

# ***Burnup Chart projet « Bataille Navale » :***

Ce document est conçu pour construire le Burnup chart du projet « Bataille Navale ».

## **Etape 1 : Backlog Produit**

Dans cette section, nous allons établir la liste des taches à accomplir dans un Backlog produit.

Les taches qu'il faut faire pour réaliser le projet sont les suivantes :

1. Programmation de la classe "Armes", qui représente une arme de manière générale.
2. Implémentation des armes : Lance-missiles antisurface, Lance-missiles anti-air et Lance-torpilles.
3. Programmation de la classe "vaisseau", qui représente un vaisseau de manière générale.
4. Le test unitaire pour valider l'implémentation des armes
5. Implémentation des vaisseaux : Cruiser, Submarine, Fregate, Destroyer et Aircraft
6. Implémentation de la méthode `go_to(x, y, z)`
7. Le test unitaire de la méthode `go_to(x, y, z)` pour valider les coordonnées de déplacement de chaque vaisseau
8. Implémentation de la méthode `fire_at(x, y, z)`
9. Programmer le code final de champ de bataille

## **Etape 2 : Estimation de la durée du projet en story point**

Dans cette étape, nous avons défini le temps nécessaire à l'accomplissement de toutes les tâches du Backlog. Les résultats sont montrés dans le tableau suivant :

	Tâches	Durée Estimée en story point
1.	Programmation de la classe "Armes", qui représente une arme de manière générale.	13
2.	Implémentation des armes : Lance-missiles antisurface, Lance-missiles anti-air et Lance-torpilles.	8
3.	Programmation de la classe "vaisseau", qui représente un vaisseau de manière générale.	13
4.	Le test unitaire pour valider l'implémentation de la méthode <code>fire_at(x,y,z)</code>	5
5.	Implémentation des vaisseaux : Cruiser, Submarine, Fregate, Destroyer et Aircraft	8
6.	Implémentation de la méthode <code>go_to(x, y, z)</code>	8

7.	Le test unitaire de la méthode go_to(x, y, z) pour valider les coordonnées de déplacement de chaque vaisseau	5
8.	Implémentation de la méthode fire_at(x, y, z)	8
9.	Programmer le code final de champ de bataille	21

Figure 1 Durée Estimée des tâches en story point

### Etape 3 : Détermination de la charge du travail réalisé après chaque Sprint

Au fur et à mesure de l'avancement du projet, nous avons déterminé la charge du travail réalisé après chaque sprint. Les résultats sont rassemblés dans le tableau suivant :

	Tâches	Charge de travail réalisée
1.	Programmation de la classe "Armes", qui représente une arme de manière générale.	10
2.	Implémentation des armes : Lance-missiles antisurface, Lance-missiles anti-air et Lance-torpilles.	9
3.	Programmation de la classe "vaisseau", qui représente un vaisseau de manière générale.	10
4.	Le test unitaire pour valider l'implémentation de la méthode fire_at(x,y,z)	5
5.	Implémentation des vaisseaux : Cruiser, Submarine, Fregate, Destroyer et Aircraft	8
6.	Implémentation de la méthode go_to(x, y, z)	5
7.	Le test unitaire de la méthode go_to(x, y, z) pour valider les coordonnées de déplacement de chaque vaisseau	5
8.	Implémentation de la méthode fire_at(x, y, z)	8
9.	Programmer le code final de champ de bataille	25

Figure 2 Charge de travail réalisé

### Etape 4 : Construction de Burnup chart avec un tableau Excel

Maintenant que nous disposons des informations nécessaires pour construire le burnup chart. A l'aide d'une feuille de calcul Excel, on crée deux colonnes qui correspondent aux unités de temps et au travail effectué. Après visualisation, on trouve le graphique suivant :

Sprint	Durée Estimée	Durée Estimée cumulé	Charge de travail réalisée
1	13	13	10
2	8	21	19
3	13	34	29
4	5	39	34
5	8	47	42
6	8	55	47
7	5	60	52
8	8	68	60
9	21	89	85

Figure 3 Données du projet

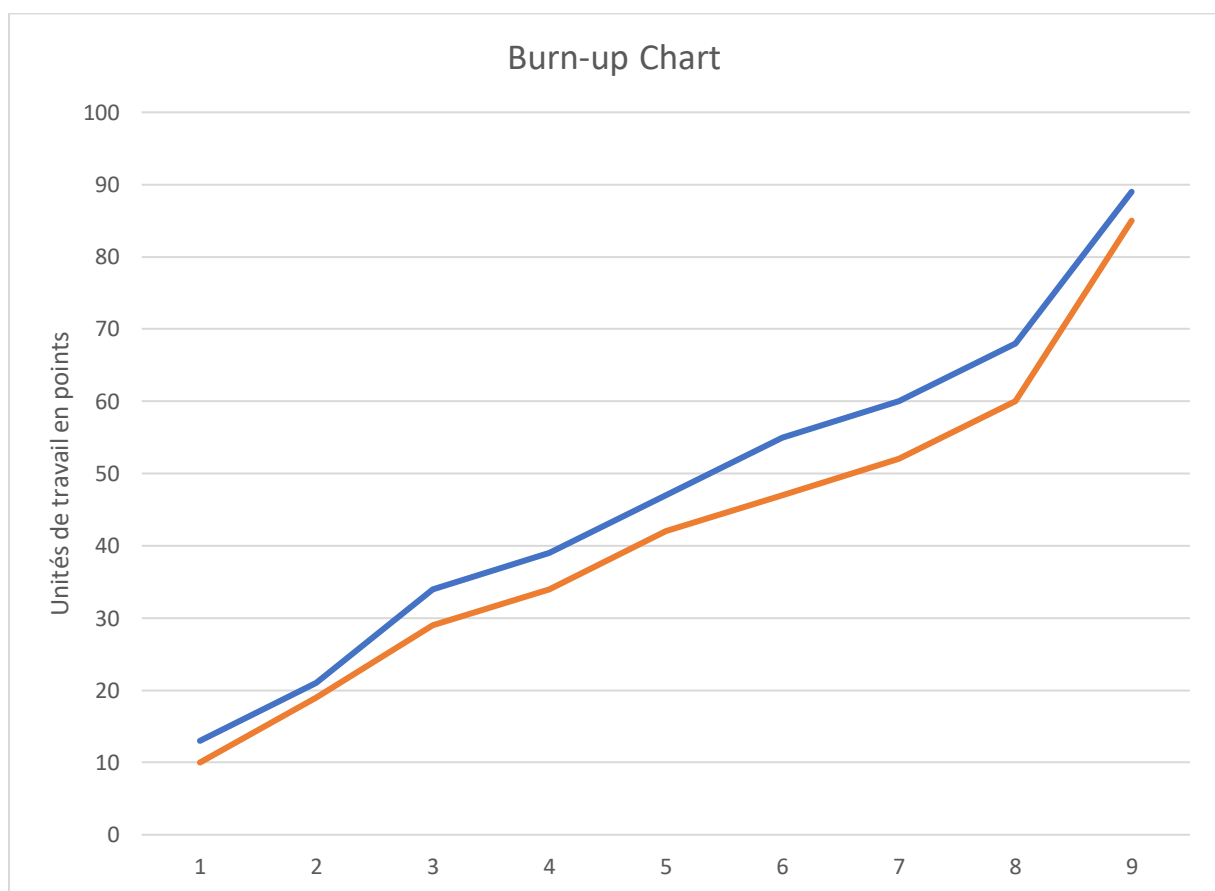


Figure 4 Burnup Chart du projet

Dans ce graphe :

- L'axe horizontal indique les unités de temps en sprint
- L'axe vertical montre les unités de travail en points
- La ligne rouge représente les travaux réalisés jusqu'à présent (avancement du projet)
- La ligne bleue représente la quantité totale de travail nécessaire pour atteindre l'objectif du sprint

### **Analyse des courbes :**

Comme son nom l'indique, on remarque que les deux courbes obtenues sont croissantes. Ainsi, la distance entre les deux courbes est petite, ce qui montre que nous avons atteint les objectifs visés et que le projet est bien géré.