite, les tests seront réalisés tout au long du développement. Chaque fonctionnalité subira tout d'abord un test unitaire. Lors de la liaison de cette fonctionnalité avec une autre, un test d'intégration sera fait. Les tests fonctionnels consisteront à vérifier que les fonctionnalités intégrées à l'application fonctionnent effectivement et retournent ce qu'elles doivent.

Les tests unitaires se doivent d'être exhaustifs, afin de détecter les soucis avant d'intégrer l'élément au projet. Autant que possible, les tests d'intégration et fonctionnels seront eux aussi exhaustifs.

Dans le cadre des tests fonctionnels, les données à utiliser pour réaliser les tests seront les données réelles, fournies avec le script de création de la base de données. Les tests unitaires et d'intégration se feront directement sur Visual Studio 2019, avec des classes de tests spécifiquement créées.

La majorité des tests étant automatisée, aucun testeur externe ne sera utilisé.

2.6 Risques techniques

Malheureusement, le risque zéro n'existe pas. Voici donc les principaux risques liés à ce projet, ainsi que les solutions utilisées pour garantir la meilleure réalisation possible. Tout d'abord, le risque de ne pas réussir à résoudre un souci d'implémentation est relativement élevé. Afin de garantir la détection la plus rapide et le temps de débogage le plus court possible, des tests unitaires seront mis en place. Cette mise en place est prioritaire, car elle permettra d'éviter une perte de temps conséquente par la suite.

Ensuite, la liaison de la base de données à l'application C# est un élément technique spécifique, qui n'a pas été abordé en cours. Pour éviter une perte de temps liée à l'apprentissage de l'implémentation de cet élément, l'aide de Mme F.Andolfatto sera demandée. Il s'agit d'une enseignante d'informatique au CPNV.

Enfin, la longueur du projet peut être un souci majeur. Afin d'éviter cela, la planification proposée respecte des délais assez serrés, laissant cependant un peu de marge en cas de besoin. Cela permet d'empiéter sur cette marge si une tâche en vient à s'éterniser.

2.8 Infrastructure

-
- Fournir tous les documents de conception:
- - le choix du matériel HW
 - le choix des systèmes d'exploitation pour la réalisation et l'utilisation
 - le choix des outils logiciels pour la réalisation et l'utilisation
 - site web: réaliser les maquettes avec un logiciel, décrire toutes les animations sur papier, définir les mots-clés, choisir une formule d'hébergement, définir la méthode de mise à jour, ...
 - bases de données: décrire le modèle relationnel, le contenu détaillé des tables (caractéristiques de chaque champs) et les requêtes.
 - programmation et scripts: organigramme, architecture du programme, découpage modulaire, entrées-sorties des modules, pseudo-code / structogramme...
-
- Le dossier de conception devrait permettre de sous-traiter la réalisation du projet !

2.8.1 Matériel hardware et système d'exploitation

Machine à disposition : un ordinateur du CPNV sous Windows10 x64

2.8.2 Outils logiciels

Développement en C# : Microsoft Visual Studio Enterprise 2019, Version 16.8.5

Paquet : MySql.Data version 8.0.23

Base de données : MySQLWorkbench, version 6.3.6 build 511 CE (64bits)

MySQL :

3 Réalisation

3.1 Problèmes rencontrés et résolution

[18.02.2021] Script d'insertion des données

Lors de la création du script d'insertion des données, deux soucis majeurs se sont posés. Le premier a été de trouver des données pertinentes. Le but est d'avoir des données suffisamment variées pour que toutes les possibilités soient clairement visibles, sans pour autant avoir 1'500 entrées dans la base de données.

Ce souci a été résolu à l'aide de Wikipédia, où je me suis inspirée pour nommer les catégories et les types.

[01.03.2021] Mise en place des classes pour les tests unitaires

Lors de la mise en place des classes des tests unitaires le lundi matin 1^{er} mars 2021, un souci s'est présenté : les classes appelées depuis le reste du projet n'étaient pas reconnues par la classe de test.

L'aide de l'enseignant J.Ithurbide a permis de résoudre le problème (la classe « Objects » n'était pas déclarée publique et le deuxième projet de la solution n'était pas référencé dans le projet de test) et a prévenu un souci qui aurait pu se poser au futur : l'organisation des fichiers.

L'organisation qu'il a proposée suit une architecture MVC, qui permet de séparer en 3 blocs distincts l'application.

[05.03.2021] Interaction entre plusieurs fichiers

Lors de l'implémentation des premières requêtes (fichier « RequestDB »), au moment de faire appel à des méthodes présentes dans un autre fichier, des erreurs de référence sont apparues.

L'aide de F.Andolfatto a permis de corriger le problème très rapidement : il s'agissait d'un oubli de ma part. Aucun objet n'était initialisé au début de la classe.

[05.03.2021] Faire appel à un autre projet dans une solution unique

La modification de la structure du projet en architecture MVC a vu un nouveau problème apparaître : la présence de types de projets différents. L'entièreté des données sont stockées dans un projet de type librairie, alors que les visuels et le contrôleur sont dans un projet de type WindowsForm. Hors, dans un projet de type « Bibliothèque de classe (.NET standard) », aucune référence ne s'affichait.

Avec F.Andolfatto, plusieurs tests successifs de création de projets ont permis de démontrer que le type de projet utilisé ne permettait pas le référencement. Afin de pouvoir avoir des références, il a été nécessaire de déplacer les fichiers dans un nouveau projet avec un type différent : « Bibliothèque de classe (.NET framework) ». Ce type permet de référencer d'autres projets. Il est cependant incompatible avec des systèmes IOS ou Android.

3.2 Dossier de réalisation

3.2.1 Répertoires et fichiers du projet

Décrire la réalisation "physique" de votre projet

- les répertoires où le logiciel est installé
- la liste de tous les fichiers et une rapide description de leur contenu (des noms qui parlent !)

3.2.2 Produit fini

- le numéro de version de votre produit !
- les versions des systèmes d'exploitation et des outils logiciels
- la description exacte du matériel (??)

3.2.3 Programmation et scripts

Interaction avec la base de données :

- programmation et scripts: librairies externes, dictionnaire des données, reconstruction du logiciel - cible à partir des sources.

NOTE : Evitez d'inclure les listings des sources, à moins que vous ne désiriez en expliquer une partie vous paraissant importante. Dans ce cas n'incluez que cette partie...

3.3 Description des tests effectués

Pour chaque partie testée de votre projet, il faut décrire:

- les conditions exactes de chaque test
- les preuves de test (papier ou fichier)
- tests sans preuve: fournir au moins une description

3.4 Erreurs restantes

S'il reste encore des erreurs:

- Description détaillée
- Conséquences sur l'utilisation du produit
- Actions envisagées ou possibles

3.5 Liste des documents fournis

Lister les documents fournis au client avec votre produit, en indiquant les numéros de versions

- le rapport de projet
- le manuel d'Installation (en annexe)

- le manuel d'Utilisation avec des exemples graphiques (en annexe)
- autres...

4 Conclusions

- Développez en tous cas les points suivants:
- - Objectifs atteints / non-atteints
 - Maintien des délais prévus
 - Points positifs / négatifs
 - Difficultés particulières
 - Suites possibles pour le projet (évolutions & améliorations)

5 Annexes

5.1 Résumé du rapport du TPI / version succincte de la documentation

5.2 Sources – Bibliographie

5.2.1 Livres utilisés

5.2.2 Pages internet consultées

Page des TPI vaudois : <http://www.tpivd.ch/> [12.02.21]

Recherche d'informations sur le contenu nécessaire dans le rapport de projet

Documentation Microsoft, pour la création de tests unitaires [01.03.2021]

<https://docs.microsoft.com/en-us/visualstudio/test/walkthrough-creating-and-running-unit-tests-for-managed-code?view=vs-2019>

Recherche de la manière correcte de mettre en place un projet de tests dans une solution

Forum commentcamarche.net, pour la syntaxe d'appel de formulaire [04.03.2021]

<https://codes-sources.commentcamarche.net/forum/affich-150911-appel-d-un-form-a-partir-d-un-autre-en-c>

Recherche de la syntaxe pour faire un appel sur un autre formulaire, afin d'avoir un code le plus propre possible

5.2.3 Personnes consultées

Mme F.Andolfatto

- [05.02.21] Précisions sur la liaison des bases de données à un projet en C#
- [12.02.21] Retour sur la pertinence et l'exactitude d'un diagramme de classe
- [05.03.21] Interactions entre différents fichiers et projets

M J.Ithurbide

- [01.03.21] Résolution du problème de mise en place du projet de test et explication sur l'architecture MVC dans un projet incluant Windows Form et librairie

5.3 Journal de travail

Ceci est le journal de travail de la semaine 6 de 2021. Il s'agit de la semaine n°1 du projet, du 1^{er} février 2021 au 5 février 2021.

Date	Semaine	Activité	Avancée	Heures	Remarques
01.02.2021	6	55 -		0:30	Discussion du déroulement du projet, reçu du CdC, réponses aux
01.02.2021	6	12 Planification initiale	En cours	1:00	Mise en place du fichier excel de la planification
01.02.2021	6	51 Rapport de pré-TPI		0:10	Mise en place du canevas pour le rapport complet
01.02.2021	6	12 Planification initiale	Terminé	0:30	complétion de la planification initiale
03.02.2021	6	13 Objectifs du projet	Terminé	0:30	rédaction des objectifs
03.02.2021	6	11 Analyse préliminaire	En cours	0:30	réflexion sur l'introduction au projet et comment la rédiger, prise de notes -> structure du paragraphe
04.02.2021	6	11 Analyse préliminaire	Terminé	0:30	Rédaction de l'introduction
04.02.2021	6	14 Stratégie de tests	Terminé	0:30	Rédaction de la stratégie de test
04.02.2021	6	22 Maquettes	En cours	0:20	début des maquettes
05.02.2021	6	12 Planification initiale	Terminé	0:45	correction d'une erreur -> passage d'un tableau d'heures à un diagramme de Gant
05.02.2021	6	51 Rapport de pré-TPI		0:10	ajout de la planification initiale au dossier
05.02.2021	6	15 Risques techniques	Terminé	0:30	rédaction des risques techniques du projet
05.02.2021	6	22 Maquettes	En cours	0:35	avancée sur maquette de l'application
05.02.2021	6	55 -		0:10	retour sur le rendu de la planification discussion du CdC
05.02.2021	6	22 Maquettes	En cours	0:20	avancée sur les maquettes -> la maquette de globale de l'application est terminée
05.02.2021	6	51 Rapport de pré-TPI		0:45	travail sur le visuel du document, avec mise en place de sections, d'une page de garde, d'une table des matières
05.02.2021	6	21 Base de données (MCD, MLD)	En cours	0:20	mise en place d'un fichier pour le MCD et réflexion sur les entités et relations

Ceci est le journal de travail de la semaine 7 de 2021. Il s'agit de la semaine n°2 du projet, du 8 février 2021 au 12 février 2021.

Date	Semaine	Activité	Avancée	Heures	Remarques
08.02.2021	7	22 Maquettes	En cours	0:45	maquette du formulaire d'ajout d'objet faite
08.02.2021	7	51 Rapport de pré-TPI		0:45	ajout de la maquette à la conception fermeture du point maquette correction d'une erreur de pied-de-page
08.02.2021	7	21 Base de données (MCD, MLD)	En cours	0:20	avancée du MCD : ajout table locataires, objets, types création des relations
08.02.2021	7	21 Base de données (MCD, MLD)	En cours	1:10	fin du MCD
08.02.2021	7	51 Rapport de pré-TPI		0:15	ajout du MCD au rapport (insertion image) rédaction d'un paragraphe d'explication
10.02.2021	7	21 Base de données (MCD, MLD)	Terminé	1:00	création du MLD sur la base du MCD
11.02.2021	7	51 Rapport de pré-TPI		0:20	le paragraphe concernant le MLD était hors-sujet -> suppression complétion de la partie MCD/MLD (insertion images, rédaction du texte)
11.02.2021	7	23 Diagrammes	En cours	0:50	création du diagramme de flux "ajouter objet" et "retourner objet"
11.02.2021	7	51 Rapport de pré-TPI		0:20	insertion des diagrammes dans le rapport ajout des légendes aux différentes figures du document
12.02.2021	7	55 -		0:15	discussion sur la pertinence du MCD/MLD retour sur le rendu du rapport de la semaine précédente -> les objectifs sont mal situés, leur précision les emmène dans le concept, ceux de l'analyse préliminaire sont à refaire
12.02.2021	7	23 Diagrammes	En cours	0:30	création du diagramme de classe -> l'aide de Mme Andolfatto sera nécessaire pour certifier l'exactitude du diagramme
12.02.2021	7	30 Base du projet C# (classe ...)	En cours	0:15	mise à jour de l'application création des classes et du windows form principal
12.02.2021	7	23 Diagrammes	Terminé	0:10	discussion avec Mme Andolfatto pour valider l'exactitude du diagramme
12.02.2021	7	24 Cas d'utilisation	Terminé	0:20	Création du diagramme pour les use cases
12.02.2021	7	51 Rapport de pré-TPI		0:45	ajout du diagramme des Use Cases insertion des légendes aux figures insertion des sources dans le rapport déplacement des objectifs dans le paragraphe correct, rédaction d'un texte simplifié pour les objectifs de l'analyse préliminaire début de la rédaction de l'infrastructure utilisée pendant le projet

Ceci est le journal de travail de la semaine 8 de 2021. Il s'agit de la semaine n°3 du projet, du 15 février 2021 au 19 février 2021.

Date	Semaine	Activité	Avancée	Heures	Remarques
15.02.2021	8	25 Scénarii	En cours	0:50	Rédaction des scénarii
15.02.2021	8	25 Scénarii	Terminé	0:45	Fin de la rédaction des scénarii
15.02.2021	8	16 Planification définitive	Terminé	0:25	Ajout de la planification définitive -> l'analyse/conception est terminée dans sa majorité, seul du détail au niveau du hardware est manquant -> il sera précisé au fil de la semaine, quand les systèmes seront tous en place les uns par rapport aux autres
15.02.2021	8	31 Script création BDD	Terminé	0:25	Le script de création de la base de données a été généré sur la base du MCD et du MLD
15.02.2021	8	30 Base du projet C# (classe ...)	En cours	0:30	Réflexion sur la meilleure façon de mettre en place les fichiers de l'application
17.02.2021	8	30 Base du projet C# (classe ...)	En cours	0:50	Mise en place du visuel de la page d'accueil de l'application, avec création des boutons, champs de recherche, etc création des constructeurs des classes Persons, Employees et des méthodes
17.02.2021	8	23 Diagrammes		0:05	Correction d'une erreur d'accessibilité sur les méthodes elles sont publiques et non pas privées
18.02.2021	8	30 Base du projet C# (classe ...)		0:10	Finition de l'implémentation de l'héritage sur "Clients"
18.02.2021	8	32 Script insertion données	En cours	0:40	Mise en place de l'insertion des données
18.02.2021	8	51 Rapport de pré-TPI		0:05	Correction d'une erreur de numérotation remplacement du diagramme de classe -> erreur de typo
18.02.2021	8	32 Script insertion données	En cours	0:25	Avancée de l'insertion des données prend plus de temps que prévu, à cause de la recherche pour trouver des informations concrètes
19.02.2021	8	55 -		0:15	discussion avec le chef de projet sur l'avancée et réponses aux questions de compréhension dans le rapport validation de l'avancée + mise en avant des corrections à apporter
19.02.2021	8	51 Rapport de pré-TPI		0:05	correction de termes inadéquats dans la partie analyse
19.02.2021	8	32 Script insertion données	En cours	0:30	réflexion sur les données à mettre dans les tables
19.02.2021	8	32 Script insertion données	En cours	1:00	avancée du script d'insertion des données -> pas de soucis pour les tables de références -> fini -> erreur de format de date pour les dates de naissance, je sèche et donc je perds du temps
19.02.2021	8	32 Script insertion données	En cours	1:00	erreur trouvée et corrigée avancée sur les données insérées

5.4 Manuel d'Installation

5.5 Manuel d'Utilisation

5.6 Archives du projet

Media, ... dans une fourre en plastique

5.7 Table des illustrations

Figure 1 : modèle conceptuel de données.....	6
Figure 2 : modèle logique de données	7
Figure 3 : maquette – vision globale de l'application	8
Figure 4 : maquette – formulaire d'ajout d'objet.....	8
Figure 5 : use cases	9
Figure 6 : diagramme de classe	13
Figure 7 : diagramme de flux – ajouter un objet	14
Figure 8 : diagramme de flux – retourner un objet.....	14