Lexique utilisé :
aa → acides aminés
Repeat → sous-unité d’un domaine bêta-propeller (= feature)
Blade → sous-unité d’un domaine bêta-propeller (= repeat)
Classe → désigne un ensemble de blades portant le même numéro de repeat (1, 2, 3 ...)

Cahier de laboratoire :
    Format :
    {**AAAA-MM-DD :**
    travail exécuté par : <"noms">
    "contenu"
    <figures/images>
    }

Nous travaillons sur les bêta-propellers. Ces protéines sont constituées d’éléments répétés qui s’agencent sous forme de donut composé de plusieurs blades. Chaque élément répété forme un blade et possède une extension qui interagit avec le blade précédent.

Figure 1 : ?

2024-10-19 :
Travail exécuté par : MOSER Mathilda
Production des premiers alignements à partir des séquences fasta des WD repeats, des WD repeats étendus de 10 aa de chaque coté et des WD repeats étendus de 20 aa de chaque coté. Les fichiers de sortie sont réorganisés.
Ces alignements ont été obtenus grâce à la version web de clustalΩ disponible à cette adresse : " https://www.ebi.ac.uk/jdispatcher/msa/clustalo " . Les paramètres d'entrée ont été laissés par défaut sur le site sauf pour le format de sortie qui à été changé vers le format FASTA.

(PARAMETRES UTILISEES :
  - OUTPUT FORMAT : Pearson/FASTA
  - DEALIGN INPUT : NO
  - MBED-LIKE CLUSTERING GUIDE-TREE : yes
  - MBED-LIKE CLUSTERING ITERATION : yes
  - COMBINED ITERATIONS : default(0)
  - MAX GUIDE TREE : default
  - MAX HMM ITERATIONS : default
  - ORDER : aligned
  - DISTANCE MATRIX : no
  - OUTPUT GUIDE TREE : yes)

Les résultats ont été obtenus au format FASTA et ont été visualisés en utilisant le programme Jalview v2.11.4.0.
À cette étape du projet, aucune conclusion ni correction n'a été apporté vis-à-vis de ces alignements.

2024-10-29
Travail exécuté par : WITTENMEYER Guillaume
Pour comparer avec avec l'alignement de clustalΩ, nous avons décidé d’utiliser en complément la génération d'alignement de ces séquences grâce au programme MAFFT via l'interface web " https://usegalaxy.eu " avec l'historique accessible ici : " https://usegalaxy.eu/u/encelade/h/wdrepeats "

(PARAMETRES UTILISES :
 - Type of sequences : Amino acids
 - Type of scoring matrix : BLOSUM
 - Coefficient of the BLOSUM matrix : 62
 - Configure gap costs : Use default values
 - Reorder output : Yes/No (générer un de chaque)
 - Output format : FASTA)

Les résultats ont été enregistrés au format FASTA avec le PATH: " ~/common/data/1_intermediate/MAFFT_alignment "
Les résultats ont été obtenus au format FASTA et on été visualisés en utilisant le programme Jalview v2.11.4.0.

2024-10-29
Travail exécuté par : WITTENMEYER Guillaume, MOSER Mathilda, KESHAVARZ-NAJAFI Mohsen
Production de nouveaux alignements en utilisant MAFFT. Nous avons généré, pour chaque fichier d'entrée, deux fichiers de sorties. Le premier fichier de sortie possèdent l'ordre du fichier d'entrée. L'ordre des séquences a été réorganisé automatiquement pour le deuxième fichier.
Ces étapes ont été réalisées à partir des séquences fasta des WD repeats, des WD repeats étendus de 10 aa de chaque coté et des WD repeats étendus de 20 aa de chaque coté.

Les résultats ont été obtenus au format FASTA et ont été visualisés en utilisant le programme Jalview v2.11.4.0.

2024-10-29
Travail exécuté par : WITTENMEYER Guillaume, MOSER Mathilda, KESHAVARZ-NAJAFI Mohsen
A partir des premiers alignements réalisés avec MAFFT, nous avons pu extraire plusieurs positions intéressantes en nous basant dans un premier temps sur les alignements des séquences sans extensions (~/common/data/1_intermediate/MAFFT_alignment). Tous d'abord nous remarquons la présence de beaucoup de positions à tendance apolaire (généralement des aa aliphatiques et parfois aromatiques). Les motifs WD-repeat sont généralement caractérisés par un dipeptide GH en début de séquence et d'un dipeptide WD en fin de séquence. Ces deux motifs semblent être présent dans tous nos alignements, et c'est ce que nous souhaitons vérifier.
Le tableau suivant récapitule toute les positions qui ont été trouvées. L'analyse a été faite par visualisation de l'alignement obtenu avec MAFFT du 2024-10-29 (format FATSA) grâce à l'application Jalview v2.11.4.1 (mais aussi avec la v.2.11.4.0). Les nombres après chaque aa indique la proportion de l'acide aminé à cette position (en pourcentage) et sont issus de Jalview.

Figure 2 : tableau récapitulatifs des positions sélectionnées.

Positions	aa dominants	Propriétés physico-chimique
71	G (23,36)	/
72	H (42, 19)	/
128	V (39,37), I (22,68), L (9,76), A (8,96)	Position hydrophobe (Aliphatique)
161	L (27,08), V (23,93), I (12,58)	Position hydrophobe (Aliphatique)
204	F (28,11), W (18,58), L (9,39), V (6,38), Y (5,96), I (4,41)	Position hydrophobe (Aliphatique, Aromatique)
215	S (25,11)	/
255	P (36,32)	/
300	G (27,92)	/
339	L (39,84), I (16,28), V (13,89), F (8,59)	Position hydrophobe (Aliphatique, Aromatique)
359	A (26,54), V (21,07), L (16,49), I (8,17), F (5,58)	Position hydrophobe (Aliphatique, Aromatique)

374	S (30,22), T (24,96), A (12,29)	Petite molécule à tendance polaire non ionisable
386	G (40,78), A (16,61)	Petite molécule apolaire
416	S (32,00), G (15,95)	/
444	D (52,84)	Position acide
469	G (33,88)	Petite molécule
482	T (20,51), S (12,39)	Petite molécule à tendance apolaire non ionisable
491	V (32,80), I (25,95), L (20,27)	Position hydrophobe
509	R (20,60), K (17,46)	Position basique
528	L (26,04), V (23,60), I (22,99)	Position hydrophobe (Aliphatique)
560	W (56,36), Y (11,45), F (10,00)	Position hydrophobe (Aromatique)
572	D (37,35), N (11,12)	Position polaire à tendance ionisable
588	L (20,69), V (13,23), I (8,22)	Position hydrophobe (Aliphatique)

2024-11-19
Travail exécuté par : WITTENMEYER Guillaume
Suite à nos précédentes observations, pour améliorer nos alignements, nous avons pensé qu'il serait intéressant d'aligner chaque classe de repeats en utilisant MAFFT indépendamment les unes des autres pour voir s'il n'existe pas de features intéressantes à mettre en évidence pour chaque classe de repeats.
Suite à nos réunions avec le Groupe 3, nous avons aussi jugé qu'il était intéressant de de ne garder que les protéines ayant un nombre de repeats maximum de 7 (donc 1 seul domaine) pour faciliter les analyses (voir https://github.com/Hudego/PTU_Project_3.git). Pour rester cohérent avec notre étude, nous avons décidé de ne nous baser que sur les protéines de nos alignements ayant un maximum de 1 domaine de 7 repeats.

Pour effectuer cela nous avons écrit et utilisé un script python (~/common/scripts/filter.py) dans un environnement conda spécifique au projet en python=3.9 (~/common/scripts/wd_repeat.yml) dont les dépendances et librairies sont spécifiées dans le fichier README.md du projet.
Les fichiers fasta filtrés générés ont été sauvegardés dans des sous-dossiers séparés dans le dossier ~/common/data/0_raw/sequences_fasta/

Alignement de ces séquences grâce au programme MAFFT via l'interface web " https://usegalaxy.eu " avec l'historique accessible ici : " https://usegalaxy.eu/u/encelade/h/wdrepeats "

(PARAMETRES UTILISES :
 - Type of sequences : Amino acids
 - Type of scoring matrix : BLOSUM
 - Coefficient of the BLOSUM matrix : 62
 - Configure gap costs : Use default values
 - Reorder output : Yes
 - Output format : FASTA)

Les résultats ont été enregistrés au format FASTA avec le PATH: " /data/projet5/common/data/1_intermediate/MAFFT_alignment/" dans un sous-dossier

séparé pour chaque raw data.
Les résultats ont été obtenus au format FASTA et ont été visualisé en utilisant le programme Jalview v2.11.4.0.

2024-11-20
Travail exécuté par : MOSER Mathilda
Avec les alignements de la veille (réalisés par Guillaume), nous avons réalisés des weblogos pour essayer de trouver des positions intéressantes. Les weblogos sont disponible au format PDF à ce PATH:" ~/common/data/1_intermediate/weblogo/weblogo_full ". Sur ces weblogos, nous retrouvons bien les motifs GH en débuts de séquences (les GH en fins de séquences qui apparaissent sur les weblogos des séquences avec des extensions représentent les GH de début de séquences du repeat suivant) et WD. Nous constatons aussi la présence de beaucoup de positions hydrophobes. Les positions hydrophobes sont en noires.

Les weblogos ont étés générés grâce à l'outil en ligne Seq2Logo (https://services.healthtech.dtu.dk/services/Seq2Logo-2.0/).

2024-11-21
Travail exécuté par : KESHAVARZ-NAJAFI Mohsen
Suite à ces observation, nous avons décidé de manuellement effectuer un fichier excel qui recense toutes les observations de conservation pour ces séquences (~/common/data/1_intermediate/MAFFT_alignment/Conservation_classfiltered_alignment.xlsx).

Dans notre fichier Excel, nous avons essayé de noter toutes les conservations supérieures à 10 % observées dans les 16 fichiers d'alignements MAFFT, visualisés dans Jalview, pour les séquences WD et WD-10. Nous avons également marqué en couleurs les conservations ayant un pourcentage plus élevé ou celles qui se répétaient dans différents fichiers, afin d'avoir une vue d'ensemble plus claire. Selon nos observations, en plus des conservations dans les régions WD, une conservation H a été identifiée dans les parties initiales de chaque alignement. Dans les fichiers contenant 10 résidus d'extension, cette conservation H se répète également dans la partie finale ce qui correspond au début de la séquence du repeat suivant. De plus, juste avant chaque H, on peut observer une conservation G. Au milieu de tous les alignements, il y a également une conservation P qui pourrait appartenir à une boucle de liaison dans le repeat. Une autre conservation commune dans tous les alignements est celle des acides aminés (V, I, L), retrouvée dans différentes régions. On peut supposer que ces trois acides aminés, en raison de leurs propriétés physico-chimiques proches, se remplacent parfois entre eux. Cependant, leurs caractéristiques fondamentales d'être hydrophobes et non polaires sont préservées, ce qui pourrait être essentiel pour la stabilité de la structure de la protéine et sa fonction.

2024-11-21
Travail exécuté par : MOSER Mathilda
Les weblogos du 20/11 sont très confus et comportent toutes les positions de la séquence. Nous avons décider d'alléger les weblogos en ne gardant que les positions les plus conservés. Sur Jalview v2.11.4.1, nous avons sélectionné les positions qui possédait le plus d'occupancy. Le travail à été fait à la main. De ces weblogos (PATH = "~/common/data/1_intermediate/weblogo/weblogo_crop/manually/filtered "), nous pouvons constatés que le motif GH n'est pas toujours strictement conservé, de même pour le motif WD. Cependant nous pouvons observés un motifs récurent : GH-3*Hydophobe-Proline-Hydrophobe-G ou A-Acide-2*Apolaire-W-D. Nous observons aussi des différences entre chaque classe de repeats. Par exemple tous les repeats de «classes 5» semble ne pas posséder l'histidine qui est présente dans les autres.
En plus des logos obtenues par sélection manuelle, nous sommes en train de générer des logos obtenu grâce un algorithme de tri crée par Guillaume.

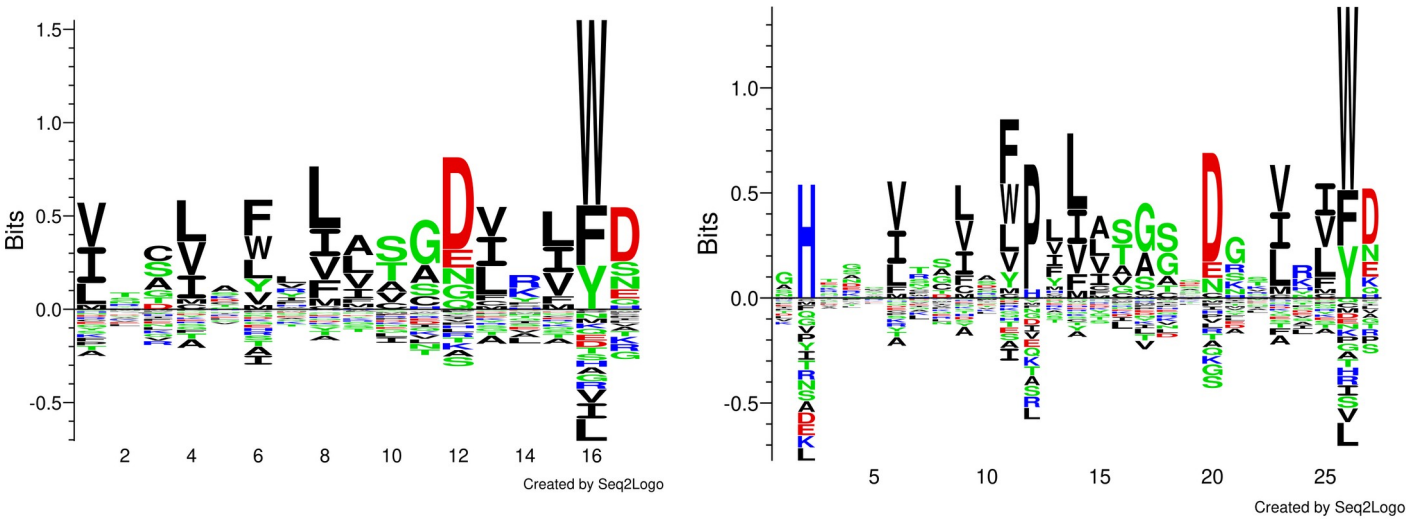

Figure 3 : Weblogos triés à la main. (A. classe 5, B. classes 6)
Les weblogos ont étés générés grâce à l'outil en ligne Seq2Logo (https://services.healthtech.dtu.dk/services/Seq2Logo-2.0/).

2024-11-18 à 2024-11-19
Travail exécuté par : WITTENMEYER Guillaume
Pour avoir une idée plus précise de ces weblogos, nous avons choisit de filtrer l'occupancy de chaque position dans ces alignements afin de pouvoir quantifier cette conservation dans l'alignement et éviter les biais humain.
Nous avons donc créer un script python permettant de créer ces nouveaux alignements filtrés en fonction de l'occupancy (~/common/scripts/occupancy.py).
Nous avons également choisi de filtrer ces différents alignements à plusieurs occupancy [0,6 ; 0,7 ; 0,8] pour les comparer. Les alignements filtrés de sorties sont sauvegardées dans des sous-dossiers '/occupancy/' pour chaque alignement intermédiaires contenus dans le dossier des alignements MAFFT ~/common/data/1_intermediate/MAFFT_alignment/.
Nous avons aussi codé dans ce script python une petite fonction permettant de rapidement visualiser des weblogos générés avec python en local (trop illisibles pour une étude poussée) (fonction se base sur un script test inutilisé dans le projet ~/common/scripts/gen_weblogo.py
Ces fichiers filtrés seront utilisés par Mathilda pour générer des logos plus visuels et compréhensibles en utilisant Seq2Logo (Les weblogos ont étés générés grâce à l'outil en ligne Seq2Logo (https://services.healthtech.dtu.dk/services/Seq2Logo-2.0/)).

Le script python a été exécuté dans l'environnement conda associé au projet ~/common/scripts/wd_repeat.yml

2024-11-19 à 2024-11-20
Travail exécuté par : WITTENMEYER Guillaume
Pour la suite du projet, nous avons trouvé intéressant d'effectuer une analyse structurale de ces alignements.

Pour cela, nous avons, à partir du fichier filtré de la classe 7 (~/common/data/1_intermediate/MAFFT_alignment/WD_sequence_filtered/MAFFT_WD_sequence_classfiltered_7.fasta) récupérer toute les structures pdb des protéines de notre alignement possédant 7 repeats au minimum et au maximum (soit 143 protéines sur les 252 disponibles dans l'étude (voir https://github.com/Hudego/PTU_Project_3.git)) grâce à un script python récupérant les structures sur AlphaFold v. ? ~/common/scripts/alphafold.py
Les structures récupérés sont stockées dans le dossier ~/common/data/0_raw/pdb/.
On remarque cependant que toutes les structures ne sont pas prédites par AlphaFold :

PDB ID
Q15751
Q6ZNJ1
Q6ZQQ6
Q99698

De plus, certaines structures possèdent plusieurs bêta-propellers :

PDB ID	Nbr of domaines prédits
O14040	3
O75717	2
P53621	2
P54198	1,5
Q9MBG6	2
Q9P2M3	1,5
Q96RY7	2

Ou alors possèdent de grandes régions mal prédites par AlphaFold comme Q55G07 :

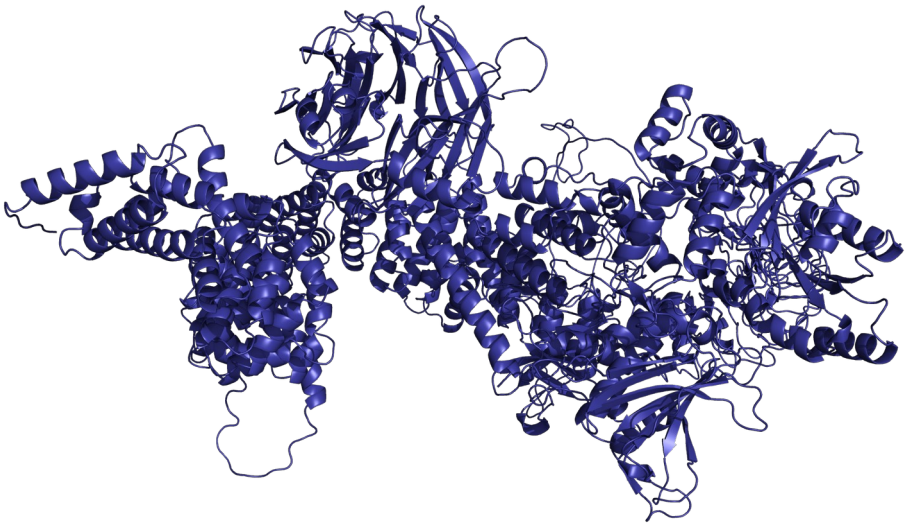

Figure 4: Visualisation Q55G07 dans PyMol

Ces structures représentent cependant les structures complète des domaines et nous nous intéressons aux différences entre chaque classe de repeats. Il faut donc décomposer ces structures en 1 structure par blade de chaque bêta-propeller.
Pour cela, nous avons codé un script python ~/common/scripts/cut_blades.py qui permet de récupérer les information de coordonnées et de longueur de chaque features de type repeat dans la séquence pour chaque protéine ~/common/data/0_raw/excel/WD_extracted_data.xlsx
Les fichier pdbs créés sont stockées dans des sous-dossiers pour chaques blades dans le dossier ~/common/data/1_intermediate/pdb_cut/

Le script python a été exécuté dans l'environnement conda associé au projet ~/common/scripts/wd_repeat.yml

2024-11-21 à 2024-11-23
Travail exécuté par : WITTENMEYER Guillaume
L'étape suivante est de pouvoir super-imposer ces structures de blades en fonction des classes dans PyMol.
Pour cela, nous avons codé un script python ~/common/scripts/superimpose.py qui permet de choisir la classe de repeat à super-imposer et qui super-impose ces structure dans une interface GUI de PyMol. Nous avons choisi de super-imposer avec la fonction super() de PyMol car les séquences des repeats étant souvent différentes, le fonction super() sera plus performante pour avoir des superpositions de structures cohérentes.

Cependant, on peut observer dans PyMol que toutes les super-impositions ne sont pas très efficaces comme la super-imposition du blade 6 de P57081 avec le blade 6 de A2RU52 :

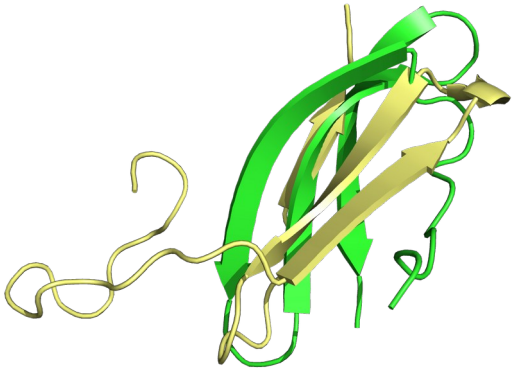

Figure 5: Super-imposition des blade 6 de P57081 et A2RU52

Nous avons donc eu l'idée de stocker grâce à ce même script les RMSD correspondant à chaque super-imposition ainsi que le nombre d'atomes pris en compte relatifs au calcul de ce RMSD pour pouvoir gérer des graphiques du nombre d'atomes pris en compte en fonction du log10(RMSD) dans des fichier txt pour chaque classe dans les sous dossiers de chaque cut per blade number dans dossier ~/common/data/1_intermediate/pdb_cut/
Pour cela, nous avons créer un script python ~/common/scripts/graphs.py qui génère ce graphiques et qui sont disponible dans ces même sous dossiers dans le dossier ~/common/data/1_intermediate/pdb_cut/

On obtient ce genre de graphique qui permet de voir la répartition des scores de RMSD en fonction du nombre d'atomes pris en compte dans la super-imposition et nous permets de potentiellement exclure les mauvaises super-imposition plus tard dans notre projet :

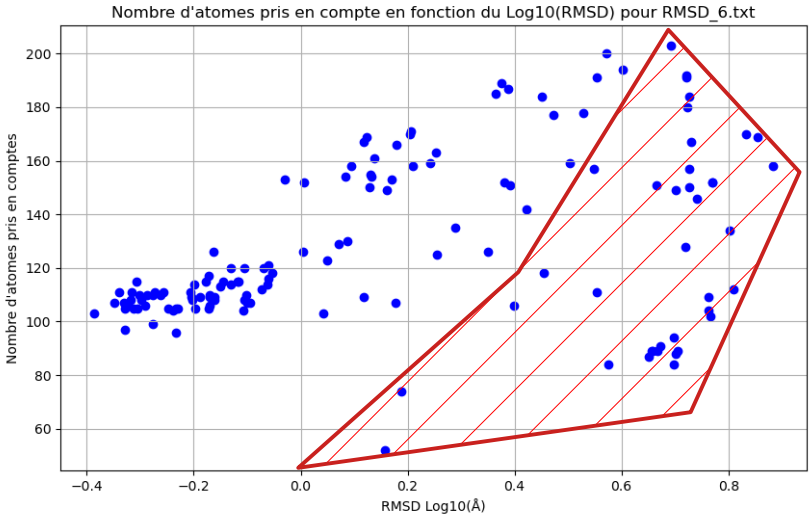

Figure 6: Graphique issu de graphs.py pour la classe 6

La zone hachurée en rouge représenterait les structures à exclure pour l'analyse.

2024-11-19 à 2024-11-20
Travail exécuté par : WITTENMEYER Guillaume
Suite aux analyses des alignements par blade effectuées par Mohsen, nous souhaitions pouvoir visualiser les structures dans d'autres logiciels de visualisation que PyMol. Pour cela, nous avons donc eu l'idée de sauvegarder les coordonnées des structures des blades super-imposées issus du script python ~/common/scripts/superimpose.py dans de nouveaux fichiers pdb pour pouvoir ouvrir et étudier ces structures dans d'autres logiciels de visualisation molécuaires tels que DiscoveryStudio.
Nous avons donc implémenté une nouvelle fonction python dans le script ~/common/scripts/superimpose.py pour effectuer cela et stocker ces fichiers dans des sous-dossiers pour chaque super-imposition de blades dans le dossier ~/common/data/1_intermediate/pdb_superstructure/

For detailed guidance or troubleshooting, please refer to the full project documentation or contact the project lead.