
Rapport du projet d'Innovation et de conception



Réalisé par :

Félix BOUTERAA
Mohamed OULAAFFART
Gaël SAGON

Encadrants :

Mme. POISSON
CAILLAULT
M. SMOCH

Table des matières

Introduction	3
A Contexte général du projet	4
A.1 La ligue des Hauts-de-France du triathlon	4
A.2 Description du processus de l'organisation des challenges	4
A.3 Les modalités du challenge	4
A.4 Expression des besoins fonctionnelles	7
A.5 Méthodologie de travail	7
A.6 Diagramme de Gantt	7
B Étude analytique du projet	10
B.1 Diagramme de packages	10
B.2 Les cas d'utilisation	10
B.3 Diagramme de classes	12
B.4 Le diagramme d'activités	16
C Réalisation du projet	18
C.1 Langages & outils utilisés	18
C.2 Les deux outils distincts du projet	21
C.3 Déploiement du projet	25
C.4 La documentation automatique	26
C.5 Schéma des tests	26

Table des figures

1	Les points attribués aux participants de la catégorie adultes selon leurs classements	5
2	Les points attribués aux participants de la catégorie jeunes selon leurs classements	5
3	Récapitulatif des modalités du challenge	6
4	Le diagramme de Gantt du projet	9
5	Le diagramme de packages du projet	10
6	Le diagramme de cas d'utilisation relatif à l'organisation d'une course.	11
7	Le diagramme de cas d'utilisation de la phase d'administration.	12
8	Le diagramme de cas d'utilisation de l'affichage des résultats pour les participants.	12
9	Le diagramme de classes du projet.	15
10	Le diagramme d'activités du projet	17
11	Le logo du R project	18
12	Le logo de Shiny	18
13	Le logo de GitHub	20
14	Le logo de LaTeX	20
15	Le logo de LaTeX	20
16	Aperçu de l'interface de classement - Onglet Course	21
17	Aperçu de l'interface de classement - Onglet Challenge	22
18	Aperçu de l'interface de classement - Onglet Téléchargement	22
19	Code du classement par clubs - Original	23
20	Code du classement par clubs - Nouveau	23
21	Aperçu de l'interface web affichant les résultats	24
22	Statistiques globales adultes	25
23	Statistiques globales jeunes	25

Introduction

Dans un contexte marqué par l'émergence du web, le besoin accru d'informatisation et d'automatisation des gestions, nous avons développé un site web pour permettre à la ligue des Hauts-de-France de mettre à jour les challenges organisés durant chaque année et aux participants de consulter leurs résultats. En effet, le long du processus de développement allant des premières phases de la définition des besoins fonctionnels du client jusqu'aux phases de création et de test, nous avons fixé trois lignes de conduite, à savoir :

1. Le site web doit satisfaire les besoins de ses utilisateurs et des administrateurs
2. Il doit répondre aux qualités requises en matière de maintenabilité et de portabilité

Afin de mettre la lumière sur les différentes démarches suivies pour l'aboutissement de ce travail, nous avons scindé ce rapport en trois grandes parties. Dans cette optique, la première sera réservée à l'illustration du contexte général du projet, tandis que la deuxième sera consacrée à l'étude analytique des différents processus de données et des phases de notre projet. Alors que la 3ème partie sera dédiée à la présentation des langages utilisés pour la réalisation du site web, de ses interfaces ainsi que des tests mis en œuvre pour s'assurer de son bon fonctionnement. Par ailleurs, il convient de préciser que ce rapport sera joint d'un guide d'utilisation qui sera mis à la disposition aux administrateurs de la ligue.

A Contexte général du projet

Cette partie traitera du contexte général de notre projet d'Innovation et de Conception, en mettant l'accent essentiellement sur une brève présentation du maître d'ouvrage et de ses exigences fonctionnelles. Également, la méthodologie de travail adoptée ainsi que la planification du projet seront mises en avant.

A.1 La ligue des Hauts-de-France du triathlon

La Ligue des Hauts-de-France de Triathlon est un organe décentralisé de la F.F.TRI. Elle est administrée par un Comité Directeur et un Bureau Directeur composés de membres élus en son sein. Pour fonctionner, la ligue Hauts-de-France, a constitué plusieurs commissions dont chacune est présidée par un membre du Comité Directeur et composées de membres de la ligue volontaires et choisis pour leurs compétences. Celle-ci n'a été créée que récemment, car elle fait suite à la fusion des régions Picardie et Nord-Pas-de-Calais. L'assemblée générale s'est alors tenue le 03 février 2018, un court article a été rédigé sur leur nouveau site web à l'adresse triathlonhdf.fr [1]

A.2 Description du processus de l'organisation des challenges

L'organisateur de la course envoie les informations de tous les participants ainsi que leurs classements à la ligue. Ces fichiers sont envoyés sous formats PDF, Word, Tableurs Excel voire même papier. Ces données sont envoyées à la ligue sans aucun traitement ou filtrage. Par conséquent, les administrateurs de la ligue ont pour première tâche de supprimer les participants qui ne sont pas inscrits au niveau de la ligue, et également de traiter le problème des homonymes d'une manière manuelle. Tâche qui demeure difficile et demande du temps. Une fois les données sont actualisées, les fichiers challenge sont mis à jour et les classements globaux sont établis. L'ensemble de ces opérations sont faites moyennant des fichiers Excel.

A.3 Les modalités du challenge

Les challenges organisés sont soumis à plusieurs règles qui sont répertoriées comme suit :

- Toutes les épreuves individuelles de cross duathlon, duathlon, aquathlon, cross triathlon et triathlon organisées par la Ligue NPdC, distance XS, S, M, L ou XL sont retenues pour le challenge.
- Les organisations s'inscrivant dans le circuit des épreuves du challenge doivent se soumettre au cahier des charges proposés par la Ligue.
- Seront retenues au challenge de l'année suivante uniquement les épreuves qui rempliront les contraintes et fourniront les résultats au format prévu au point 8 au plus tard dans la semaine suivant l'épreuve.

Attribution des points. Un classement individuel par catégorie et par sexe est réalisé à l'issue de chacune des épreuves. Seule la distance la plus longue accessible à la catégorie sera prise en compte dans l'obtention des points.

L'attribution des points se fait en fonction du classement par catégorie et sexe.

Les tableaux figurant ci-dessous récapitulent le nombre de points attribués selon le classement du participant :

1. Pour les adultes

Place	1	2	3	4	5	6	7	8	9	10	11	..
Points	300	250	220	200	190	180	170	160	150	145	140	

FIGURE 1 – Les points attribués aux participants de la catégorie adultes selon leurs classements

Au-delà de la 142ème place, 1 point est attribué à chacun des participants. Les points obtenus sont pondérés par un coefficient dépendant de la distance de l'épreuve :

- 0,5 pour un XS ;
- 0,75 pour un S ;
- 1 pour un M ;
- 1,5 pour un L ;
- 2 pour un XL ou XXL.

2. Pour les jeunes

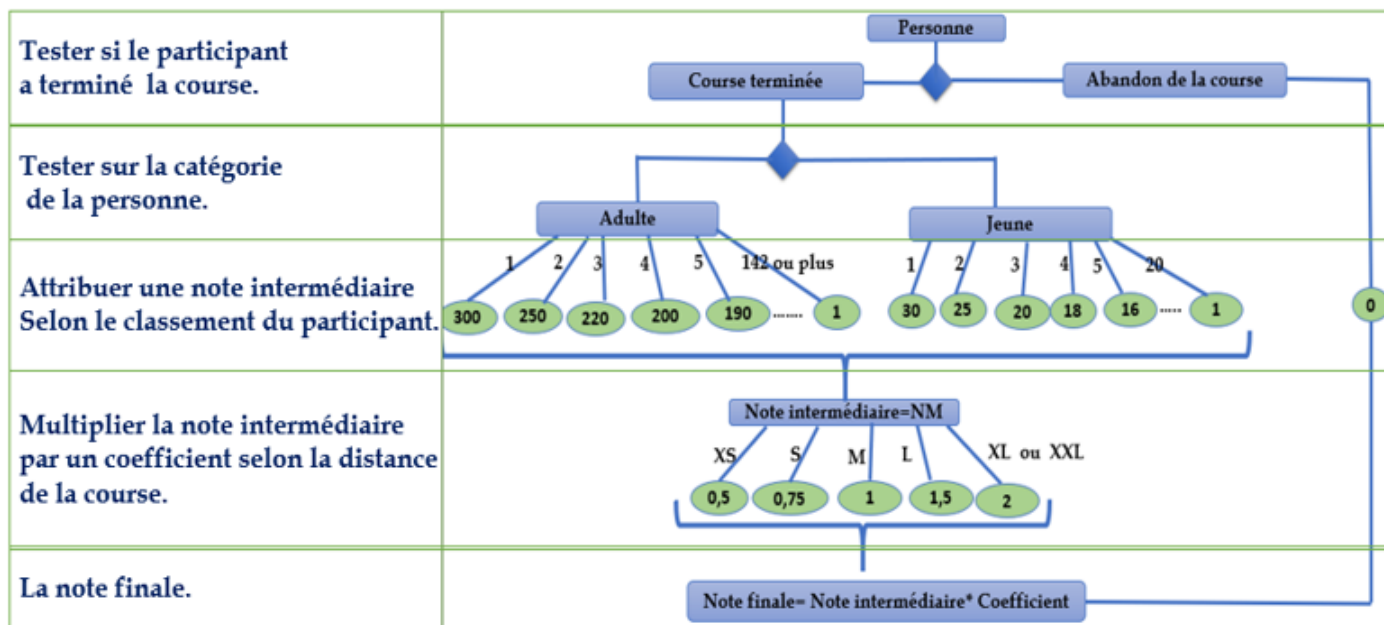


FIGURE 2 – Les points attribués aux participants de la catégorie jeunes selon leurs classements

Au-delà de la 20ème place, un point est attribué à chacun des participants. Les abandons et les absents ne marquent pas de point. L'ensemble des règles susmentionnées sont récapitulées dans le schéma suivant :

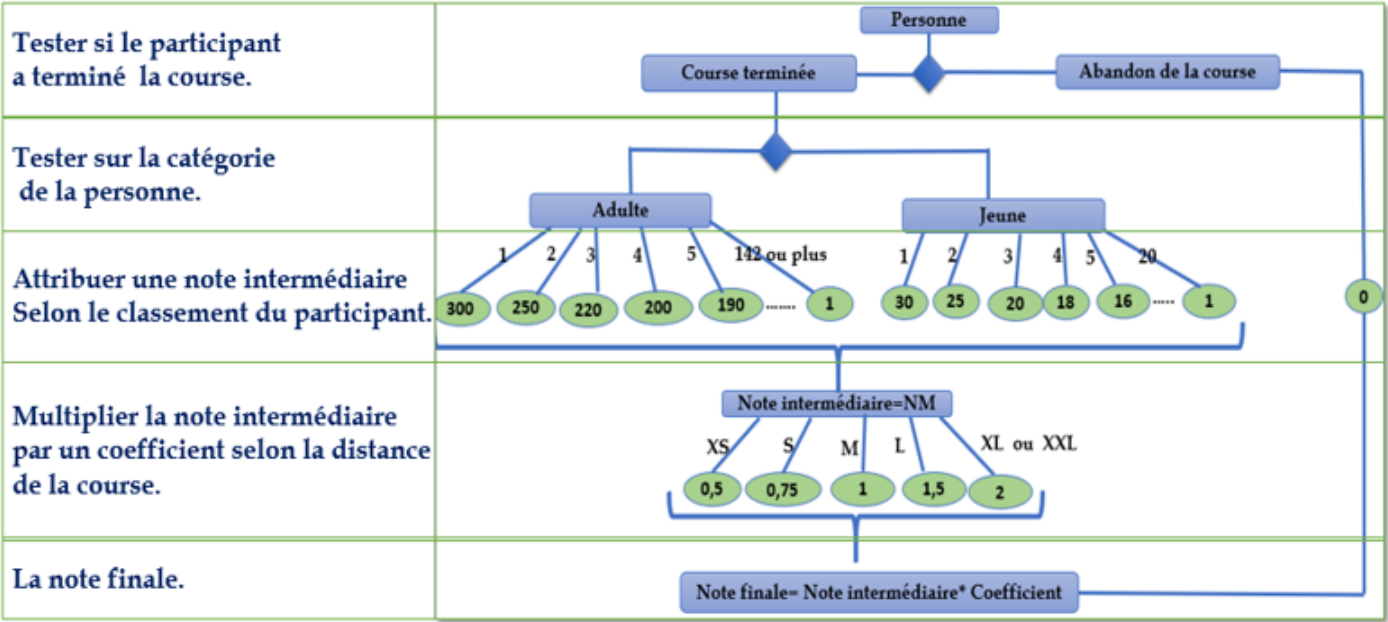


FIGURE 3 – Récapitulatif des modalités du challenge

A.4 Expression des besoins fonctionnelles

L'outil développé dans le cadre de ce projet, doit répondre aux fonctionnalités définies en concertation avec le client lors des réunions. Ainsi l'ensemble de ces fonctionnalités sont répertoriées comme suit :

- Supprimer les participants dans une courses et qui ne sont pas inscrits dans la ligue ;
- Pallier au problème des homonymes ;
- Création d'un challenge
- La mise à jour d'un challenge ;
- Affichage des résultats globaux des courses ;
- Afficher les statiques individuels avec les graphes correspondants ;
- Effectuer des recherches des participants par plusieurs critères ;
- Toutes ces opérations doivent être réalisées émanant une interface graphique reluisante.

A.5 Méthodologie de travail

Afin de garantir la réussite du projet, il a été décidé de recourir à la méthodes agile SCRUM. En effet, celle-ci consiste à découper le projet en boites de temps appelées sprints. Chaque sprint commence par une estimation suivie d'une planification opérationnelle. Le sprint se termine par une démonstration de ce qui a été achevé. Avant de démarrer un nouveau sprint, l'équipe réalise une rétrospective. Cette technique analyse le déroulement du sprint achevé, afin d'améliorer ses pratiques. Le flot de travail de l'équipe de développement est facilité par son auto-organisation. Pour mettre cette méthode en pratique. Une réunion est organisée chaque jeudi après-midi avec nos Encadrants pour discuter des avancements de la semaine en cours en de déterminer les tâches à effectuer durant la semaine à venir. Par ailleurs, l'ensemble des tâches à réaliser sont répartis sur les membres de l'équipe afin de converger les efforts pour la réussite du projet.

A.6 Diagramme de Gantt

La planification est parmi les phases d'avant-projet. Elle consiste à prévoir le déroulement du projet tout au long de ses phases. Grâce aux réunions tenues avec nos encadrants et le client, on a été éclairés sur les différentes étapes du projet. Ainsi le diagramme ci-après résume les différentes étapes ainsi que le temps qui leurs a été allouée.

Comme il est clairement illustré dans ce diagramme, le déroulement du projet s'est fait en plusieurs citées comme suit :

- La définition des besoins du le maître d'ouvrage ;
- Le choix du langage R et du Serveur Shiny pour l'implémentation du projet ;
- La formation su les langages à utiliser qui a duré le long du processus du développement du projet ;
- Implémentation des fonctionnalités souhaitées par le client en se basant le code R fourni ;
- Effectuer des tests nécessaire sur chaque fonctionnalité ;
- Déploiement de l'application sur serveur de test et réaliser le test d'acceptance sur l'ensemble des fonctionnalités ;
- La documentation du code : cette étape est cruciale pour faciliter la comprhension du code, l'utilité de chaque fonction et également pour faciliter les étapes futures de l'évolution du code.
- La rédaction du rapport qui est une étape faite en parallèle des autres.

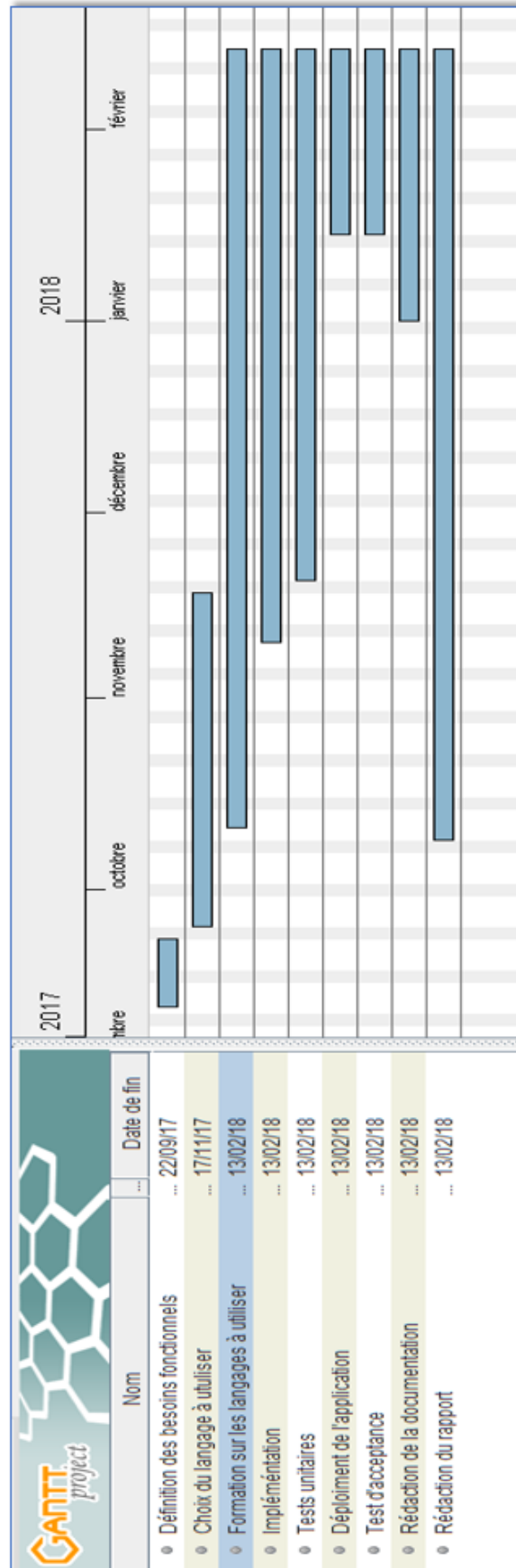


FIGURE 4 – Le diagramme de Gantt du projet

B Étude analytique du projet

Ce chapitre présentera les outils utilisés pour la réalisation de la conception de notre projet. Par ailleurs, il relate ses différentes étapes, ainsi que les diagrammes dégagés après chaque étape.

B.1 Diagramme de packages

Définition

Le diagramme de packages est un diagramme UML qui fournit une représentation graphique de haut niveau de l'organisation de l'application à réaliser.

Diagramme de packages du projet

De prime abord, il convient de distinguer les différents modules à développer dans le cadre du projet. Ainsi le diagramme de packages illustré dans la figure ci-après montre clairement les trois modules à mettre en jeu.

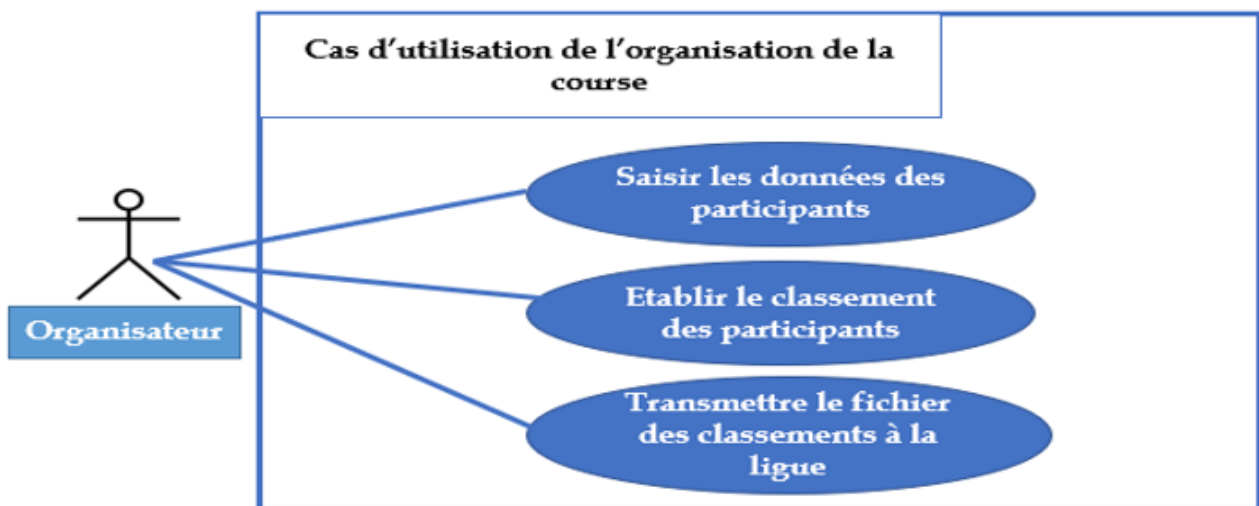


FIGURE 5 – Le diagramme de packages du projet

Ce diagramme permet de dégager trois packages principaux qui seront détaillés dans le cadre des cas d'utilisation illustré dans l'axe suivant.

B.2 Les cas d'utilisation

Définition

Les cas d'utilisation sont définis par une description textuelle, décrivant les objectifs et interactions entre le système et ses acteurs. Le format de présentation textuelle des cas d'utilisation est libre.

Diagramme de cas d'utilisation

Comme illustré par le diagramme de packages, trois modules sont à développer. Ces modules nécessitent l'élaboration de trois cas d'utilisation. Le premier cas d'utilisation du projet traite de la phase de l'organisation d'une course (tel montré sur la figure ci-dessous). En effet le seul acteur qui interagit avec le système dans ce cas est l'organisateur de la course. Ainsi le système doit permettre de réaliser un certain nombre de fonctionnalités qui sont répertoriées comme suit :

- La saisie des données des participants dans une course ;
- L'établissement du classement des participants dans la course ;
- L'envoi des résultats finaux à la ligue pour permettre à l'administrateur de mettre à jour le fichier des challenges.

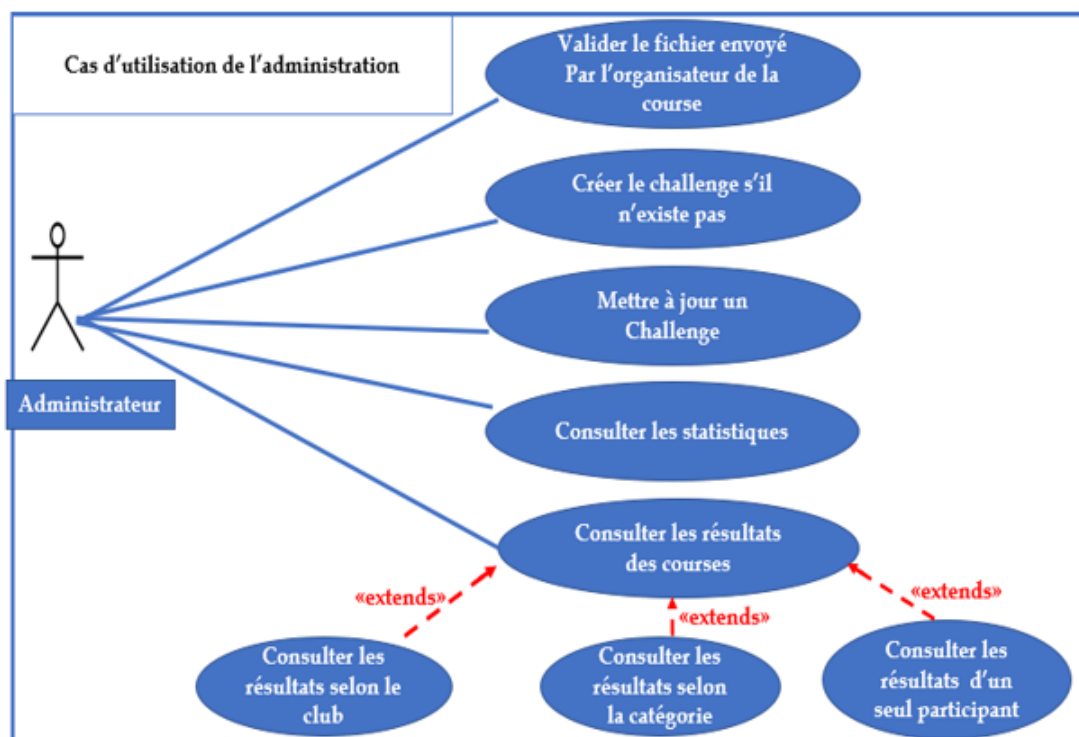


FIGURE 6 – Le diagramme de cas d'utilisation relatif à l'organisation d'une course.

S'agissant de l'étape de l'administration, l'acteur qui utilise le système est l'administrateur de la ligue. Dans ce cas l'administrateur doit pouvoir effectuer les tâches suivantes :

- Valider les fichiers des courses envoyé par les organisateurs des courses (Supprimer les participants qui ne sont pas inscrits dans la ligue, traiter les homonymes...) ;
- Créer le fichier challenge, s'il n'existe pas et le mettre à jour ;
- Consulter les résultats et les statistiques inhérentes aux courses selon un certain nombre de critères de choix (comme les clubs, les catégories...).

Ainsi la figures ci-desous illustre clairement le cas d'utilisation de cette étape.

Le dernier cas d'utilisation concerne l'étape relative à l'affichage des résultats pour les sportifs. Ainsi, le système devra permettre l'affichage des résultats globaux des courses, les classements relatifs et quelques graphes montrant l'évolution individuel pour chaque sportif. La figure ci-après montre ces étapes :

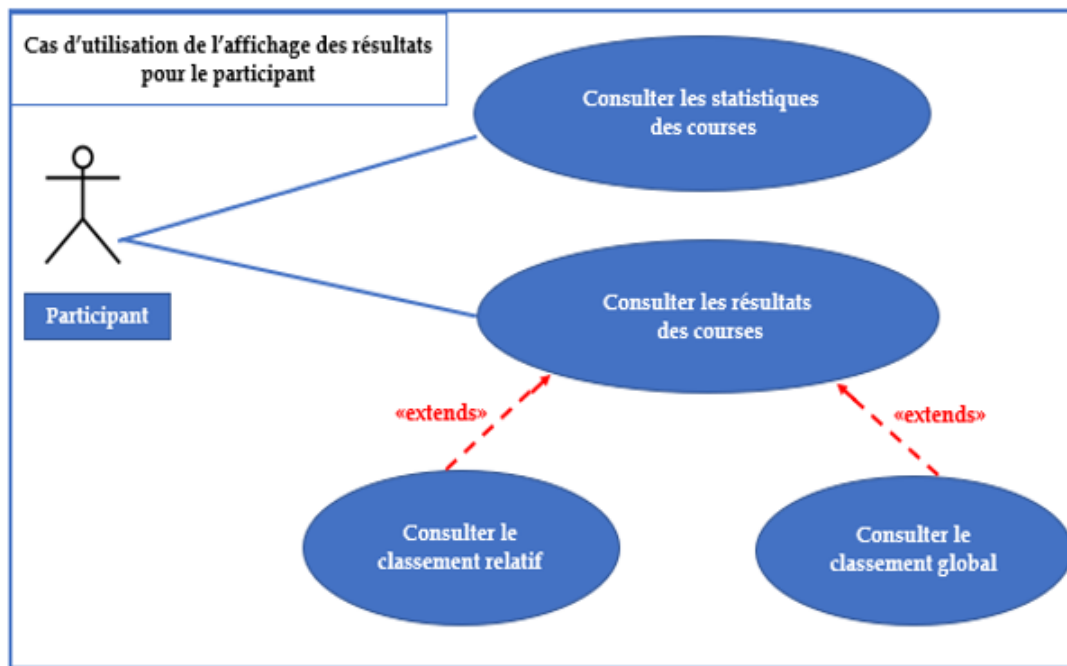


FIGURE 7 – Le diagramme de cas d'utilisation de la phase d'administration.

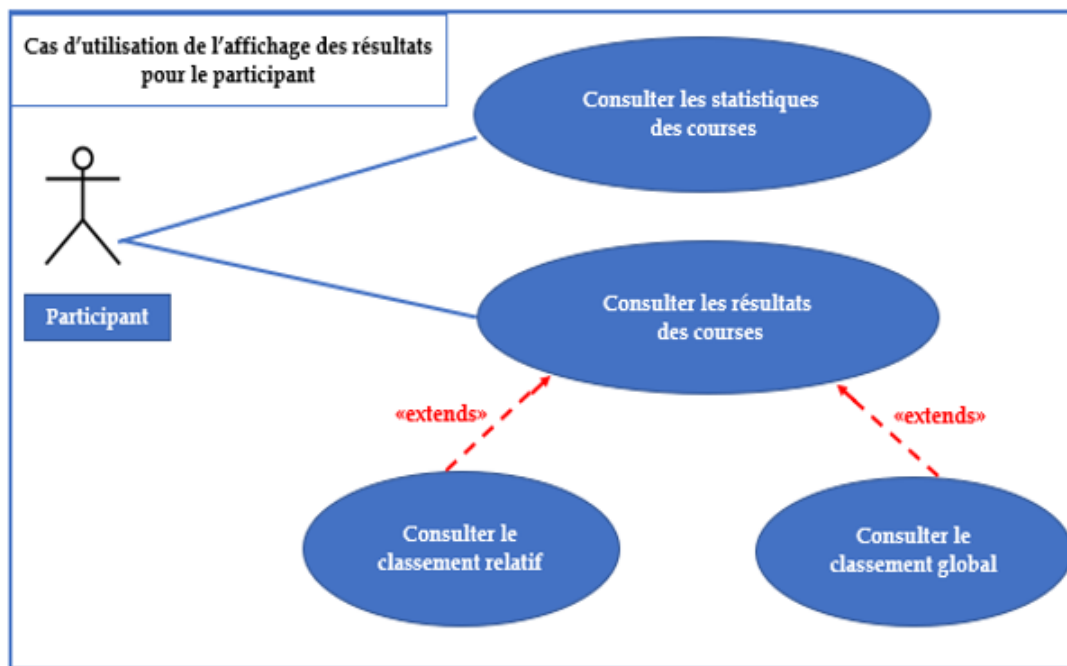


FIGURE 8 – Le diagramme de cas d'utilisation de l'affichage des résultats pour les participants.

B.3 Diagramme de classes

Définition

Le diagramme de classes est un schéma utilisé en génie logiciel pour présenter les classes et les interfaces des systèmes ainsi que les différentes relations entre celles-ci. Ce diagramme fait partie de la partie statique d'UML car il fait abstraction des aspects temporels et dynamiques.

Les classes

Le diagramme de classes du projet est illustré par la figure ci-après :

Le diagramme de classes du projet, figurant dans la capture d'écran précédente est composé de huit classes répertoriées comme suit :

- La classe *Personne* qui est une classe abstraite dont héritent les classes *Participant*, *Organisateur* et *Administrateur*. Cette classe possède les attributs suivants : identifiant de la personne, son nom, son prénom, son email et son numéro de téléphone.
- La classe *Participant* qui est une classe fille de la classe *Personne* qui possède, en plus des attributs hérités de la classe *Personne*, un attribut identifiant national qui va permettre de pallier au problème des homonymes dans l'avenir si une migration des données vers une base de données nationale est projetée.
- La classe *Administrateur* qui est une classe fille de la classe *Personne* ayant, en plus des attributs hérités de la classe *Personne*, des attributs d'authentification (nom utilisateur et mot de passe) permettant à l'administrateur de s'authentifier.
- La classe *Organisateur* qui hérite ses attributs et ses méthodes de la classe *Personne*.
- La classe *Club* : elle représente le club auquel appartient le participant, avec les attributs suivants : nom et la date de création du club.
- La classe *Challenge* : est une classe qui symbolise les challenges, elle est caractérisée par les attributs : nom, et l'année de création du challenge.
- La classe *Course* : contient les informations relatives aux courses organisées (la date de l'organisation de la course, le nombre de participants, le lieu d'organisation).
- La classe *Catégorie* : Comme les participants dans les courses appartiennent à des catégories différentes, il est nécessaire d'implémenter une classe qui stocke les informations inhérentes à chaque catégorie.

Les relations entres les classes

Les différentes classes du diagramme peuvent être liées par des types de relations variées selon la type de relation entre elles et également selon le nombre d'instances générées de chacune d'entre elle lors de l'exécution. Ainsi cet axe est dédié pour détailler ces relations.

- Les classes *Participant*, *Organisateur* et *Administrateur* sont des classes filles de la classe *Personne*, c'est-à-dire qu'elles héritent des méthodes et des attributs de celle-ci. Il reste à préciser que les méthodes seront surchargées ou redéfinies selon chaque classe.
- La classe *Club* et la Classe *Participant* sont liées par une relation d'agrégation c'est-à-dire que plusieurs participants appartiennent à un club et une personne adhère à un seul Club.
- La classe *Participant* est liée à la classe *course* par une relation d'association, en effet un participant peut participer à plusieurs courses et une course est ouverte à plusieurs personnes.
- Une *Course* doit être insérer dans un seul challenge, vu que la classe *Course* et *Challenge* sont liées par une relation d'association avec les cardinalités 1..1 du coté de la course.
- Un participant appartient à une et une seule catégorie, alors qu'une catégorie peut contenir plusieurs participants.
- Une course est organisée par un seul *Organisateur* et un organisateur peut organiser plusieurs courses.
- Un administrateur peut mettre à jour plusieurs *Challenge*, et un *Challenge* est obligatoirement par un seul administrateur.
- Les résultats d'une course figurent dans un seul fichier *Challenge* et un challenge peut contenir plusieurs courses.

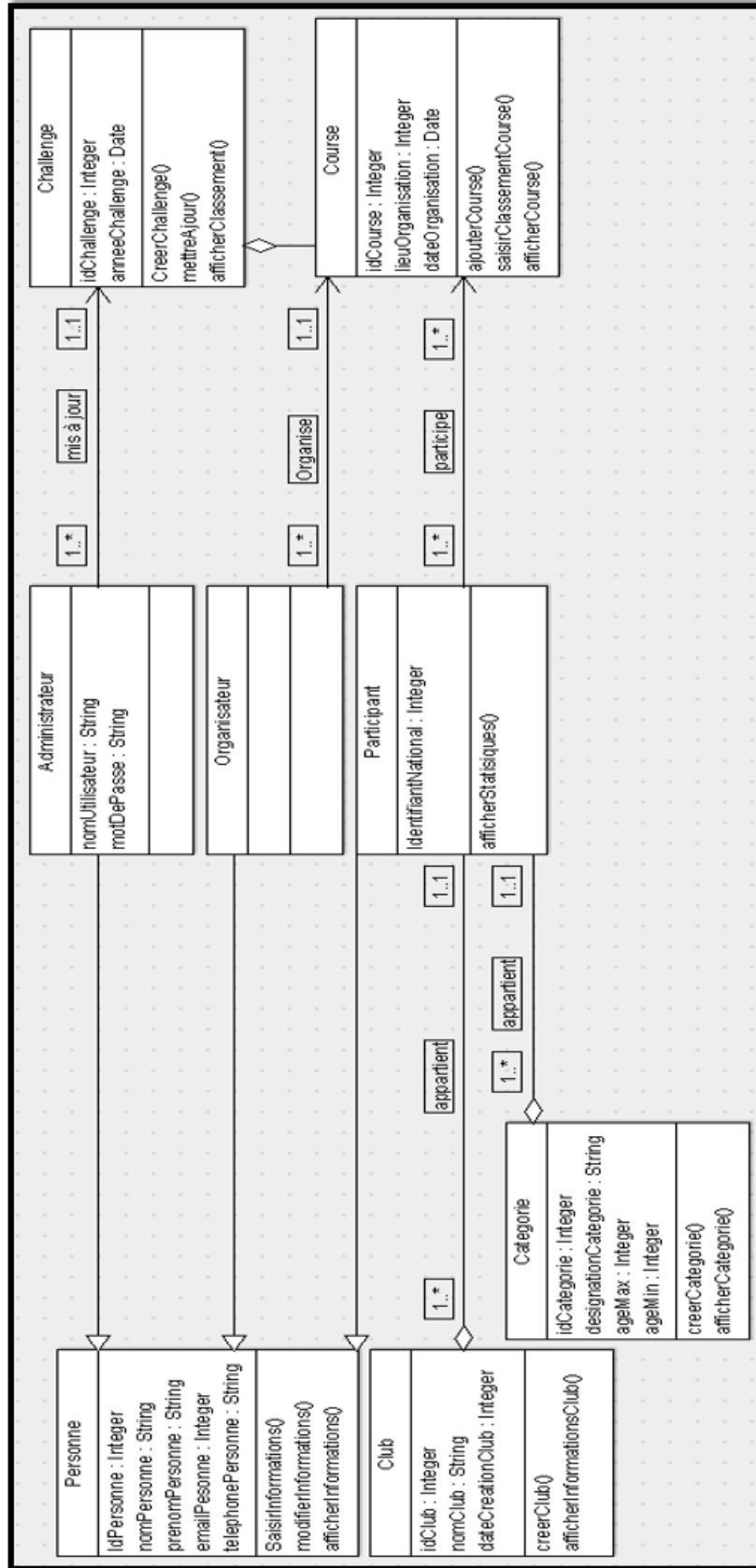


FIGURE 9 – Le diagramme de classes du projet.

B.4 Le diagramme d'activités

Définition

Les diagrammes d'activités permettent de mettre l'accent sur les traitements. Ils sont donc particulièrement adaptés à la modélisation du cheminement et de contrôle des flots de données.

Le diagramme d'activité du projet

Le diagramme illustré dans la figure ci-dessous montre le diagramme d'activités du projet. En effet, il représente les événements déclencheurs des différents processus. Également, il illustre les différentes tâches exécutées d'une manière séquentielle.

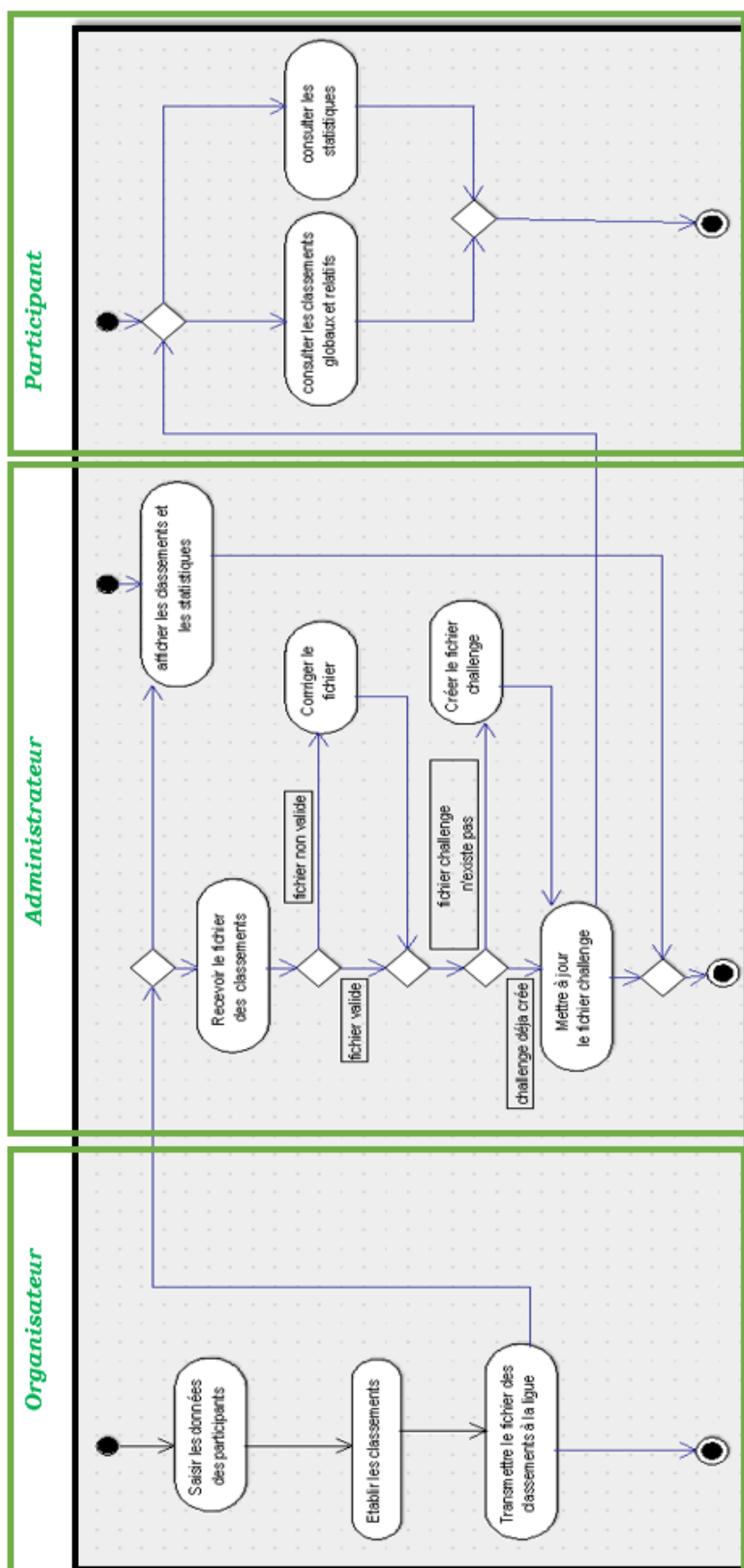


FIGURE 10 – Le diagramme d'activités du projet

C Réalisation du projet

Ce chapitre est l'occasion pour présenter les différents langage et outils utilisés pour réaliser le projet. Également, la mise en avant des différentes interfaces mises en œuvres pour l'exécution des différentes fonctionnalités. Enfin les tests effectués pour s'assurer du bon fonctionnement de logiciel.

C.1 Langages & outils utilisés

Langage R & package Shiny



FIGURE 11 – Le logo du R project

R est un langage de programmation et un logiciel libre dédié aux statistiques et à la science des données soutenu par la R Foundation for Statistical Computing. Il est dérivé du langage S développé par John Chambers et ses collègues au sein des laboratoires Bell.

Le projet R naît en 1993 comme un projet de recherche de Ross Ihaka et Robert Gentleman à l'université d'Auckland (Nouvelle-Zélande).

Le langage R existe en plusieurs distribution, cependant la plus connue demeure celle du R Project et du Comprehensive R Archive Network (CRAN). Il existe d'autres distributions comme la distribution proposée par Microsoft¹⁴ ou encore celle de l'entreprise Oracle, Oracle R Distribution¹⁵.

Shiny est un package R qui permet la création d'applications Web interactives directement à partir de R. Il permet d'héberger des applications autonomes sur une page Web ou les intégrer dans des documents R Markdown ou créer des tableaux de bord. Il offre également la possibilité d'étendre les applications Shiny avec des thèmes CSS, html widgets et des actions JavaScript, car le package entier repose sur ces différents langages.

Shiny permet de déployer une page web gérée par R et soutenu par HTML, CSS, Javascript et des framework reconnus comme Bootstrap ou encore Datatables.

R en combinaison avec Shiny ont été adoptés en concertation avec nos encadrants et le maître d'ouvrage pour les raisons suivantes :



FIGURE 12 – Le logo de Shiny

- Les statistiques relatives aux classements des sportifs devront être faites en un langage capable de traiter des données volumineuses (les données relatives aux challenges dans le cas de notre projet) en réduisant au maximum les ressources utilisés en mémoire.
- Des raisons pédagogiques. En fait, ce projet est l'occasion de découvrir les avantages et les limitations du langage R en termes de réalisation d'interfaces graphiques dynamiques et interactives.
- La simplicité de développement : R est un langage dit de "Haut niveau", c'est à dire d'après Wikipédia : "un langage de programmation orienté autour du problème à résoudre, qui permet d'écrire des programmes en utilisant des mots usuels des langues naturelles (très souvent de l'anglais) et des symboles mathématiques familiers."
- Un premier algorithme de classement a déjà été écrit, et fourni, en R.

Cependant, nous avons soulevé les point suivants :

- Déployer un serveur web fonctionnant sous R et Shiny n'est pas commun. Les hébergements web sont généralement en php.
- Les données concernés ne sont actuellement pas volumineuse, n'importe quel autre langage peut se permettre de les traiter avec autant d'efficacité.
- Shiny, avec l'appui des langages HTML, CSS et Javascript, se contente de générer une seule page web. Ce principe reste très réducteur de la fonction d'un site internet. De plus, cela oblige l'utilisateur à charger l'intégralité du site en une seule fois, incluant l'intégralité des données.
- Shiny permettant également de créer l'interface graphique, les étudiants ont émis des doutes quand à la création d'une interface graphique dynamique.

Pour le projet, les packages suivants ont également été utilisés :

- "shinythemes", afin d'avoir une page web élégante.
- "DT", alias "DataTables", qui nous permet d'afficher des tableaux dynamiques avec énormément d'options intéressantes (Recherche, tri, mise en forme...)
- "xlsx", qui nous permet d'utiliser des fichiers Excel (Lire, enregistrer).
- "plyr", pour de la manipulation de tableaux avancée.
- "stats", pour des fonctions de statistique intéressante.
- "ggplot2", afin d'afficher des graphiques plus perfectionnés.

GitHub, LaTeX, RStudio



FIGURE 13 – Le logo de GitHub

GitHub est un service web d'hébergement et de gestion de développement de logiciels, utilisant le logiciel de gestion de versions Git.

Le versionnage permet de travailler à plusieurs sur un code informatique, et de conserver l'historique des modifications du code.

GitHub propose ses services aux entreprises, mais le statut étudiant nous permet de privilégier d'un compte gratuit, nous permettant d'héberger notre code dans des répertoires privés.



FIGURE 14 – Le logo de LaTeX

LaTeX est un langage et un système de composition de documents, qui nous a permis d'écrire ce rapport de manière efficace.



FIGURE 15 – Le logo de LaTeX

RStudio est un environnement de développement gratuit, libre et multiplateforme pour R. L'ensemble des développements de ce projet ont été effectués avec ce logiciel.

C.2 Les deux outils distincts du projet

Pour permettre à l'utilisateur de bénéficier des fonctionnalités de notre logiciel, nous avons réalisé deux interfaces web. En effet, la première sera utilisée uniquement par l'administrateur de la ligue afin d'ajouter les différentes courses au challenge correspondant et également de consulter les informations et les statistiques relatives aux clubs et aux adhérents.

Outil classement

Dans un premier temps, l'outil "Classement" permet de réaliser les classements des sportifs. Il s'agit uniquement d'un outil de calcul en ligne : l'utilisateur entre une course, un challenge (optionnel) et il pourra télécharger un classement mis à jour.

Voici trois aperçu de l'interface, pour chacun des onglets :

FIGURE 16 – Aperçu de l'interface de classement - Onglet Course

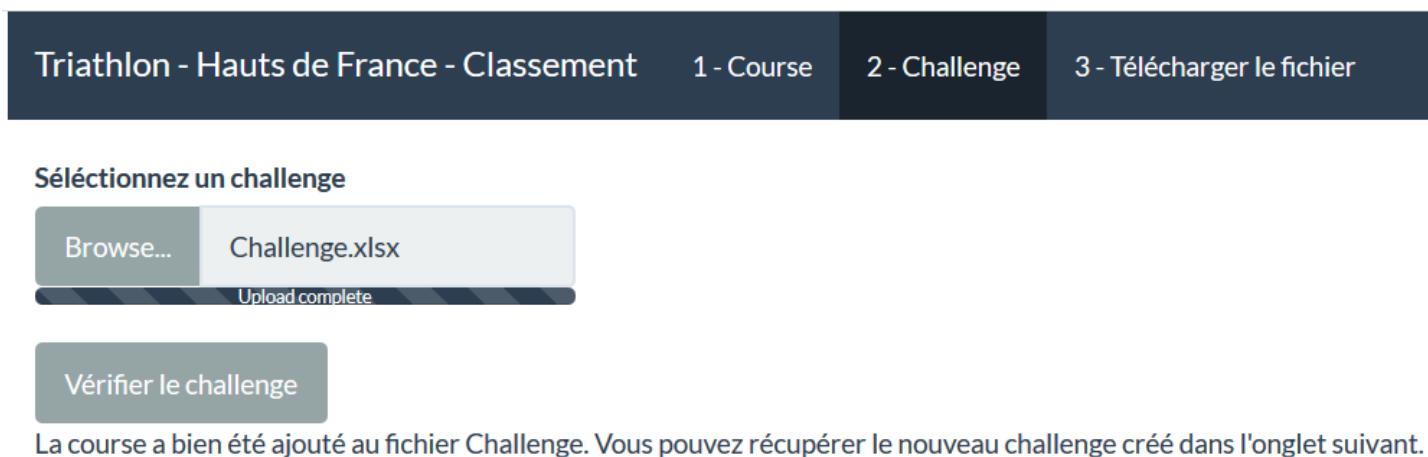


FIGURE 17 – Aperçu de l’interface de classement - Onglet Challenge

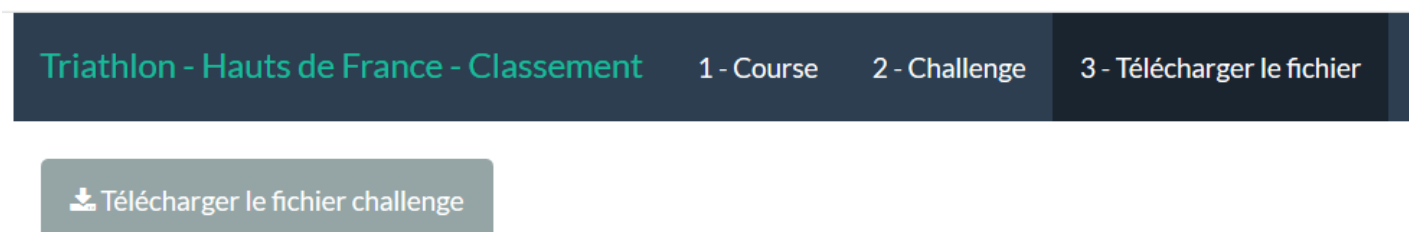


FIGURE 18 – Aperçu de l’interface de classement - Onglet Téléchargement

Le processus se déroule en trois étapes :

1. La course à ajouter au challenge doit être vérifiée, l’administrateur doit donc entrer le fichier au format excel et certains paramètres :
 - Le nom de la course
 - La discipline concernée (Triathlon, Duathlon, Aquathlon)
 - La distance concernée (XS, S, M, L XL)
 - La date de la course
 - La catégorie (Adultes, Jeunes)
2. Le challenge existant peut être ajouté par l’administrateur, s’il ne l’est pas alors un nouveau challenge sera créé.
3. Le challenge mis à jour est téléchargeable.

Plus en détail, voici les différents points que l’algorithme aborde :

1. Gérer les erreurs de saisies utilisateur
2. Vérifier la licence de l’utilisateur grâce au fichier Référence de la ligue.
3. Gérer les éventuels cas d’Homonymes présents.
4. Calculer les points gagnés pour la course.

5. Intégrer le fichier course au fichier challenge, en vérifiant qu'une course du même nom n'existe pas.
6. Calculer le classement par club.
7. Enregistrer le fichier et le télécharger.

Pour rappel, un code R nous a été fourni qui calculait ces classements. Nous avons travaillé dessus, puis remis à jour. Voici un exemple de ce qu'il est possible de faire grâce à l'interprétation matricielle de R, en évitant d'utiliser des boucles. Un aperçu ci-dessous du classement par clubs, original puis retravaillé à l'aide des fonctionnalités de R :

```
# Création du classement par clubs

CLUB<-levels(factor(chf$CLUB[-c(1,2,3)]))
#
# Exclusion du classement Club des licenciés sans Club
#
CLUB<-CLUB[!CLUB==""]
long_club<-length(CLUB)

class_club<-data.frame(matrix(0,ncol=(ncchf-9),nrow=long_club),check.names=FALSE)
colnames(class_club)<-c("CLUB",names(chf[11:ncchf]))
class_club$CLUB<-CLUB
if (rep1==1){
  for (j in ncchf:11){
    for (i in 1:long_club){
      class_club[i,(j-9)]=sum(as.numeric(as.character(chf[chf$CLUB==class_club$CLUB[i],j])),na.rm=T)
    }
    class_club[, (j-9)]<-as.factor(class_club[, (j-9)])
  }
}
if (rep1==2){
  for (j in ncchf:11){
    for (i in 1:long_club){
      class_club[i,(j-9)]=sum(as.numeric(chf[chf$CLUB==class_club$CLUB[i],j]),na.rm=T)
    }
  }
}
class_club<-class_club[order(class_club$TOTAL,decreasing=T),]
Classement<-1:long_club
class_club<-data.frame(Classement,class_club,check.names=FALSE)
nclass<-ncol(class_club)
class_club<-class_club[,c(2,1,3:nclass)]
class_club[, "Classement"]<-as.factor(class_club[, "Classement"])

res_club<-rbind(chf[1:4,c(7,6,11:ncchf)],class_club)
nres<-ncol(res_club)
# Duplication colonne TOTAL et adjonction en début de tableau
res_club<-res_club[,c(1,nres,2:nres)]
nres<-ncol(res_club)
colnames(res_club)[colnames(res_club)=="TOTAL.1"] <- "TOTAL"
res_club1<-res_club
colnames(res_club1)[colnames(res_club1)=="TOTAL.1"] <- "TOTAL"
```

FIGURE 19 – Code du classement par clubs - Original

```
clubs_all = subset(challenge, select = -c(NOM,Prénom,Sexe,Cat,Classement,Identifiant,Date.nais.,Ema
clubs <- aggregate(. ~ CLUB, clubs_all, sum)
```

FIGURE 20 – Code du classement par clubs - Nouveau

Outil challenge

Dans un second temps, l'outil "Challenge" permet d'afficher ces résultats dans un format intéressant. Dans une seule page web seront contenu tout les tableaux de résultats des différentes années à partir de 2017, et ce pour les catégories Adultes et Jeunes, ainsi que pour les Clubs.

Voici un aperçu de l'interface web :

Triathlon - Hauts-de-France		Challenge 2017														
2017		Sportifs - Adultes Clubs - Adultes Sportifs - Jeunes Clubs - Jeunes Statistiques														
Afficher 25 éléments		Rechercher :														
Nom	Prénom	Points	Courses réalisées	Sexe	Catégorie	Classement	CLUB	Avion Triathlon S 15/01/2017	Lievin.3 Triathlon S 29/01/2017	Lievin.2 Triathlon XS 29/01/2017	Lievin.1 Duathlon S 29/01/2017	Houdain.Lez.Bavay.1 Duathlon XS 11/03/2017	Houdain.Lez.Bavay.2 Duathlon S 11/03/2017	Roubaix.2 Aquathlon S 11/03/2017	Roubaix.1 Aquathlon XS 11/03/2017	Arras.2 Duathlon M 12/03/2017
BAELEN	Delphine	2475	10	F	Ve	1	SKWALATHLON	0	0	0	0	0	0	0	0	250
SAISON	Antoine	2110	8	M	Se	1	TRIATHLON CLUB DU MONTREUILLOIS	0	0	0	225	0	0	0	0	0
PEYTOUR	Olivia	1725	7	F	Se	1	CLUB LOISIRS LEO LAGRANGE	225	225	0	0	0	0	0	0	0
LELIEVRE	Christelle	1587,5	6	F	Ve	2	TRIATHLON CLUB DU MONTREUILLOIS	0	0	0	225	0	0	0	0	0
ROUBLIC	Arnaud	1486,25	5	M	Se	2	GRAVELINES TRIATHLON	0	0	0	0	0	0	0	0	0
DEBROUX	Marie Claude	1412,5	6	F	Ve	3	VALENCIENNES TRIATHLON	225	225	0	0	0	0	0	0	0
CANU	Laurent	1372,5	6	M	Ve	1	COTE D OPALE TRIATHLON CALAIS SAINT OMER	0	0	0	0	0	0	0	0	0
FOCQUENOY	Heloise	1365	5	F	Se	2	LILLE TRIATHLON	0	0	0	0	0	0	225	0	0
SIX	Sylvie	1336,25	6	F	Se	3	CLUB LOISIRS LEO LAGRANGE	127,5	165	0	0	0	0	0	0	0

FIGURE 21 – Aperçu de l'interface web affichant les résultats

Voici les éléments composant cette interface :

- Une barre d'onglet pour les années du challenge, pour le moment uniquement 2017.
- Une barre d'onglet pour l'année courante, comportant les challenge adultes et jeunes ainsi que les classements par clubs.
- Le tableau de résultat.

Les avantages offert par le plugin DataTables sont :

- Une recherche globale, en haut à droite du tableau.
- Un tri possible pour toute les colonnes.
- Une recherche possible pour toute les colonnes.
- La sélection de lignes.
- Un affichage dynamique sur plusieurs pages.

En cliquant sur la ligne d'un participant, des statistiques plus précise sont disponible ainsi que ses informations personnelles. Ainsi, un indice de performance est calculé, globalement, par sexe et enfin par catégorie.

Des statistiques globales sont également présentes, dans l'onglet correspondant. En voici un aperçu :

Adultes

Il y a **52** courses dans le challenge.

Il y a **831** participant(e)s appartenant à **44** clubs.

Il y a **164** femmes et **667** hommes.

Par catégorie, ils sont **415** en senior et **416** en vétéran.

La moyenne des points est de **335.24** par sportif et de **6307.61** par club.



FIGURE 22 – Statistiques globales adultes

Jeunes

Il y a **104** courses dans le challenge.

Il y a **571** participant(e)s appartenant à **25** clubs.

Il y a **217** femmes et **354** hommes.

Par catégorie, ils sont **112** en benjamin, **89** en cadet, **55** en junior, **113** en minimes, **91** en poussin et **111** en pupille.

La moyenne des points est de **56.08** par sportif et de **1280.84** par club.



FIGURE 23 – Statistiques globales jeunes

C.3 Déploiement du projet

Comme énoncé précédemment, le projet a besoin d'un serveur R shiny pour déployer les outils sur le web. Pour cela nous disposons d'un serveur personnel debian ainsi que d'une adresse ipv4 valide.

Voici comment s'est déroulé ce déploiement :

1. Installation de R, avec les packages nécessaires cités pus haut. [2]
2. Installation de shiny-server [3]

3. Configuration de shiny-server : deux serveurs lancés, l'un sur le port 3838 et l'autre sur le 3939. Les deux pointent vers deux dossiers de l'utilisateur "triathlon" créé pour l'occasion.
4. Grâce aux identifiants OVH [4] fournis par la ligue, nous avons eu accès au nom de domaine "triathlonhdf.fr" (où leur nouveau site web venait d'être déployé) et avons créés les sous-domaine challenge.triathlonhdf.fr et classement.triathlonhdf.fr. Ces deux sous-domaines pointent à l'aide d'une redirection invisible vers notre ipv4 et le port souhaité.
5. Un accès FTP sera fourni à la ligue afin de modifier les fichiers excel nécessaire au fonctionnement.

C.4 La documentation automatique

La documentation du projet consiste commenter toutes les fonctions pour faciliter la compréhension du codes et les changements futurs à effectuer. Dans le cadre de notre projet, la documentation consiste à ajouter les commentaires concernant les boucles et les conditions utilisées au sein du code.

C.5 Schéma des tests

Les tests revêtent une importance capitale avant, lors et après le déploiement du logiciel sur le serveur. En effet, ils permettent d'identifier un nombre maximum de comportements problématiques du logiciel afin d'en augmenter la qualité. Ainsi, il demeure important de le schéma et le type de tests à effectuer.

Dans cette optique, nous avons décider d'effectuer des tests unitaires pour chaque fonctionnalité implémentée, un test global pour l'ensemble du projet et à la fin un test d'acceptation lors du déploiement du projet sur le serveur dédié.

L'ensemble de ces tests concernent les deux modules développés à savoir : -l'outil de classement ; -l'outil challenge.

Outil classement

La première phase des tests sur l'outil challenge consiste en la validation du code Comparaison.R fourni et Challenge.R . Par conséquent, les tests effectués sont répertoriés comme suit :

1. Tester les la modification des entêtes du fichiers Excel
2. Tester le choix des valeurs qui doivent être entrées par l'utilisateurs
3. Tester l'existence du fichier xl de la course
4. Tester l'existences des homonymes(validé)
5. Tester l'ajout d'une course
6. Tester les homonymes

Outil challenge

S'agissant de l'outil challenge, les tests réalisé sont les suivants :

1. La visualisation des résultats sous la forme dans un tableau sur l'interface graphique
2. Tester le tri des résultats sur tous les critères spécifiés par le client (tri selon le sexe, la catégorie, ...)

3. La visualisation des résultats d'un seul sportif en cliquant sur la ligne correspondante
4. L'affichage des graphes représentant les statistiques à chaque sportif.

Dans cette optique, nous avons décidé d'effectuer des tests unitaires pour chaque fonctionnalité implémentée, un test global pour l'ensemble du projet et à la fin un test d'acceptation lors du déploiement du projet sur le serveur dédié.

===== Dans cette optique, nous avons décidé d'effectuer des tests unitaires pour chaque fonctionnalité implémentée, un test global pour l'ensemble du projet et à la fin un test d'acceptance lors du déploiement du projet sur le serveur dédié.

ensemble de ces tests concernent les deux modules développés à savoir :

Conclusion

En somme, notre projet qui consistait en la réalisation d'un logiciel permettant la gestion et la mise à jour des challenges au profit de la fédération Française de triathlon, nous a permis dans un premier temps de prendre contact avec le maitre d'ouvrage afin de déterminer et de statuer sur ses besoins fonctionnels. Également, il nous a permis de comprendre le processus de l'organisation des courses et de la mise à jour des challenges organisés par le Fédération Française du Triathlon. Par ailleurs, le projet était pour l'occasion pour mettre en pratique les acquis théoriques relatifs aux méthodes de gestion de projet, essentiellement SCRUM, de réaliser la conception moyennant le langage de modélisation UML ainsi que la manipulation de nouveaux langages informatiques essentiellement R et son serveur dédié Shiny.

En termes de perspectives, malgré que le langage R soit puissant en termes de calcul et de manipulation des statistiques de données volumineuses tout en diminuant la complexité des traitements. Néanmoins, il demeure limité en termes d'interfaces graphiques dynamiques, comparé à d'autres langages dédiés au web comme le PHP ou le JAVA EE. Étant donné que le nombre de courses organisées durant l'année est limité et que les challenges de chaque année sont traités séparément, il serait donc judicieux, dans le cadre de l'évolution du projet de le reprendre en utilisant un langage dédié au web en assurant la persistance donnée par une base de données et également d'implémenter le module relatif à l'organisation de la course afin que le système d'information soit entièrement.

Bibliographie

- [1] LigueHDF. Création ligue triathlon, Février 2018. <http://triathlonhdf.fr/samedi-03-fevrier-2018-la-ligue-des-hauts-de-france-de-triathlon-est-creee/>.
- [2] R project. Installation de r sur debian, 2018. <https://cran.r-project.org/bin/linux/debian/>.
- [3] developpez.com. Installation de shiny-server sur debian, 2018. <https://r.developpez.com/actu/89499/Installer-et-configurer-un-serveur-R-shiny-su-Ubuntu/>.
- [4] ovh.fr. Ovh, 2018. <https://www.ovh.com/fr/>.