|  |
| --- |
| P\_Fun Covid |

Lucas Lordon – MID3

Vennes – B22

Xavier Carrel

Table des matières

[1 Analyse préliminaire 3](#_Toc175917886)

[1.1 Introduction 3](#_Toc175917887)

[1.2 Objectifs 3](#_Toc175917888)

[1.3 Gestion de projet 4](#_Toc175917889)

[2 Analyse / Conception 4](#_Toc175917890)

[2.1 Gameplay 4](#_Toc175917891)

[2.2 Concept 4](#_Toc175917892)

[2.3 Analyse fonctionnelle 4](#_Toc175917893)

[2.4 Stratégie de test 4](#_Toc175917894)

[3 Réalisation 4](#_Toc175917895)

[3.1 Points de design spécifiques 4](#_Toc175917896)

[*3.1.1* *…* 5](#_Toc175917897)

[*3.1.2* *…* 5](#_Toc175917898)

[*3.1.3* *…* 5](#_Toc175917899)

[3.2 Déroulement 5](#_Toc175917900)

[3.3 Mise en place de l’environnement de travail 5](#_Toc175917901)

[3.4 Description des tests effectués 6](#_Toc175917902)

[3.5 Erreurs restantes 6](#_Toc175917903)

[4 Conclusions 6](#_Toc175917904)

[5 Annexes 7](#_Toc175917905)

[5.1 Manuel de référence 7](#_Toc175917906)

[5.2 Journal de travail 7](#_Toc175917907)

# Analyse préliminaire

## Introduction

Le projet « P\_Fun Covid » à été réaliser par Lucas Lordon dans le cadre du module de Programmation Fonctionnel I323. Ce projet à pour objectif d’afficher des graphiques à partir d’un jeu de données choisi. Les technologies utiliser sont libres du moment que l’on code en C# dans l’IDE Visual Studio 2024.

Personnellement j’ai créé un projet Windows Forms reposant sur le package nugget « ScottPlot Form ». Le jeu de donnée choisit vient de ce repo GitHub : https et concerne différente information collecter durant le covid par canton Suisse.

## Objectifs

Filtre sur les dates :

Cet objectif a été atteint car je peux filtrer les dates grâce à deux élément « DateTimePicker » et à une méthode utilisant Linq.

Au moins 3 courbes :

Cet objectif a été atteint car par défaut je peux afficher entre 1 et 26 courbes simultanément via les fichier CSV disponible sous le répertoire suivant : P\_Fun\Code\Data

L’axe X correspond à la temporalité : Cet objectif a été atteint voir image ci-dessou :  
Une image contenant texte, Tracé, ligne, diagramme

Description générée automatiquement

## Gestion de projet

La gestion de projet c’est effectuer via deux outils : IceScrum et un jounal de travail Excel. IceScrum pour la partie planification et validation du travaille effectuer. Le journal de travail pour avoir un suivit sur le travail effectuer.

Il est important de noter que la méthodologie utiliser est l’agilité (ce référer au Scrum guide) et que le chef de projet a été l’enseignant X.Carrel.

# Analyse / Conception

## Domaine

J’ai choisi le domaine des maladies et plus précisément du covid 19 en suisse. Les données sont un ensemble d’information tels que le nombre d’hospitalisation ou bien le nombre de mort journalier, ces données sont récoltées par chaque canton et enregistrer dans différents fichiers CSV.

Echelle de temps est très variable selon les maladies qui pourrait être étudier mais dans le cas du covid 19 les données commencent en 2019 et finissent en 2024 (Attention : selon le canton les date peuvent différée légèrement).

Ces données sont principalement destinées à des professionnelle du monde de la santé, des analyste. Mais l’on peut très bien imaginer d’autres cas d’utilisation comme des journaliste voulant appuies leur propos au moyen de graphique. L’on peut aussi imaginer que des professionnelles du domaine de la finance si intéresse dans le but d’expliquer la crise économique durant la période de pandémie.

## Concept

* *Diagramme de classe*
* *Diagramme(s) d’état*

## Analyse fonctionnelle

***Reprendre le contenu des User Stories d’IceScrum : Story + tests d’acceptance (avec IceTools) + maquettes***

## Stratégie de test

La stratégie de test repose sur plusieurs point essentiel pouvant être séparer en trois grand bloque : Planification, Code et Utilisateur.

Planification : durant cette partie des test l’on crée des user story et l’on y implémente des tests d’acceptances. C’est une partie très importantes où l’on confronte la fonctionalitée avant même de la coder. Cela permet de gagner beaucoup de temps sur le long therme en évitant de devoir foncer dans le code tête baisser pour ce rendre compte que l’on code quelque chose qui ne nous a pas été demander.

***Décrire quels sont les MOYENS utilisés pour faire les tests, ne pas décrire les tests à effectuer !!!***

***Décrire l’environnement dans lequel se fait la sprint review***

*Décrire la stratégie globale de test :*

* *types de des tests et ordre dans lequel ils seront effectués.*
* *les moyens à mettre en œuvre.*
* *couverture des tests (tests exhaustifs ou non, si non, pourquoi ?).*
* *données de test à prévoir (données réelles ?)* ***et comment elles seront mises en place****.*
* *les testeurs extérieurs éventuels.*

# Réalisation

## Points de design spécifiques

***Ce chapitre est constitué de plusieurs sous-chapitre.***

***Chaque sous-chapitre explique un point de design technique particulier, quelque chose que vous avez dû inventer pour répondre au besoin et qui ne peut pas s’expliquer par de simples commentaires dans le code.***

***Il s’agit d’explications techniques sur le fonctionnement du système. Les explications sont appuyées par des diagrammes, ou de très brefs éléments de code.***

***NE PAS mettre ici des pratiques usuelles que tout professionnel de la branche connaît déjà. Par exemple, n’EXPLIQUEZ PAS ICI CE QU’EST LE PATTERN MVC.***

***Exemple (simplifié à l’extrême) : Protection contre des formulaires mal intentionnés ou modifiés***

* ***Au moment de générer le formulaire, le script php :***
  + ***Concatène les noms de tous les champs contenus dans le formulaire***
  + ***Calcule un hash SHA256 de la chaîne obtenue***
  + ***Ajoute un input nommé « CSRF » de type hidden dans le form***
* ***A la réception du POST du fromulaire***
  + ***Concatène les noms des indices de $\_POST***
  + ***Calcule un hash SHA256 de la chaîne obtenue***
  + ***Vérifie que la valeur du champ CSRF correspond***

### *…*

### *…*

### *…*

## Déroulement

***Résumer comment s’est passé la réalisation de chaque story, ses difficultés, les alternatives envisagées mais rejetées, ses surprises, …***

## Mise en place de l’environnement de travail

* ***Comment accéder au code source***
* *la liste de tous les fichiers et une rapide description de leur contenu (des noms qui parlent !)*
* *les versions des systèmes d'exploitation et des outils logiciels*
* *la description exacte du matériel*

***Ce chapitre décrit précisément comment un employé qualifié peut recréer l’environnement dans lequel vous avez effectué ce travail***

## Description des tests effectués

***Reprendre les tests d’acceptance d’IceScrum au moyen de la feuille ad hoc d’IceTools***

## Erreurs restantes

*S'il reste encore des erreurs :*

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

***Reporter la*** [***dette technique***](https://www.premaccess.com/qu-est-ce-que-dette-technique-comment-la-maitriser/#:~:text=La%20dette%20technique%20survient%20quand,de%20plus%20en%20plus%20fr%C3%A9quents.) ***connue. S’appuyer sur la pratique des // TODO***

# Conclusions

*Développez en tous cas les points suivants :*

* *Objectifs atteints / non-atteints*
* *Points positifs / négatifs*
* *Difficultés particulières*
* *Suites possibles pour le projet (évolutions & améliorations)*

# Annexes

## Journal de travail