



## 1 INFORMATIONS GENERALES

Élève :	Nom: Nademo	Prénom: Yosef 
Lieu de travail :	ETML / Avenue de Valmont 28b, 1010 Lausanne	
Client	Nom: Carrel / Melly	Prénom: XAVIER / JONATHAN
	 xavier.carrel@eduvaud.ch / jonathan.melly@eduvaud.ch	
Dates de réalisation :	1 <sup>er</sup> trimestre	
Temps total :	24/32 périodes	

## 2 PROCÉDURE

- Tous les apprentis réalisent le projet sur la base d'un cahier des charges.
- Le cahier des charges est présenté, commenté et discuté en classe.
- Les apprentis sont entièrement responsables de la sécurité et sauvegarde de leurs données.
- En cas de problèmes graves, les apprentis avertissent le client au plus vite.
- Les apprentis ont la possibilité d'obtenir de l'aide externe, mais ils doivent le mentionner.
- Les informations utiles à l'évaluation de ce projet sont disponibles au chapitre 8.

## 3 TITRE

***Plot those lines !***

## 4 SUJET

Concevoir un logiciel pour afficher des graphiques sur des données

(Dans la suite de ce document, nous ferons référence à ce logiciel par le terme PTL – Plot Those Lines)

## 5 MATÉRIEL ET LOGICIEL À DISPOSITION

- Un PC ETML
- Accès à Internet
- Github / Github Project

## 6 PRÉREQUIS

Modules de programmation de base ainsi que le module 323

## 7 CAHIER DES CHARGES

### 7.1 Gestion de projet

Les directives spécifiques à la gestion de projet, vous seront fournis séparément par votre chef de projet. Dans tous les cas, les points suivants doivent être respectés :

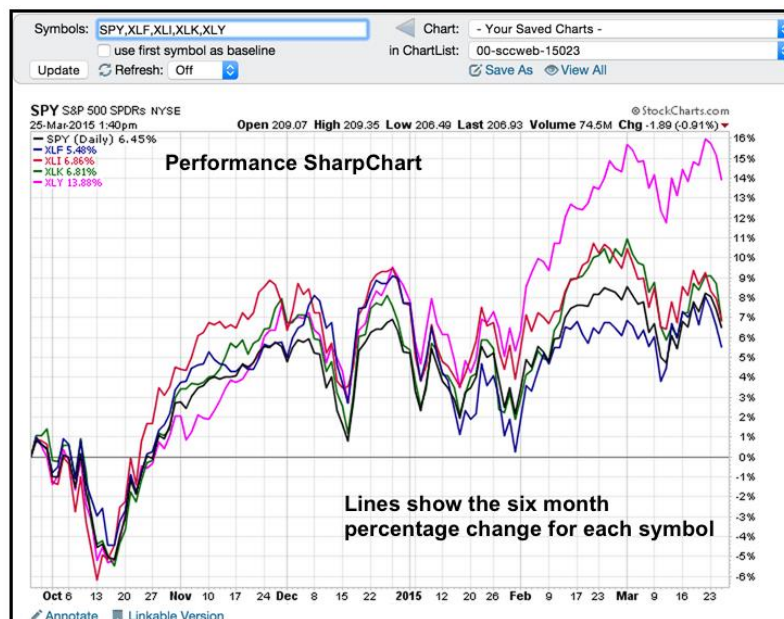
1. Vous êtes inscrit au projet P\_FUN sur Marketplace :  
<https://apps.pm2etml.ch/jobDefinitions/102>
2. Vous devez effectuer une analyse sous forme de User Stories (US) avec tests d'acceptance et maquettes **avant** de commencer à coder.
3. Vous devez fournir une planification simple de la réalisation de ces US **avant** de commencer à coder.
4. Vous devez tenir un Journal de travail

### 7.2 Qualité

1. Réaliser un programme informatique de qualité
  - Organisé (namespace, classes, commit log,...)
  - Compacté (pas de copié/collé,...)
  - Optimisé (utilisation de structures adaptées)
  - Testé (tests unitaires)
  - Commenté
  - Complet (code, modèle de données, maquettes PDF, exécutable, ...)
2. Prouver que vous êtes digne de confiance lorsqu'on vous confie un projet
  - Journal de travail à jour
  - Pro-activité
    - **Poser des questions** au client
    - Faire des démonstrations
    - Utiliser un système de versioning de code (GIT)

### 7.3 Illustration non contractuelle

Ci-dessous, à titre indicatif, un exemple avec un graphique présentant plusieurs courbes pour un axe temporel commun...



### 7.4 Domaine d'application

Le domaine dans lequel votre logiciel sera utilisé est à votre choix. Voici quelques exemples possibles :

- Météo : température, précipitations, pollen, pollution, ...
- Energie : consommation, production, ...
- Démographies : naissance, décès, âge, ...
- Sport : records, classements, budget, ...
- Santé : nombre de cas d'une maladie, nombre de pas, ...
- Trafic : sur internet, sur les routes, dans les trains, ...
- Finance : action en bourse, monnaie, matière première, ...
- ...

Une fois que vous avez choisi votre domaine, il vous faut définir par quel moyen vous allez obtenir les données sur lesquels vous allez travailler. Il existe un grand nombre de sources publiques pour ce genre d'information. Si vous n'en trouvez pas une qui vous convient, vous avez toujours la possibilité de générer des données fictives, par vous-même (script) ou avec l'aide d'une IA.

## 7.6 Fonctionnalités voulues par le client

- En tant qu'utilisateur, je veux afficher une représentation graphique de plusieurs séries temporelles (time serie) simultanément.
- Je veux pouvoir bénéficier d'une grande flexibilité d'affichage afin de pouvoir analyser mes données en détail.
- En tant qu'utilisateur, je veux importer des séries de données de façon permanente. PTL me permet d'importer un ou plusieurs formats de données, comme par exemple : fichiers CSV, fichiers JSON, JSON reçu d'une API,...
- En tant qu'utilisateur, je veux afficher plusieurs intervalles de temps pour une même donnée. Par exemple, si j'ai une série temporelle pour l'année 2010 et une autre série (fichier, source) pour l'année 2011, je veux pouvoir afficher en une seule ligne les deux années consécutives.
- En tant qu'utilisateur, je veux aussi avoir un mode pour afficher des fonctions plutôt que des séries temporelles (onglet, option, ...). Par défaut, je peux voir  $x^2$ ,  $\sin(x)$ ,  $\sin(x) + \sin(3x)/3 + \sin(5x)/5$ ,  $x * \sin(x)$ . De plus, un champ texte me permet d'écrire une expression personnalisée qui sera exécutée avec Roslyn dynamiquement. Les tokens "sin", "cos", "^" sont donc remplacés par leur équivalent C#.

Attention au temps à disposition ! Il se pourrait qu'il soit inadapté pour réaliser l'ensemble de ces désirs. A vous de proposer une planification réaliste au client après analyse des fonctionnalités.

### 7.6.1 Contraintes techniques

- Utiliser LINQ (pas de boucle for)
- Implémenter au moins 2 extensions du langage C# pour faciliter l'écriture du code
- Librairie graphique à choix (forms, maui, uno, wpf, fna, ...)
- Librairie pour présenter les données à choix (<https://scottplot.net/> par exemple)

## 7.7 Livrables

Il n'y a qu'un livrable : une release Github avec le code source et à laquelle sont attachés :

1. Rapport PDF contenant
  - a. Introduction, incluant :
    - Objectifs produit et pédagogique
    - Description du domaine (et des sources de données)
  - b. Planification
  - c. Rapport de tests
  - d. Journal de travail
  - e. Chapitre explicatif de l'usage fait de l'IA dans ce projet
  - f. Conclusion / bilan

## 9 Évaluation

1. Auto-évaluation challengée par le client basé sur des éléments observables. (voir grille)
2. Jokers.

Vous recevez trois Jokers en début de projet. Ces Jokers peuvent vous être retirés en cours de projet sur observation d'un comportement ou d'une pratique inacceptable, tels que :

- Du code est commité dans votre repo et vous n'êtes pas en mesure de l'expliquer de manière précise et convaincante.
- Abus des libertés qui vous ont données : pause trop longue, discussion extra professionnelles, inactivité, usage non professionnel des outils mis à votre disposition, ...
- Recours inadapté et répété à l'IA
- Comportement dérangeant

À noter : cette liste n'est pas exhaustive. Il est donc possible qu'un joker vous soit retiré pour une raison qui n'est pas listée ci-dessus.

L'impact des jokers sur l'évaluation est décrit dans la grille d'évaluation