



# Pick'UP - Suivi de projet

02.04.2021

Nel BOUVIER  
Kesly GASSANT  
Aleryc SERRANIA

Imad ROUAMI  
Niels DE BARBANSON

Loïc DUBOIS-TERMOZ  
Romain CHIKIROU

<b>Introduction</b>	<b>2</b>
<b>Users Stories</b>	<b>3</b>
Bob veut importer et afficher un plan de ville	3
Bob veut importer et afficher les points de P&D d'une planification de livraison sur le plan de ville	3
Bob veut voir le chemin optimisé d'un livreur	4
Bob veut ajouter une requête P&D	4
Bob veut supprimer une requête P&D	5
Bob veut déplacer un point de pickup ou de delivery	5
David veut visualiser l'ordre de passage des points de P&D	5
David veut connaître le chemin vers un point de passage	5
Bob veut changer l'ordre de passage des points	5
Bob veut paramétrer la durée maximale allouée dans le calcul du chemin	5
Bob veut paramétrer la durée maximale allouée dans le calcul du chemin	5
<b>Annexes</b>	<b>6</b>
Import du plan de la ville	6
Sélection du fichier dans le système de l'utilisateur (exemple sur Windows)	6
Affichage de la carte sans les points de P&D	7
Affichage de la carte avec les points de P&D	7

## Introduction

Pick'Up est une application développée dans le cadre du projet Optimod'Lyon porté par le Grand Lyon. Ce projet vise à proposer de nouveaux services pour la mobilité urbaine (exemple : récolte et fourniture d'information en temps réel sur l'état du réseau urbain). L'objectif de l'application Pick'Up est de permettre à des sociétés de livraison de préparer leurs tournées. Celle-ci est spécialisée dans le service de P&D (Pickup & Delivery) à vélo. Ce document vise à suivre l'évolution de l'organisation de l'équipe lors du développement de cette application.

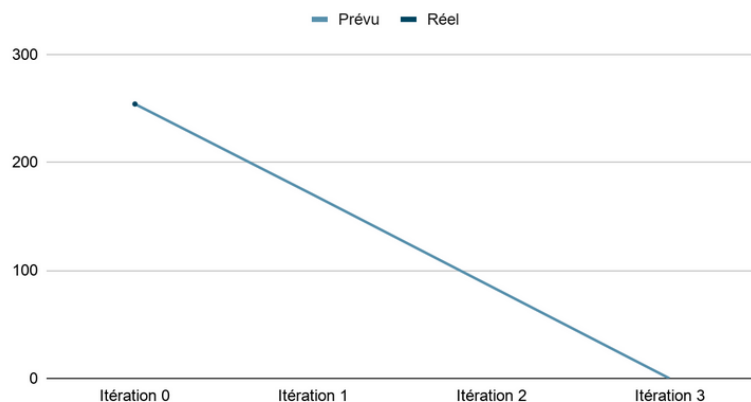
Lien github : <https://github.com/Voltariuss/PickupAndDelivery>

## Burndown Charts

### Itération 0 - Estimation Initiale

- 3 Itérations
- 239 points
- 8h / Personnes / Itérations
- 80 points / Itérations
- 12 points / Personnes / Itérations

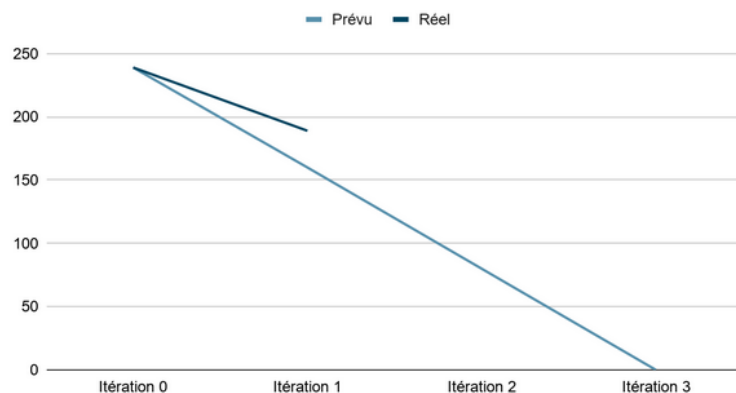
Burndown



### Itération 1

- Estimé : 80
- Réalisé : 50
- Restant prévu : 160
- Restant réel : 189

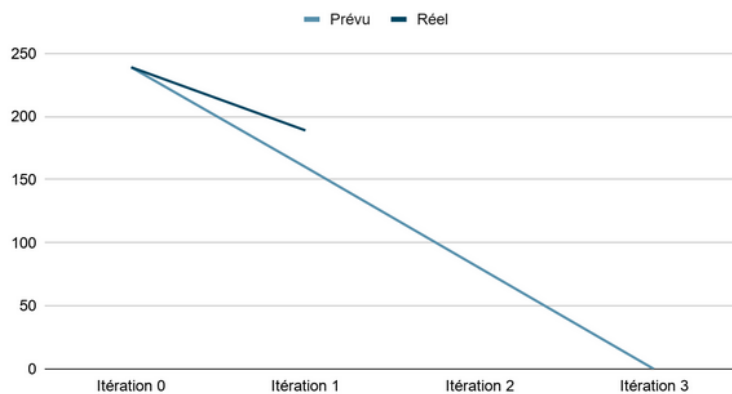
Burndown



### Itération 2

- Estimé : 83
- Réalisé : 107
- Restant prévu : 83
- Restant réel : 92

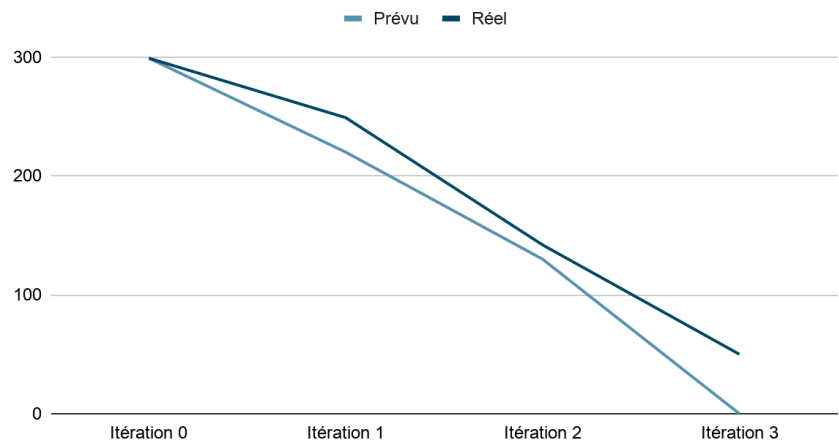
Burndown



### Itération 3

- Estimé : 130
- Réalisé : 98
- Restant prévu : 0
- Restant réel : 50

BurnDown



## Rétrospectives

### Itération 1

+	-	↗
Bonne répartition des tâches	Mauvaise estimation du temps pour préparer la livraison	Mieux estimer nos capacités à faire des US
Architecture du projet bien mise en place	Temps d'apprentissage des techno sous-estimé	Mieux prendre en compte la phase de test
Bonne dynamique de groupe	Temps de test sous-estimé	Transfert de compétences pour faciliter le temps d'apprentissage des collègues
Intégration rapide		Mieux définir nos engagements de livraison

### Itération 2

+	-	↗
Beaucoup de points US réalisés	Jar très long à créer et non fonctionnel sous Linux et Mac	Se concentrer sur les livrables afin de préparer la fin du projet
Bonne répartition du travail	En retard sur les spécifications	Acheter du champagne pour fêter la fin du projet
Bon système d'intégration continue		Ajouter une couverture de test
		Plus communiquer avec le PO durant le sprint (ex si tâche finis en avance)

### Itération 3

+	-	↗
Réalisation du projet satisfaisante	Rédactions de livrables	Allouer plus de ressource dans la rédaction de livrable
Bonne répartition d'équipe	Import XML n'a pas de test unitaires	
Bonne dynamique / bonne productivité		
Meilleure communication avec la PO		

## Risques rencontrés

### Travail en équipe

Le travail en équipe peut être un risque si on se répartit mal les tâches entre nous.

### Algorithme complexe et long

Les algorithmes pour calculer les chemins les plus rapides et le meilleur trajet sont complexes et longs. Dans un premier temps, ne pas réussir à l'implémenter est un risque, en effet sans celui-ci, l'implémentation de l'application serait impossible. De plus, il faut savoir les implémenter efficacement afin que les temps de calculs soient d'une longueur satisfaisante.

### Nouvelle technologies

L'utilisation de nouvelles technologies pour nous tels que JavaFx comporte toujours un risque que nous mettons trop de temps à s'adapter à la technologie et perdons trop de temps sur le développement du projet.