

Projet Avancé Équipe 006

Parties prenantes	Nom
MOE : Professeur	Mr. VINCENT DUGAT
MOA1 : Chef de projet + Développeur	GIHALU RUSSEL
MOA2 : Développeur	DUNANT TONFACK
MOA3 : Développeur	WASSIM GAHA

Contexte du client:

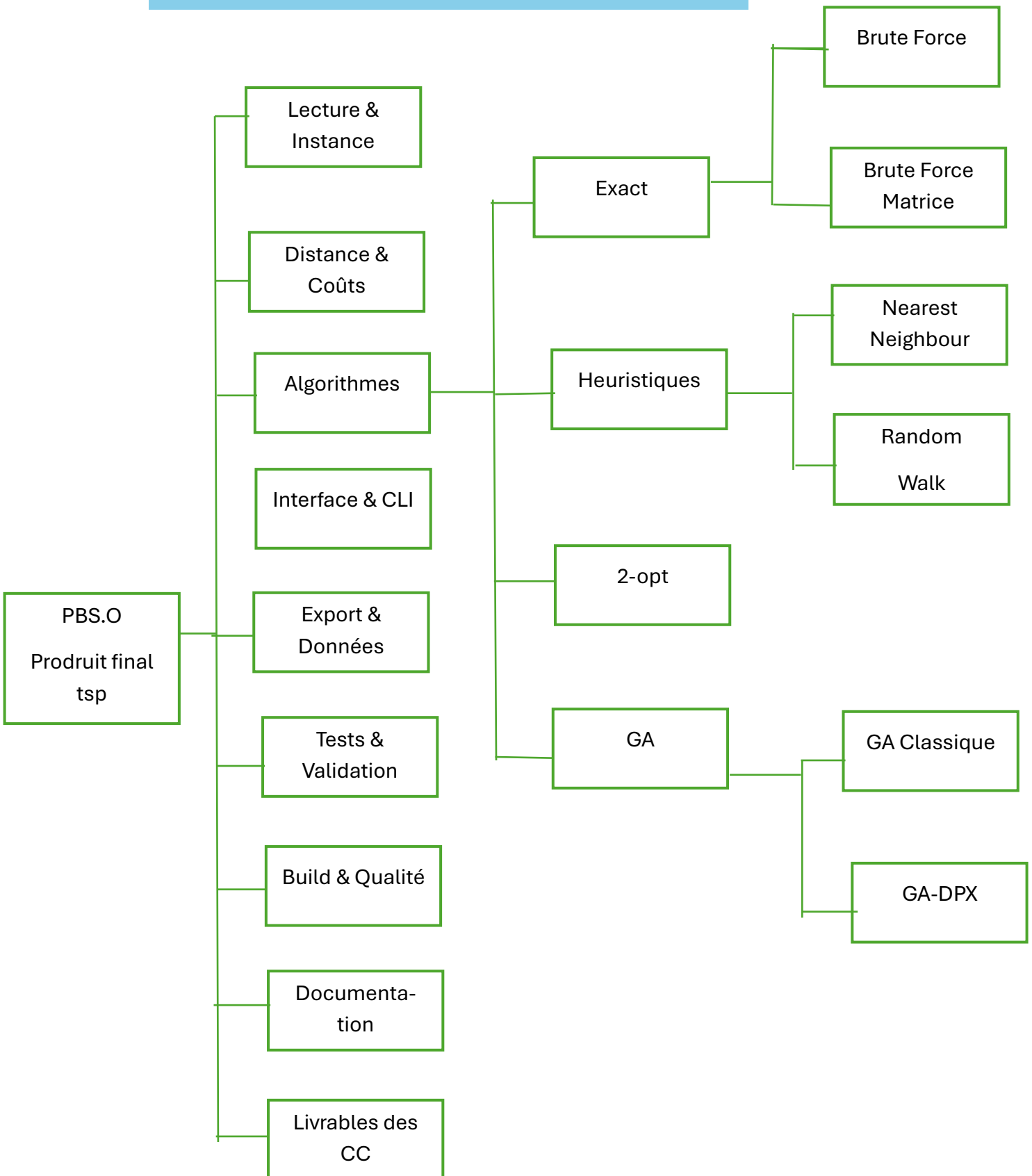
Le projet demandé par Mr Vincent Dugat consiste à développer en langage C un ensemble d'algorithmes pour résoudre le problème du voyageur de commerce (TSP). Ce problème d'optimisation combinatoire illustre plusieurs approches : exhaustive, heuristique et évolutionnaire. L'objectif est d'appliquer les compétences en algorithmique, programmation en C, modularité, gestion de projet et tests, en suivant une méthodologie structurée.

Capture du besoin

Categorie	ID d'exigence	Titre Exigence	Criticité (Obligatoire/ Souhaitée/ exemple)	Description Exigence	Methode d'acceptation	Explication
Developpement Projet	DP001	Parsing TSPLIB	O	Le programme doit lire et interpréter les fichiers .tsp (TSPLIB95) en extrayant les champs obligatoires (DIMENSION, NODE_COORD_SECTION, EDGE_WEIGHT_TYPE).	Test fonctionnel sur att10.tsp, att15.tsp	Base indispensable pour exécuter les algorithmes.
Developpement Projet	DP002	Calcul des distances	O	Le programme doit calculer les distances entre villes selon la métrique définie dans le fichier (EUC_2D, ATT, GEO).	Vérification des résultats sur cas simples et validation avec solutions connues	Garantit la cohérence des algorithmes.
Developpement Projet	DP003	Algorithme brute force (BF)	O	Test sur att10.tsp (résultat = optimum connu).		Sert de référence pour les petites instances.
Developpement Projet	DP004	Algorithme brute force avec matrice (BFM)	O	Le programme doit calculer l'optimum en utilisant une matrice de distances.	Comparaison résultats avec DP003.	Variante optimisée en mémoire.
Developpement Projet	DP005	Algorithme plus proche voisin (NN)	O	Le programme doit générer une tournée valide en partant d'un sommet et en choisissant à chaque étape le voisin le plus proche.	Vérification tournée valide + comparaison avec optimum connu (att15.tsp).	Heuristique rapide.
Developpement Projet	DP006	Random Walk (RW)	O	Le programme doit générer une tournée aléatoire valide en visitant toutes les villes.	Test sur att15.tsp (validité tournée).	gSert de base pour 2-opt.
Developpement Projet	DP007	Optimisation locale 2-opt	O	Le programme doit améliorer une tournée (NN ou RW) par la méthode 2-opt pour réduire les croisements.	Vérification que le coût de la tournée est réduit par rapport à la tournée initiale.	Méthode d'amélioration.
Developpement Projet	DP008	Algorithme génétique (GA)	O	Le programme doit implémenter un algorithme génétique avec paramètres (population, générations, taux mutation) choisi par les membres du groupe.	Test convergence sur att48.tsp avec affichage résultats CSV.	Méta-heuristique.
Developpement Projet	DP009	Algorithme génétique DPX (GADPX)	O	Le programme doit implémenter une variante GA avec croisement DPX.	Test sur att48.tsp et comparaison avec GA classique.	Introduit une diversité accrue.
Developpement Projet	DP010	Mode ALL	O	Le programme doit exécuter toutes les méthodes disponibles et générer un fichier CSV consolidé.	Vérification sur att15.tsp (résultats multiples).	Permet la comparaison globale.
Developpement Projet	DP011	Interface CLI	O	Le programme doit être utilisable en CLI (Command Line Interface) avec les options -f, -m, -o, -h.	Exécution ./tsp -h et test erreurs si options manquantes.	Interface utilisateur imposée par le cahier des charges.
Developpement Projet	DP012	Export CSV	O	Le programme doit générer un fichier CSV contenant : Instance, Méthode, Temps, Longueur, Tournée.	Vérification fichier généré et conformité format.	Livrable attendu.

Gestion de projet	GP001	CC1 - Cahier des charges	O	L'équipe doit produire un cahier des charges et une planification initiale.	Remise document CC1.	Premier jalon validant la capture du besoin.
Gestion de projet	GP002	CC2 - Prototype partiel	O	L'équipe doit livrer parsing, distances, BF, NN et un premier export CSV.	Démonstration lors de la séance CC2.	Vérifie la progression.
Gestion de projet	GP003	CC3 - Algorithmes heuristiques	O	L'équipe doit livrer RW, 2-opt, GA et rapport comparatif.	Démonstration lors de la séance CC3.	Montre l'évolution du projet.
Gestion de projet	GP004	CC4 - Recette finale	O	L'équipe doit livrer l'exécutable final avec toutes les méthodes, le Makefile et la documentation.	Soutenance + recette CC4.	Livrable final.
Gestion de projet	GP005	Qualité du code	O	Le code doit être clair, commenté et modulaire (Makefile, séparation modules).	Revue de code par l'enseignant.	Garantit la maintenabilité.
Gestion de projet	GP006	Tests & validation	O	L'équipe doit définir et exécuter un plan de tests (att10, att15, att48).	Vérification résultats.	Assure fiabilité du logiciel.
Gestion de projet	GP007	Organisation équipe	O	Chaque membre doit assumer ses responsabilités	Évaluation lors des jalons.	Répartition équilibrée des tâches.

PBS (Product Breakdown Structure)



WBS(Work Breakdown Structure)

Phase 1 – CC1 : Cadrage & conception (15–28 septembre 2025)

- WBS.1.1 Capture du besoin & cahier des charges.
- WBS.1.2 PBS/WBS/OBS & planning.
- WBS.1.3 Répartition des tâches et organisation de l'équipe.

Phase 2 – CC2 : Socle & premiers algorithmes (15–26 octobre 2025)

- WBS.2.1 Parsing fichiers TSPLIB.
- WBS.2.2 Calcul des distances (EUC_2D, ATT, GEO).
- WBS.2.3 Implémentation brute force (bf) et brute force avec matrice (bfm).
- WBS.2.4 Implémentation plus proche voisin (nn).
- WBS.2.5 Interface CLI (première version).
- WBS.2.6 Export CSV (première version).
- WBS.2.7 Jeux de tests att10, att15.

Phase 3 – CC3 : Heuristiques & GA (15–23 novembre 2025)

- WBS.3.1 Implémentation Random Walk (rw).
- WBS.3.2 Implémentation 2-opt (nn/rw).
- WBS.3.3 Implémentation Algorithme Génétique (GA).
- WBS.3.4 Export CSV (version enrichie).
- WBS.3.5 Tests sur att48 + comparaison avec optimum.

Phase 4 – CC4 : Finalisation & recette (10–21 décembre 2025)

- WBS.4.1 Implémentation GA-DPX.
- WBS.4.2 Implémentation mode ALL.
- WBS.4.3 Makefile & qualité (organisation code, commentaires, modularité).
- WBS.4.4 Tests de scalabilité (instances plus grandes).
- WBS.4.5 Recette finale & documentation utilisateur (README).

OBS(Organisation Breakdown Structure)

Organisation de l'équipe

Notre projet sera réalisé par une équipe composée de trois développeurs :

Responsable de projet : Russel Gihalu

- Développeur Dunant Tonfack (DD)
- Développeur Russel Gihalu (DR)
- Développeur Wassim Gaha (DW)

Le travail sera réparti de manière équitable, chaque développeur se voyant attribuer des tâches de programmation précises à chaque étape (CC1 à CC4). La répartition sera proportionnelle afin d'assurer un engagement équilibré et une contribution homogène de tous les membres de l'équipe.

Planning

ID de la tâche	Nom de la tâche	Début initial prévu	Fin initial prévue	Prédécesseurs	Noms ressources	Travail initial prévu
1	Projet	15/09/2025	21/12/2025			21hj
2	Cadrage et conception	15/09/2025	28/09/2025			3hj
3	Capture du besoin & cahier des charges.	15/09/2025	21/09/2025	-	RP[50%] , DD[25%] ,DW[25%]	1hj
4	PBS/WBS/OBS & planning.	24/09/2025	28/09/2025	3	RP[25%] , DD[50%] ,DW[25%]	2hj
5	Socle & Premiers Algorithmes	15/10/2025	26/10/2025			7,4hj
6	Parsing fichiers TSPLIB.	15/10/2025	15/10/2025	-	DR[33%] , DD[33%] ,DW[33%]	4hh
7	Calcul des distances (EUC_2D, ATT, GEO).	16/10/2025	17/10/2025	6	DR[33%] , DD[33%] ,DW[33%]	2hj
8	Implémentation brute force (bf) et brute force avec matrice (bfm).	18/10/2025	21/10/2025	7	DR[33%] , DD[33%] ,DW[33%]	2hj
9	Implémentation plus proche voisin (nn).	22/10/2025	24/10/2025	7	DR[33%] , DD[33%] ,DW[33%]	1hj
10	Interface CLI (première version).	25/10/2025	25/10/2025	7 ; 8 ; 9	DR[50%] , DD[50%]	3hh
11	Export CSV (première version).	26/10/2025	26/10/2025	8 ; 9 ; 10	DW[100%]	3hh
12	Jeux de tests att10, att15.	26/10/2025	26/10/2025	7 ; 8 ; 9 ; 10	DR[33%] , DD[33%] ,DW[33%]	2hh
13	Heuristiques & GA	15/11/2025	23/11/2025			5,4hj
14	Implémentation Random Walk (rw).	15/11/2025	16/11/2025	7	DR[33%] , DD[33%] ,DW[33%]	1hj
15	Implémentation 2-opt (nn/rw).	16/11/2025	18/11/2025	7 ; 9 ; 14	DR[33%] , DD[33%] ,DW[33%]	6hh
16	Implémentation Algorithme Génétique (GA).	19/11/2025	20/11/2025	7	DR[33%] , DD[33%] ,DW[33%]	2hj
17	Export CSV (version enrichie).	21/11/2025	21/11/2025	10 ; 14 ; 15 ; 16	DW[100%]	3hh
18	Tests sur att48 + comparaison avec optimum.	22/11/2025	22/11/2025	14 ; 15 ; 16	DR[33%] , DD[33%] ,DW[33%]	3hh
19	Finalisation et recette	10/12/2025	21/12/2025			5,2hj
20	Implémentation GA-DPX.	10/12/2025	11/10/2025	7 ; 16	DR[33%] , DD[33%] ,DW[33%]	1hj
21	Implémentation mode ALL.	12/12/2025	13/10/2025	8 ; 9 ; 10 ; 14 ; 15 ; 16 ; 20	DR[33%] , DD[33%] ,DW[33%]	1hj
22	Makefile & qualité (organisation code, commentaires, modularité).	12/12/2025	13/10/2025	-	DR[33%] , DD[33%] ,DW[33%]	6hh
23	Tests de scalabilité (instances plus grandes).	15/12/2025	16/10/2025	-	DR[33%] , DD[33%] ,DW[33%]	1hj
24	Recette finale & documentation utilisateur (README).	18/12/2025	20/10/2025	6 jusqu'à 23	DR[33%] , DD[33%] ,DW[33%]	1hj

Legende:

hh: Un humain heure

hj: Un humain jour

1 jour: 5 heures de temps réelles.

DT : Développeur Dunant

DR : Développeur Russel

DW : Développeur Wassim