

**Année Universitaire 2019-2020**

**UNIVERSITE DE FIANARANTSOA**

**ECOLE DE MANAGEMENT ET D’INNOVATION TECHNOLOGIQUE**

**Mention : Informatique**

**Parcours : Modélisation et Ingénierie Informatique**

**Mémoire fin d’études en vue d’obtention du diplôme de Master professionnel**

MISE EN PLACE DES EXTENSIONS DE LA PLATEFORME AREN

Présenté par **ANDRIAMBOLAHARIMIHANTA Havana**

Sous la codirection du : Docteur **RAKOTONIRAINY Hasina Lalaina** et du

Docteur **NIRINA AVO Antsa Nasandratra**

Et l’encadrement de : Monsieur **RAMINOSON Tsiriniaina**

Identification des membres du jury :

Président : Docteur DIMBISOA William Germain

Examinateur : Docteur RAZAFIMAHATRATRA Hajarisena

Docteur TAREHY Brice Evrard

Encadrants : Docteur **Hasina Lalaina RAKOTONIRAINY**, Maître de Conférences,

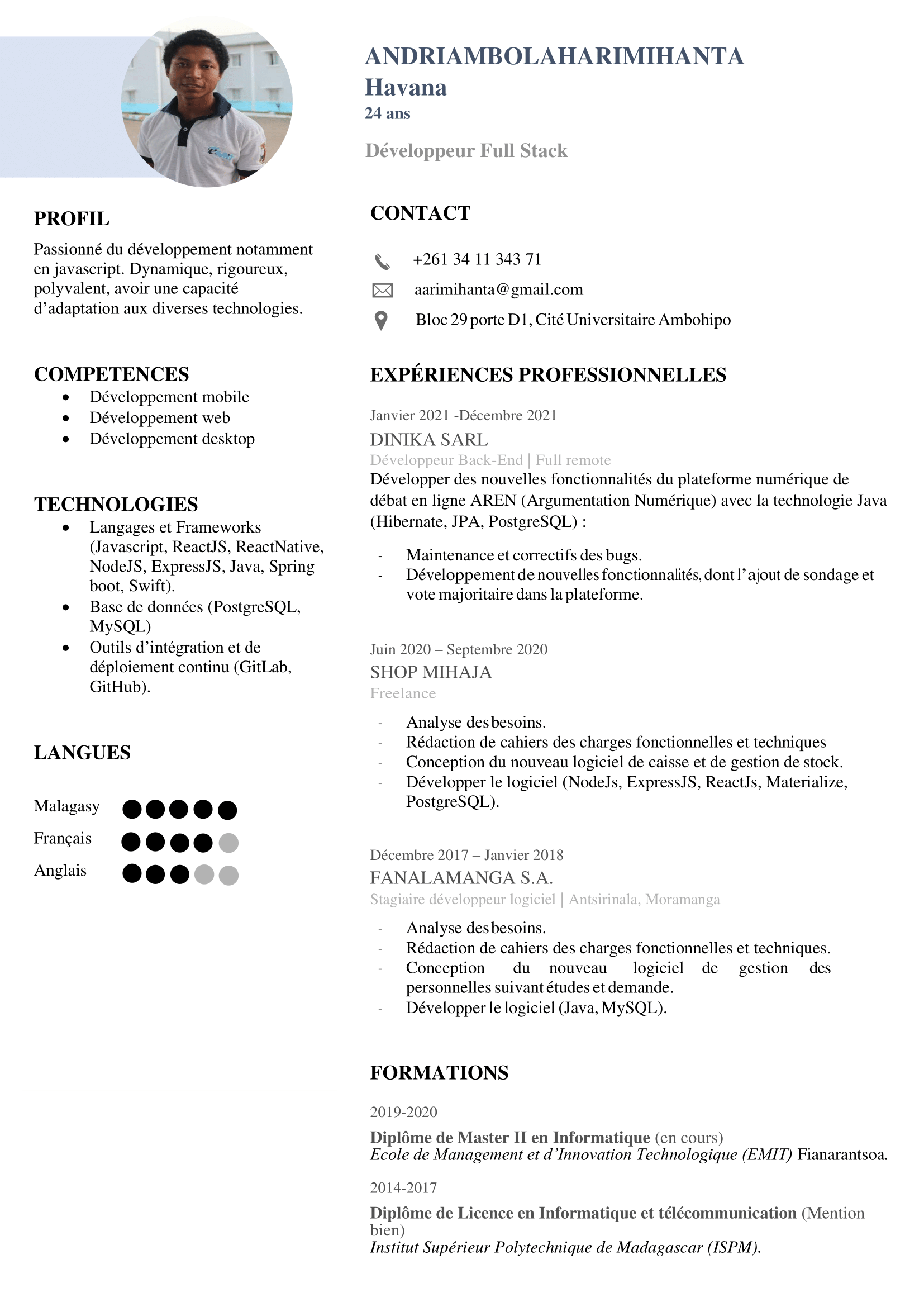
EMIT – Université de Fianarantsoa.

Docteur **Antsa Nasandratra NIRINA AVO**, Enseignant Chercheur,

Institut Supérieur de Technologie AMBOSITRA.

Monsieur **Tsiriniaina RAMINOSON**, Responsable au sein de l’Entreprise DINIKA SARL

# Curriculum Vitae



# Avant-propos

Afin d’appliquer les méthodologies enseignées à l’EMIT, en tant qu’étudiant en cinquième année, une activité professionnelle est obligatoirement nécessaire pour mettre en pratique la formation académique durant ces deux années d’étude en Master.

Le but en est de pouvoir orienté dans le monde professionnel afin que le titre de diplôme attribué soit digne de ce nom, les travaux confiés durant le stage devraient nécessairement apporter des impacts et profitables pour la société dans la conduite de ses affaires.

Pourtant, la société DINIKA ou Datascience INnovation Intelligence Knowledge Acquisition est chargée à développer une nouvelle fonctionnalité dans la plateforme AREN. Le projet AREN (ARgumentation Et Numérique) vise à élaborer une plateforme numérique de débats en ligne qui favorise le développement des compétences argumentatives des élèves et de leur esprit critique, dans une perspective d’éducation à la citoyenneté. L’implémentation des fonctionnalités de vote majoritaire et de débat sur carte font partie des grands œuvres dans le développement de aren.

# Remerciements

Avant de vous entraîner dans les recoins les plus complexes de cette présentation du mémoire de la fin du cycle, on souhaiterait remercier différentes personnes qui, d’une manière ou d’une autre, ont rendu possible l’existence de ce moment très particulier et précieux.

Tout d’abord, nous tenons à remercier Dieu de nous avoir donné la santé durant tous ces périodes afin de pouvoir réaliser le projet.

Nous tenons à remercier :

* Monsieur **HAJALALAINA Richard Aimée**,Maitre de conférence HDR et Président de l’Université de Fianarantsoa, d’avoir veillé à la bonne qualité de formation des étudiants de l’Université de Fianarantsoa.
* Monsieur **RAKOTONIRAINY Hasina**, Maitre de conference etDirecteur de l’Ecole de Management et d’Innovation Technologique (EMIT), de nous avoir donné l’opportunité de suivre les études durant ces deux dernières années ainsi que pour sa patience en mon égard, son précieux aide et conseils en tant que directeur de recherche.
* Monsieur **RAZAFINIMARO Tsimiala Toky Arisetra Eddy,** responsable de la mention informatique.
* Au Docteur **Antsa Nasandratra NIRINA AVO** et à Monsieur **Tsiriniaina RAMINOSON**, qui ont fait office d’encadreur professionnel et qui ont sacrifié une partie de leur temps précieux, pour nous donner les orientations à suivre pour la bonne marche de la mission  au sein de l’Entreprise DINIKA SARL.
* Tous les personnels administratifs et les enseignants de l'établissement de l’EMIT pour les formations acquises.
* Tous les équipes de DINIKA SARL, pour leur accueil et leur entière coopération.
* Les chers parents, les chers frère et sœur, la famille et tous ceux qui ont contribué de près ou de loin à la réalisation de ce mémoire. Qu’ils trouvent ici l’expression de ma profonde gratitude et reconnaissance.

# Liste des figures

Figure 1.1 Offre des formations de l'EMIT.................................................................................3

Figure 1.2 Organigramme de l'EMIT..........................................................................................4

Figure 4.1 Le cycle de vie en cascade.......................................................................................13

Figure 4.2 les diverses phases du modèle en V.........................................................................14

Figure 4.3 le cycle de développement de la méthode Scrum....................................................16

Figure 4.4 Exemple de tableau kanban.....................................................................................18

Figure 4.5 Cycle de développement de Extrem Programing....................................................20

Figure 6.1 Diagramme de cas d’utilisation...............................................................................28

Figure 6.2 Diagramme d'activité du cas d'utilisation « s’authentifier.............................29

Figure 6.1 Diagramme d'activité du cas d'utilisation "Création du vote majoritaire"...............31

Figure 6.2 Diagramme d'activité du cas d'utilisation " Création des débats sur carte".............32

Figure 6.3 Diagramme d'activité du cas d'utilisation " Argumenter sur les débats "................33

Figure 6.4 Diagramme d'activité du cas d'utilisation " Effectuer une vote".............................34

Figure 6.5 Diagramme d'activité du cas d'utilisation " Consulter les existants "......................35

Figure 6.6 Le diagramme de classe de l'application.................................................................36

Figure 7.1 Cursus d’exécution d’un programme java...............................................................38

Figure 7.2 Architecture logiciel de l’application......................................................................42

Figure 7.3 Architecture matériel de l’application.....................................................................43

Figure 9.1 Interface graphique de la page permettant de créer un vote majoritaire..................45

Figure 9.2 Interface graphique de la page permettant d’effectuer un vote...............................47

Figure 9.3 Interface graphique de la page permettant de créer un nouveau document pour un débat sur carte...........................................................................................................................48

Figure 9.4 Interface graphique de la page montrant la première étape de la création du débat sur carte. 49

Figure 9.5 Interface graphique de la page montrant la deuxième étape de la création du débat sur carte. 51

Figure 9.6 Interface graphique de la page montrant la liste des débats déjà publiés dans la plateforme. 52

Figure 9.7 Interface graphique de la page montrant les détails d’un débat avec une carte 53

Figure 9.8 Interface graphique de la page montrant le formulaire d’argumentation. 54

# Liste des tableaux

[Tableau 6.1: Dictionnaire de données 25](#_Toc90537045)  
[Tableau 6.2: Description textuelle du cas d'utilisation "s'authentifier" 30](#_Toc90537045)  
[Tableau 6.3: Description textuelle du cas d'utilisation " Création du vote majoritaire " 31](#_Toc90537045)

[Tableau 6.4: Description textuelle du cas d’utilisation « Creation du debat sur carte » 32](#_Toc90537045)

[Tableau 6.5: Description textuelle du cas d'utilisation " Argumenter sur les débats " 33](#_Toc90537045)

[Tableau 6.6: Description textuelle du cas d'utilisation " Effectuer un vote " 34](#_Toc90537045)

[Tableau 6.7: Description textuelle du cas d'utilisation " Consulter les existants " 35](#_Toc90537045)

# Liste des abréviations

AES : Administration Economie Sociale

AN : Alpha Numérique

AREN : ARgumentation Et Numérique

CFC : Centre de Formation Continue

CIRAD : Centre de Coopération Internationale en Recherche Agronomique pour le Développement)

CM: Communication Multimédia

CNRE : Centre de Recherche en Education Environnementale

CNRS : Centre national de la recherche scientifique

CSS : Cascading Style Sheets

CUFP : Centre Universitaire de Formation Professionnalisant

DAII : Développement d’Application Internet/Intranet

DINIKA : Datascience- Innovation- Intelligence & Knowledge Acquisition

DOM : Document Object Model

EMIT : Ecole de Management et d’Innovation Technologique

HTML : HyperText Markup Language

IRD : Institut de Recherche pour le Développement

ISPM : Institut Supérieur Polytechnique de Madagascar

JSON : JavaScript Objet Notation

LIMAD : Laboratoire d’Informatique et de Mathématiques pour le développement

LIRMM : Laboratoire d'Informatique, de Robotique et de Microélectronique de Montpellier

LMD : Licence, Master et Doctorat

M2I : Modélisation et Ingénierie Informatique

MD : Management Décisionnel

MISS : Mathématique et Informatique pour les Sciences Sociales

MIT : Massachusetts  Institut technologie

MMS : Multimedia Messaging Service

MVC: Model View Controller

MVP : Minimal Vital Product

ODT : OpenDocument Text

ORM : Object Relational Mapping

SA : Société Anonyme

SARL : Société Anonyme à Responsabilité Limité

SGBDR : Système de Gestion de Base de Données Relationnelle

SIGD : Système d’Information, Géométrique et Décision

SMS : Short Message Service

UF: Université de Fianarantsoa

UML: Unified Modeling Language

# Glossaire

**AREN :** ARgumentation Et Numérique est une plateforme numérique open source

**Débat sur une carte :** c’est une méthode de discussion qui implique l'utilisation de cartes pour faciliter la communication et l'échange d'informations entre les participants.

**Dialoguea :** est une plateforme de débats numériques en ligne.

**Linfluent :** est une plateforme de communication dédiée à tous ceux qui souhaitent profiter du plein potentiel des médias sociaux.

**UML :** c'est un langage de modélisation graphique qui permet de décrire et de représenter les différents aspects d'un système informatique. Il est utilisé pour la conception, la spécification, la documentation et la communication des systèmes logiciels.

**Vote majoritaire**: Le vote majoritaire est une forme de vote par valeurs. L’utilisateur attribut des mentions à chaque choix de vote et se permet d’attribuer les mêmes mentions pour différent choix. Pour la restitution des résultats, elle se fait par la médiane plutôt que par la moyenne.

# Sommaire

[Curriculum Vitae](#_Toc128318695) 0

[Avant-propos I](#_Toc128318696)

[Remerciements II](#_Toc128318697)

[Liste des figures III](#_Toc128318698)

[Liste des tableaux V](#_Toc128318699)

[Liste des abréviations VI](#_Toc128318700)

[Glossaire VIII](#_Toc128318701)

[Sommaire IX](#_Toc128318702)

[Introduction générale 1](#_Toc128318703)

[PARTIE 1 : PRESENTATION GENERALE](#_Toc128318704) 0

[Chapitre 1 : Présentation de l’Ecole Management et d’Innovation Technologique (EMIT) 3](#_Toc128318705)

[Chapitre 2 : Présentation de Datascience INnovation Intelligence Knowledge Acquisition 6](#_Toc128318710)

[Chapitre 3 : Présentation du projet 8](#_Toc128318717)

[PARTIE 2 : ANALYSE ET CONCEPTION DU PROJET 0](#_Toc128318722)

[Chapitre 4 : Méthodes et notations utilisés 12](#_Toc128318723)

[Chapitre 5 : Analyse 22](#_Toc128318727)

[Chapitre 6 : Conception du projet 26](#_Toc128318732)

[PARTIE 3 : REALISATION DU PROJET 0](#_Toc128318736)

[Chapitre 7 : Spécification des outils utilisés 37](#_Toc128318737)

[Chapitre 8. Mise en œuvre et implémentation 42](#_Toc128318744)

[Chapitre 9 : Présentation du logiciel 44](#_Toc128318747)

[Conclusion et perspective 55](#_Toc128318757)

[Bibliographie VIII](#_Toc128318758)

[Webographie X](#_Toc128318759)

[Table des matières XI](#_Toc128318760)

[Annexes XIV](#_Toc128318761)

# Introduction générale

Un débat numérique est une discussion qui se déroule en ligne, sur une plateforme numérique. Ce type de débat peut prendre différentes formes. Il peut porter sur une grande variété de sujets, allant de la politique et de l'actualité à la culture populaire, en passant par les affaires et les technologies. Les participants peuvent être issus de différents horizons et régions géographiques, ce qui peut offrir une perspective plus diversifiée sur les questions discutées. Cependant, les débats numériques peuvent aussi présenter des défis, tels que la difficulté de maintenir des normes de civilité et d'assurer la qualité des informations partagées. Il favorise également l’appropriation des connaissances et permet aux participants de construire un point de vue éclairé sur les questions débattues. Il s’avère nécessaire de renouveler la pratique et les usages du débat numérique en ligne, de diffuser largement l’usage des versions futures de la plateforme de débat numérique.

Par conséquent, la plateforme élaborée, portant le nom AREN[[1]](#footnote-1), c’est une nouvelle plateforme numérique de débats en ligne qui favorise le développement des compétences argumentatives. Elle exploite toujours les moyens plus fiables pour le débat numérique. Cette plateforme permet la participation simultanée d’un grand nombre d’élèves à un débat, offre du temps pour construire les arguments et permet de garder la trace des échanges, support pour un travail réflexif. Le projet comporte deux volets de recherche. Un volet didactique consiste à expérimenter la plateforme dans les classes afin d’évaluer son impact et d’en proposer des évolutions. Un volet intelligence artificielle (IA) porte sur le traitement automatique du langage naturel et vise à développer et évaluer un service IA de classement thématique des interventions des élèves facilitant leur analyse et la préparation de la synthèse réflexive.

Pour mieux cerner à ce sujet, le travail sera divisé en trois parties, dans la première partie dont :

* Une partie de présentation générale qui va se concentrer sur la présentation de l’EMIT, et de l’entreprise DINIKA et une présentation détaillée du projet AREN.
* Une partie de l’Analyse et Conception du projet qui consiste à développer les différentes étapes suivies pour la réalisation du projet.
* Une partie la réalisation du projet qui consiste à présenter les outils utilisé pour la mise en œuvre du projet et de l’application developpée.

# PARTIE 1 : PRESENTATION GENERALE

## Chapitre 1 : Présentation de l’Ecole Management et d’Innovation Technologique (EMIT)

### Historique [emit.mg]

L’École de Management et d’Innovation Technologique (EMIT) a été créée au sein de l’université de Fianarantsoa par le Décret N° 2016-1394 du 15 novembre 2016.Auparavant, elle a été connue sous le nom de Centre Universitaire de Formation Professionnalisante (CUFP) par le Décret N° 2005/205 du 26 avril 2005 et dispensait de formations continues en vue de l’obtention du diplôme de licence professionnelle.

L’établissement fut aussi appelé Centre de Formation Continue (CFC) créé par l’arrêté rectoral N° 99-23/UF/R du mars 1999 et dispensait de formation de techniciens supérieurs.

### Missions

La principale mission de l’école est de dispenser de formations continues en informatiques, en administration et en relation publique. Former des techniciens supérieurs spécialisés et immédiatement opérationnels en entreprise. L’école assure le perfectionnement professionnel des étudiants, des demandeurs d’emploi, des employées et des cadres d’entreprise.

### Formations existantes

L’école présente actuellement deux cycles : Licence et Master. Chaque cycle est divisé en plusieurs parcours assurés par un chef de mention. L’année universitaire 2013-2014, l’EMIT est basculé totalement vers le système Licence- Master – Doctorat (LMD) et les enseignements qu’elle offre s’organisent par plusieurs mentions à savoir :

* La mention informatique : qui se divise en deux parcours notamment :
  + Grade de licence
    - Le parcours Développement d’Application Internet et Intranet (DA2I).
    - Le parcours Conception Intégration et Gestion des Systèmes d’Information (CIGSI).
  + Grade de master
    - Le parcours Système d’Information Géomatique et de Décision (SIGD).
    - Le parcours Modélisation et Ingénierie Informatique (M2I)
* La mention administration avec :
  + Grade de licence
    - Le parcours : Administration Economique et Sociale (AES).
  + Grade de master
    - Le parcours : Management Décisionnel (MD).
* La mention relation publique et multimédia avec :
  + En grade de licence
    - Le parcours Communication et Multimédia. (CM)
    - Le parcours Relation Publique et Communication Organisationnelle (RPCO)
  + En grade de master
    - Le parcours Communication et Médias Numériques (CMN)
    - Le parcours Relations Publiques et Communications (RPC)

La figure 1.1 montre la liste des formations offertes par l’établissement



Figure 1.1 Offre des formations de l'EMIT

### Organigramme de l’EMIT

L’EMIT compte actuellement 56 enseignants, malagasy et étrangers dans son corps professoral. Répartis dans les différents parcours selon leurs spécialités, ces enseignants travaillent de manière directe avec le personnel administratif de l’école pour offrir la meilleure des formations aux étudiants.

La figure 1.2 montre l’organigramme de l’EMIT

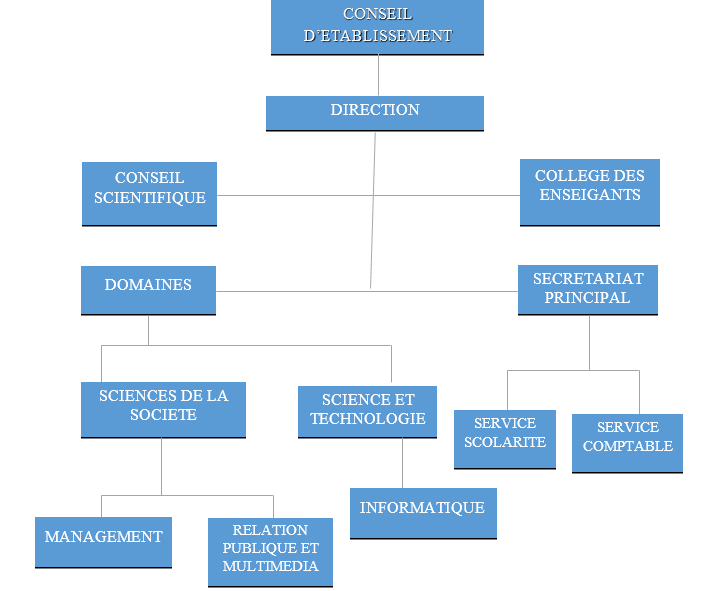


Figure 1.2 Organigramme de l'EMIT

## Chapitre 2 : Présentation de Datascience INnovation Intelligence Knowledge Acquisition

### Création

DINIKA (Datascience- Innovation- Intelligence & Knowledge Acquisition) SARL, est une société anonyme à responsabilité limité (SARL) crée en 2014. Elle est inscrite au registre de commerce sous le numéro 2014 B 00003 et ayant comme numéro d’identité fiscale 4001508721 et statistique 46512 21 2014 0 00 782.

### Siege social

Lot 127B/3405 Mahazengy Fianarantsoa (301) Madagascar

### Mission et vision

Dinika SARL est une jeune entreprise malgache dans le développement d’outils de gestion de débat public. Ayant son siège social à Madagascar, Dinika SARL offre des opportunités de stages et travail pour les jeunes ingénieurs et développeurs de Fianarantsoa, toutes en offrant une prestation de qualité à ses clients.

Nos objectifs :

* Être une entreprise pérenne dans le domaine des nouvelles technologies.
* Répondre aux besoins informatiques et télécommunications des problématiques liées à l’aide au développement à Madagascar (santé...).
* Optimiser ces besoins par le développement d’une infrastructure innovante et opensource dans la circulation des informations numériques par télécommunication (SMS, MMS, Web…).
* Faire de l’argumentation numérique un domaine de recherche pour les thèses

### Les services offerts

* Création et développement de logiciels de gestion de bases de données géographiques et de mise en débat public.
* Ingénierie informatique des débats numériques.
* Amélioration informatiques de la circulation des informations sanitaires des populations rurales et urbaines vers les centres décisionnels et opérationnels (hôpitaux,ministère…) par la création des outils informatiques : exemples logiciels de débats entre la population rurale et les personnels de santé.
* Création de logiciels pour améliorer les collectes de données statistiques des débats numériques

### Les clients

* IRD : installation du plateforme AREN à l’université de Fianarantsoa et réalisation d’un manuel d’installation.
* LIRMM : Développement de nouvelle fonctionnalités sur le plateforme (agenda, vote majoritaire, export et partage des arguments de AREN vers l’influent, mise en relation entre plusieurs plateforme AREN).

### Les projets en cours

* Forum de débat : Hébergement et maintenance d’un débat à huit clos sur une version du plateforme AREN. Modélisation mathématique et développement d’une nouvelle intelligence artificiel dans AREN.
* L’influent : Mise à jour de tous les codes de l’influent.

## Chapitre 3 : Présentation du projet

### Formulation du contexte et de l’origine du projet

Le projet AREN (ARgumentation Et Numérique) vise à élaborer une plateforme numérique de débats en ligne qui favorise le développement des compétences argumentatives des élèves et de leur esprit critique, dans une perspective d’éducation à la citoyenneté. La forme particulière du débat numérique en ligne, fait de brèves interventions argumentatives sur un sujet ciblé, permet une analyse automatique relativement complète en temps réel, objectif hors d’atteinte pour le discours ou le dialogue en général. La faisabilité de ce projet ambitieux est confirmée par l’utilisation d’une première plateforme de débats numériques en ligne « Dialoguea », qui est utilisée depuis novembre 2015 dans divers lycées de l’académie de Montpellier, en particulier en Philosophie et en SVT.

### Problématique

Vu les enjeux sociaux, politiques, économiques et environnementaux du projet, les discussions textuelles semblent insuffisantes pour la production de bons résultats. Il est aussi très difficile d'effectuer des analyses effectives et prendre des décisions à cause du manque de moyens de communication et de représentation des phénomènes étudiés.

Mais comment mettre en place un tel service ? Et quels outils devrons-nous utiliser pour mener à bien ce projet ?

### Objectifs et besoins de l’utilisateur

En prenant en considération les racines qui nous ont conduits à la concrétisation de notre projet, et suite à notre participation à diverses discussions émanant des organes de la plateforme AREN hébergés sur les serveurs de l’EMIT et de l’entreprise DINIKA SARL, nous avons pu définir les buts à atteindre pour ce projet.

En vue d'améliorer l'interaction entre et avec les utilisateurs, l'ajout d'un moyen permettant aux utilisateurs d'effectuer des débats est plus que nécessaire. La présence de supports de discussion telle que les cartes peut aussi être utiles pour l'amélioration de l'expérience utilisateur. Dans cette même optique, il faut aussi trouver des moyens pour faciliter la prise de décision collectif tel que l'intégration d’un système de vote majoritaire.

### Moyens personnels et techniques

#### Moyens personnels

Pour une bonne réalisation, le projet AREN est constitué de membres complémentaires venant de différents domaines d’expertises et des participants intéressés et compétents pour contribuer au débat sur la base de leurs connaissances et de leurs perspectives.

Les compétences en Mathématiques seront fournies par le docteur Antsa Nasandratra NIRINA AVO.

L’animation des débats dirigée par des modérateurs qualifiés, en s'assurant que tous les participants ont la possibilité de contribuer et que le débat reste productif et respectueux.

L’expertise sur les représentations spatiales et de cartographie assurée par une équipe des géographes.

Les compétences en informatique seront assurées par des développeurs qui contribuent à la réalisation de ce projet web, en veillant à ce que le site soit fonctionnel et facile à utiliser pour les utilisateurs finaux.

L’entreprise DINIKA en collaboration avec l’Ecole de Management et d’Innovation Technologique (EMIT) et le Laboratoire d’Informatique, de Robotique et de Microélectronique de Montpellier (LIRMM), qui est une Unité Mixte de Recherche, dépendant conjointement de l’Université Montpellier (UM) et du Centre National de la Recherche Scientifique (CNRS).

#### Moyens matériels et logiciels

Pour la concrétisation du projet, l'équipe AREN dispose de plusieurs ressources matérielles telles que l’ordinateur pour le développement et le teste Des serveurs pour l'hébergement des applications Des téléphones mobiles pour les tests. Des supports de présentations pour les conférences et les ateliers.

Avec ces différents matériels, le projet utilise aussi plusieurs outils logicieles pour faciliter la mise en place de la plateforme.

##### 3.4.2.1. GitHub

Github est une plateforme de gestion de version et de collaboration utilisée par les développeurs logiciels qui repose sur Git. Git est un logiciel de gestion de versions décentralisé. C'est un logiciel libre créé par Linus Torvalds, auteur du [noyau Linux,](https://fr.wikipedia.org/wiki/Noyau_Linux) et distribué selon les termes de la [licence publique générale GNU](https://fr.wikipedia.org/wiki/Licence_publique_générale_GNU) version 2.

En 2016, il s’agit du [logiciel de gestion de versions](https://fr.wikipedia.org/wiki/Logiciel_de_gestion_de_versions) le plus populaire qui est utilisé par plus de douze millions de personnes.

Git possède deux [structures de données](https://fr.wikipedia.org/wiki/Structure_de_données): une base d'objets et un cache de [répertoires.](https://fr.wikipedia.org/wiki/Répertoire_(informatique)) Il existe cinq types d'objets :

* L'objet **Blob** (pour [binary large object](https://fr.wikipedia.org/wiki/Binary_large_object) désignant un ensemble de données brutes), qui représente le contenu d'un [fichier](https://fr.wikipedia.org/wiki/Fichier_informatique);
* L’objet **Tree** (mot anglais signifiant « arbre »), qui décrit une arborescence de fichier. Il est constitué d'une liste d'objets de type *blob*s et des informations qui leur sont associées, tel que le nom du fichier et les [permissions.](https://fr.wikipedia.org/wiki/Droits_sur_un_fichier) Il peut contenir [récursivement](https://fr.wikipedia.org/wiki/Type_récursif) d'autres *trees* pour représenter les sous-répertoires ;
* L’objet **Commit** (résultat de l'opération du même nom signifiant « valider une transaction »), qui correspond à une arborescence de fichiers (*tree*) enrichie de métadonnées comme un message de description, le nom de l'auteur, etc. Il pointe également vers un ou plusieurs objets *commit* parents pour former un graphe d'historiques
* L’objet **Tag** (étiquette) qui est une manière de nommer arbitrairement un commit spécifique pour l'identifier plus facilement. Il est en général utilisé pour marquer certains commits, par exemple par un numéro ou un nom de version.
* L’objet **Branch** (branche) qui contient une partie de l'avancement du projet. Les branches sont souvent utilisées pour avancer dans une partie du projet sans impacter une autre partie.

Git permet de retracer toutes les modifications apportées à notre code source, elle permet aux développeurs de travailler en collaborations et propose des outils de gestion de conflits de version du code source. Notre projet utilise l’outil git et le code source de notre application est hébergé sur Github.

##### 3.4.2.2. Editeur IntelliJ

IntelliJ IDEA est un environnement de développement intégré destiné au développement de logiciels informatiques reposant sur la technologie Java. Il est développé par JetBrains et disponible en deux versions, l'une communautaire, open source, sous licence Apache 2 et l'autre propriétaire, protégée par une licence commerciale. Tous deux supportent les langages de programmation Java, Kotlin, Groovy et Scala.

##### 3.4.2.3. Apache Tomcat

Apache Tomcat est un logiciel de serveur d’applications web multiplateforme, gratuit et open source conçue pour la programmation en Java et développé et maintenu par Jakarta, le groupe de projets open source Java de la fondation Apache. Par ailleurs, les mises à jour du Tomcat sont régulières.

L’objectif initial du logiciel Apache Tomcat est d’héberger et de déployer les servlets Java.

Les servlets sont les programmes Java exécutés du côté serveur et qui reçoivent les requêtes des clients, les interprètent et génèrent les réponses demandées.

L’envoi des résultats peut se faire de façon directe ou via un protocole comme http.

Apache Tomcat fournit les fonctionnalités de base du traitement du serveur web pour les servlets Java. Il couvre tout le cycle de vie des servlets, qui se compose de 3 méthodes :

* init(): qui se charge d’initier la servlet. Cette méthode est exécutée une seule fois, soit lors du démarrage du serveur, soit lorsqu’elle est déclenchée par le client.
* service(): qui se charge de traiter les requêtes clients et générer les réponses adéquates.
* destroy(): la méthode destroy() est invoquée par Tomcat à la fin de l’exécution de la servlet pour nettoyer les traces des activités et libérer les ressources

Tomcat est le logiciel de serveur web préféré des développeurs pour les implémentations Java. La dernière version stable d’Apache Tomcat 9.0.21 est sortie le 7 juin 2019.

D’un point de vue global, Apache Tomcat est chargé de fournir un environnement d’exécution pour les servlets. Il permet donc aux développeurs d’exécuter leurs applications web Java.

D’un point de vue plus détaillé, Tomcat sert à :

* Écouter toutes les requêtes des clients.
* Charger les classes de servlets respectives, en utilisant les mappages de servlets pour traiter les demandes des clients.
* Exécuter les classes de servlets.
* Et enfin, décharger ces classes.

# PARTIE 2 : ANALYSE ET CONCEPTION DU PROJET

## Chapitre 4 : Méthodes et notations utilisés

### Introduction

Il existe plusieurs méthodes pouvant être utilisées pour la réalisation de ce projet. Mais il est idéalement bien d’utiliser celle qui est mieux adaptée.

### La méthode de conduite de projet

Un logiciel est un produit complexe et difficile à réaliser. En effet, l'écriture « d'un petit utilitaire maison » qui répond à des besoins précis mais qui souvent fait appel à une connaissance implicite de l'utilisateur, ne nécessite pas la même organisation matérielle et intellectuelle que la conception et la réalisation d'un logiciel professionnel : un programme comportant plusieurs dizaines, voire centaines de milliers de lignes ne s'improvise pas. Cependant, disposer des moyens matériels et humains ne suffit pas. Une méthode est indispensable. [GAUDEL Marie-Claude et al., 1996]

#### Les méthodologies traditionnelles

Les méthodologies traditionnelles sont les plus couramment employées dans la gestion de projet. L'usage de cahiers des charges très complets est donc impératif. La vérification de chaque étape est réalisée par le responsable de projet. Chaque étape doit être achevée avant de passer à la suivante.

##### La méthode Waterfall ou en cascade

La méthode Waterfall a été créée par le docteur Winston Royce vers les années 1970. C’est une méthode de gestion de projet linéaire et séquentielle dans laquelle le processus de création de logiciels est divisé en plusieurs phases. [Moreau, Charlotte, 2016]

**La spécification :** consiste à définir les besoins et exigences du client et déterminer les fonctionnalités correspondantes. Cette phase sert aussi à identifier l’essence du projet.

**La conception :** consiste à trouver des solutions concrètes pour répondre aux besoins et définir l’architecture globale du projet.

**Programmation et test unitaire :** c’est la phase d’implémentation des fonctionnalités trouvées et leur mise en place dans l’architecture définie. Il y a aussi les tests unitaires pour vérifier le fonctionnement d’un composant indépendamment des autres. [Tremblay G, 2018]

**Intégration et test système :** il s’agit de l’assemblage des composants pour former le produit final. L’intégration est directement suivie par le test système pour vérifier le fonctionnement global du système. [Tremblay G, 2018]

**La production et maintenance :** c’est la phase de diffusion du logiciel construit qui sera suivie par les diverses activités de maintenances.

La figure 4.1 montre le cycle de vie en cascade



Figure 4.1 Le cycle de vie en cascade

##### La méthode en V ou modèle en V

Le modèle en V est une extension du modèle classique waterfall. Le modèle consiste à associer chacune des étapes du modèle en cascade à des niveaux de validations.[Strohmeier A.2000] Suivant la même philosophie que le modèle waterfall, le modèle en V est aussi composé de plusieurs phases :

**L’analyse des besoins** : consiste à définir les besoins et les exigences du client et déterminer la raison d’être du projet.

**La spécification** : consiste à trouver les fonctionnalités correspondantes aux besoins du client et avoir un aperçu sur le fonctionnement global du logiciel à développer.

**La conception architecturale** : a pour but d’identifier les meilleurs architectures logiciel et matériel pour la réalisation du projet.

**La conception détaillée** : revient à établir de façon concrète les solutions répondants aux besoins du client et mettre en place la structure concrète du projet.

**Programmation**: c’est la phase d’implémentation de la structure et des fonctionnalités conçues.

**Les tests unitaires** : premier niveau de test associé à la conception détaillée. Il s’agit de vérifier que les composants codés permettent vraiment de réaliser le fonctionnement attendu de manière indépendante. [Tremblay G, 2018]

**Les tests d’intégration** : test associé à la conception architecturale. Consiste à vérifier l’interaction entre les composants. [D. Longuet ,2017]

**Le test système** : il s’agit de vérifier si le système conçu fonctionne vraiment en accord avec son cahier de charge.

**Les tests d’acceptation** : c’est le dernier test effectué par le client pour vérifier que la solution développée répond vraiment à ses besoins. [Tremblay G, 2018]

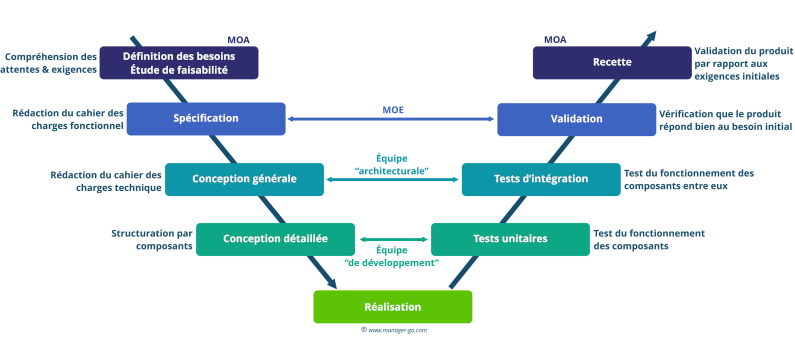
La figure 4.2 montre les diverses phases du modèle en V

Figure 4.2 les diverses phases du modèle en V

*Figure 5.6 : Phases du modèle en V*

Les méthodes traditionnelles reposant sur une bonne planification, les différentes phases de développement d'un logiciel sont préétablies. Les méthodes traditionnelles opèrent de manière séquentielle et sont également faciles à mettre en place, ce qui permet aux équipes qui les utilisent de minimiser les risques d'erreur.

Cependant, le fait que les méthodes de cette catégorie reposent sur des étapes fixes ne permet pas de tenir compte des modifications des besoins des clients. Par conséquent, elles peuvent conduire à des échecs et à des augmentations des coûts de développement.

Les méthodes traditionnelles sont donc adaptées pour les grands projets dont les objectifs sont fixés et définitives depuis le départ avec des cahiers de charges très détaillés.

#### Les méthodologies Agile

En 1986, Barry W. Boehm introduit un nouveau modèle de développement reposant sur une structure commune itérative, incrémentale et adaptative, les méthodes agiles dont le fil conducteur consiste à découper le déroulement d’un projet en plusieurs objectifs plus petits afin d’obtenir plus sûrement et rapidement un résultat (ou minimal vital product – MVP). En 2001, un manifeste introduit quatre valeurs fondamentales qui définissent une nouvelle façon de développer des logiciels en valorisant :

* les individus et leurs interactions plus que les processus et les outils ;
* des logiciels opérationnels plus qu’une documentation exhaustive ;
* la collaboration avec les clients plus que la négociation contractuelle ;
* l’adaptation au changement plus que le suivi d’un plan.

Les méthodes agiles répondent aux méthodes classiques, trop prédictives et trop rigides, en exposant de nouveaux principes plus souples dont l’anticipation, l’auto-régulation, le feedback et la collaboration. Elles renforcent aussi la capacité d’une organisation « apprenante » au changement et à la transformation. [COLLIGNON, Alain et SCHÖPFEL, Joachim, 2016]

##### La méthodologie SCRUM

SCRUM est un outil réactif de développement logiciel et de gestion de projet. Il a été créé en 1993 par Jeff Sutherland dans le but d’avoir une méthodologie de gestion de développement améliorée suivant le principe agile. Il se base sur une stratégie visant à tirer la meilleure des équipes en les mettant au service d’un objectif commun et en divisant le projet en plusieurs sprints bien défini. Globalement l’équipe Scrum est composée de 3 grandes catégories [Permana, P. A. G. ,2015] :

* L’équipe de développement : composé par l’ensemble des entités qui travaillent dans le développement du logiciel.
* Le product owner : qui s’occupe de transmettre les besoins du client à l’équipe et qui travaille dans la transformation des besoins du client en fonctionnalités compréhensible pour l’équipe de développement. C’est donc l’intermédiaire entre l’équipe et le client.
* Scrum master : c’est lui qui dirige l’équipe et qui veille sur le bon déroulement du projet. Il a aussi pour mission de gérer les obstacles et assurer l’application de la méthode Scrum en instaurant une bonne communication entre les membres de l’équipe. [Hasina R., Jérome Q, 2019]

La figure 4.3 montre le cycle de développement de la méthode Scrum.

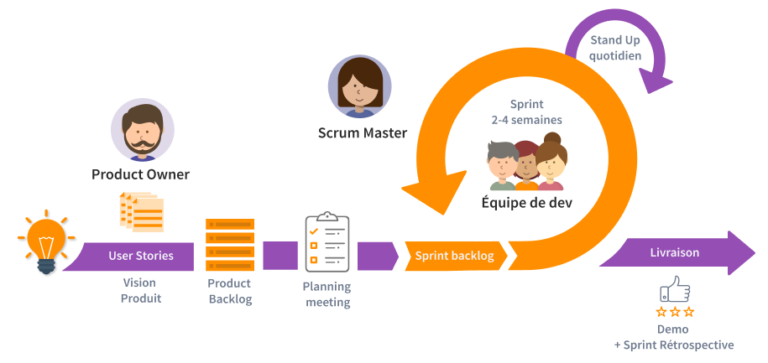


Figure 4.3 le cycle de développement de la méthode Scrum

Le « product owner » développe le backlog, c’est-à-dire repartir le projet en plusieurs petites taches et planifie leur réalisation. Ce baklog est ensuite exposé aux membres de l’équipe et est regroupé en plusieurs sprints. Le Scrum master s’occupe ensuite du suivie de ces sprints et organise les réunions pour le suivie d’avancement. Après la réalisation de chaque sprint, on obtient une version utilisable du projet avec les fonctionnalités développées.

###### Les avantages de la méthodologie Scrum.

Les principaux avantages de la méthodologie Scrum sont :

* Elle permet d’avoir un produit de qualité en respectant le délai prévu.
* Facilite la prise en compte des changements grâce aux sprints de petite taille et aux interactions continues avec les clients.
* Les réunions Scrum régulières permettent d’anticiper les problèmes et mesurer l’évolution (global du projet ainsi que de chacun des membres)
* Scrum est aussi itérative comme toutes les méthodologies agiles et il est facile de s’adapter à l’évolution des besoins du client.

###### Les inconvénients de la méthodologie Scrum

Malgré les avantages cités, Scrum est loin d’être la parfaite méthodologie pour toute gestion de projet. Effectivement, ce style de mise en œuvre agile présente aussi certaines failles.

Le projet dépend beaucoup de chaque collaborateur, donc :

* + - * Si l’un d’être eux ne réalise pas correctement sa part du travail, le projet peut échouer.
      * Si un des membres n’est pas assez compétent, cela peut entrainer un retard sur la production
      * La perte d’un collaborateur a de graves conséquences sur l’évolution du développement du projet. [Saleh et al,2019]

##### La méthodologie Kanban

Kanban est un style de méthodologie agile basé sur le principe de production « Just in Time » de Toyota dans les années 1950. Ce style de production signifie : « ne faire que ce qui est nécessaire, quand c’est nécessaire et juste la quantité nécessaire » [Howard Lei et al., 2017]. Kanban a été introduit dans le développement logiciel en 2004 par David J Anderson dans le but de faciliter le développement en se focalisant sur 5 principes :

* La visualisation du travail et de l’organisation de travail suivi en utilisant le « Virtual kanban system »
* Limitation les taches en cours en utilisant grâce à la principe Kanban.
* Gestion du flux du travail
* Rendre les politiques de gestion de projet clair et précis
* Utilisation des méthodes scientifiques pour améliorer l’équipe et la collaboration [Anderson, 2010]

La figure 4.4 montre un exemple de tableau Kanban.

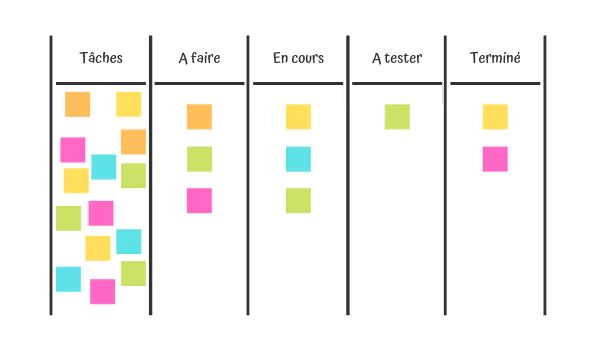


Figure 4.4 Exemple de tableau kanban

###### Les avantages de la méthodologie kanban

Les principaux avantages de l’utilisation de kanban sont les suivants :

* Permets de motiver les équipes en apportant une visualisation directe de l’évolution du travail.
* Facilite la gestion du travail par la visualisation des taches. Permets aussi d’anticiper les obstacles et les retards. » [Nevenka k, Saso k,2015].
* Aide les collaborateurs à évoluer rapidement sur le plan technique
* Réduis les répétitions
* Réduis le risque d’échec du projet.
* Améliore le travail d’équipe et la communication entre collaborateurs
* Réduis les travaux en cours pour plus de productivité.

###### Les inconvénients de Kanban

La méthodologie Kanban présente aussi plusieurs inconvénients.

* Elle est peu efficace en cas de travail avec des ressources distribuées
* Elle accepte le changement de rythme des collaborateurs et cela peut entrainer un retard significatif de la production. [Saleh et al,2017]
* Les liens de dépendances entre les taches ne sont pas clairement identifiés.
* La gestion de l’agenda et des deadlines n’est pas trop visible dans le kanban board
* La mise en place de la méthodologie kanban dans une équipe est très fastidieuse. Il faut un effort supplémentaire pour transmettre la philosophie kanban. [Marikova O.,2008].

##### La méthodologie Extrême Programming

C’est une méthodologie de développement logiciel, flexible et léger qui est particulièrement destinée à répondre aux exigences changeante et imprécise des clients. Elle a été initiée par Beck, Jeffries et al. vers les années 1998 et se base sur 12 principes :

* **La planification** : consiste en une réunion entre les clients, les dirigeants et les développeurs pour présenter (les user story : besoin des clients sur des 'story card'), estimer et arranger les taches selon leur niveau de priorité. Cette réunion se fait au début de chaque itération en utilisant des cartes d’où le nom : « planning poker ».
* **Des petites productions** : consiste à effectuer la production après chaque itération
* **Exemples de cas pratiques** : l’équipe utilise des exemples pratiques pour construire le modèle de l’application.
* **Conceptions simples** : Consiste à garder la conception le plus simple possible. Assure que le code soit clair et explicite, qu’il puisse effectuer toutes les testent possibles, qu’il n’y ait pas de doublons et qu’il y ait le minimum de nombre de classes et de méthodes. [Dietmar p., 2014]
* **Teste** : les tests unitaires, les tests d’intégrations et les tests d’acceptation sont effectués à chaque itération.
* **Simplification de code** : le code doit évoluer au cours du développement pour être le plus simple possible.
* **Développement en duo** : les développeurs travaillent en duo pendant le développement.
* **Intégration continue** : consiste à intégrer directement les changements dans le système à chaque soumission de nouvelles fonctionnalités.
* **Possession collective** : tous les développeurs ont le droit de lecture et d’écriture de toutes les parties du code.
* **Client sur site** : un représentant légitime du client doit travailler sur site avec l’équipe pendant tout le processus de développement.
* **40h de travail par semaine** : les taches doivent être les plus simples et détaillées possibles pour que les développeurs puissent les finir en temps et éviter le « Overtime ».
* **Espace de travail commun** : tous les membres de l’équipe doivent travailler dans une espace de travail commune pour faciliter le dialogue. [David C et al., 2003]

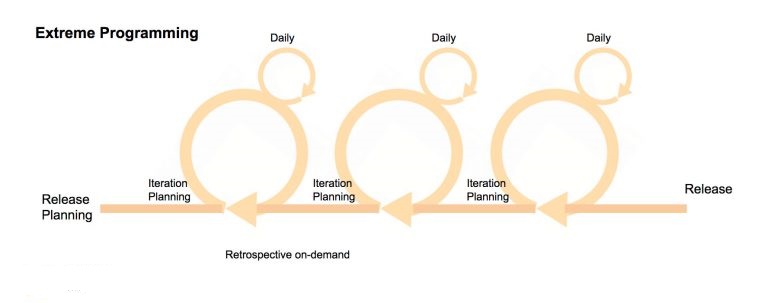
La figure 4.5 montre le cycle de vie de Extrem Programming

Figure 4.5 Cycle de développement de Extrem Programing

Extrem Programming commence par une évaluation du système et des itérations à réaliser par tous les membres de l’équipe. Celle-ci choisit ensuite le « user story » ou besoin client le plus important, le divise en plusieurs fonctionnalités et planifie sa réalisation. Assistés par le client, les développeurs travaillent en un duo de « writer & observer » pour le développement et effectuent les intégrations et les tests. À chaque itération, une nouvelle version du système est produite et le processus de développement continue jusqu’à ce que tous les user stories soient réalisés.[Anwer F, Aftab S., 2017]

###### Avantages de Extrem Programming

Dernièrement, la méthodologie Extrem Programming gagne en popularité, car :

* Elle est flexible et rapide à implémenter
* Elle est centrée sur la satisfaction du client et met ce dernier à la disposition des développeurs pour la réalisation de chaque fonctionnalité [Singh, A. 2018]
* Elle met en avant la discussion entre les membres de l’équipe.
* Elle est orientée sur des tests fréquents et assistés pour assurer la qualité du produit.
* Elle se concentre sur des petites incrémentations pour éviter les risques. [Saleh M et al., 2017]

###### Inconvénients de la méthodologie Extrem Programming

La méthodologie Extrem Programming présente plusieurs inconvénients :

* Les étapes ne sont pas documentées et il faut des efforts supplémentaires pour tout comprendre.
* La méthode ne se concentre pas beaucoup sur la conception et rend difficile la distribution des tâches entre les collaborateurs.
* Elle dépend fortement sur les membres de l’équipe. [Anwer F, Aftab S., 2017]
* Pas très pratique en cas de ressources distribuées et d’équipe de grande taille. (>10)

### Méthodologie choisie

Comme notre mode de travail fonctionne de façon itérative, la méthodologie AGILE répond effectivement la méthodologie requise pour notre projet et notre style de travail. Par conséquent, nous avons opté pour la méthodologie de « Extrem Programming » pour la gestion du développement de notre application.

## Chapitre 5 : Analyse du projet

Dans ce chapitre, nous allons parler de l’analyse du projet. Tout d’abord, l’analyse de besoins, ensuite l’analyse de l’existant dans l’aren puis l’état de l’art et la solution utilisée.

### Définition et contexte de l’AREN

Comme nous avons déjà cité dans la paragraphe précédente, projet AREN ou ARgumentation Et Numérique est une plateforme numérique de débats en ligne qui favorise le développement des compétences argumentatives des élèves et de leur esprit critique, dans une perspective d’éducation à la citoyenneté. Le site web du LIRMM a clairement décrit ce projet : « D’un point de vue pédagogique, l’enjeu est de mettre au point et d’évaluer une plateforme numérique dédiée au débat écrit en ligne qui aide les élèves, futurs citoyens, à développer leur sens de l’argumentation et de la pensée critique : un débat rationnel de qualité est vecteur de démocratie. Du point de vue du numérique, l’enjeu est non seulement de concevoir et de développer ladite plateforme, mais aussi d’inventer ou de renouveler certaines méthodologies de l’intelligence artificielle : interfaces fluides qui guident le débat et organisent les données (IHM), analyse automatique des interventions dans le débat (traitement automatique des langues), analyse automatique des raisonnements dans le débat (logique). La forme particulière du débat numérique en ligne, fait de brèves interventions argumentatives sur un sujet ciblé, permet une analyse automatique relativement complète en temps réel, objectif hors d’atteinte pour le discours ou le dialogue en général. La faisabilité de ce projet ambitieux est confirmée par l’utilisation d’une première plateforme de débats numériques en ligne que nous avons développée : Dialoguea est utilisée depuis novembre 2015 dans divers lycées de l’académie de Montpellier, en particulier en Philosophie et en SVT. La plateforme actuelle, à laquelle les élèves se connectent via un serveur web, permet déjà une certaine structuration du débat, et son intérêt pédagogique est déjà perceptible dans les expériences réalisées. »

### But de l’AREN

Un outil numérique permet de garder la trace écrite des échanges, et ainsi à la fois de prendre conscience de l’argumentation déployée et de l’analyser dans le cadre d’un travail réflexif métacognitif, elle offre ainsi de nouvelles possibilités pour mener des débats en classe. [Gabriel Pallarès, 2020]

La plateforme a pour but :

* Améliorer les collectes de données statistiques des débats numériques.
* Offrir des opportunites de developpement des competences argumentatives et les esprit critique des participats aux debats.
* Développer une vision plus riche de la nature des connaissances, de leurs modes de production et de validation (importance des faits ou des données empiriques, prise en compte des incertitudes et du caractère évolutif des connaissances, dimension collective de la construction des connaissances, etc).

### Analyse des besoins

Les utilisateurs ont donc besoin l'ajout d'un moyen permettant d'effectuer des débats à l’aide d’une présence de supports de discussion telle que les cartes avec l'amélioration de l'expérience utilisateur. Et aussi, l’intégration des moyens pour faciliter la prise de décision collectif, c’est le système de vote majoritaire.

### Analyse de l’existant dans l’AREN

Le projet AREN ou ARgumentation Et Numérique est une plateforme numérique de débats en ligne qui favorise le développement des compétences argumentatives des élèves et de leur esprit critique, dans une perspective d’éducation à la citoyenneté. Elle est maintenue et développée par une communauté de développeurs engagés.

La plateforme possède des diverses fonctionnalités tels que :

* **Debat : Effectuer un debat sur un sujet ou theme choisi qui est publié par un administrateur.**
* **Notification** : soyez notifié d'un nouveau debat.
* **Exporter** : récupérer les détails d’un débat dans un fichier ODT.

### Etat de l’art

Nous présentons dans cette section quelques plateformes de dialogues et d’un vote majoritaire. Il est évident que chaque plateforme possède ses propres atouts et faiblesses, mais en règle générale, ils sont spécifiquement adaptés à la tâche qu'ils cherchent à accomplir et ne sont pas génériques.

dialogue homme-

machine (DHM)

dialogue homme-

machine (DHM)

le dialogue homme-

machine (DHM)

le dialogue homme-

machine (DHM)

#### Principe de débat

Le débat est une « discussion généralement animée entre interlocuteurs exposant souvent des idées opposées sur un sujet donné. » Il peut aussi être un échange autour d’un même point de vue, mais en lui apportant des nuances.

#### Principe de vote majoritaire

Le jugement majoritaire est une nouvelle méthode de vote inventée par deux directeurs de recherche du CNRS, Rida Laraki et Michel Balinski. La proposition d'introduire le vote majoritaire comme nouveau mode de scrutin repose sur deux constats. L'expression d'une opinion par le scrutin majoritaire ne reflète pas nécessairement une préférence collective. Chaque fois qu'il y a plus de deux candidats en lice, les résultats du vote basés sur le paradoxe Condorcet-Arrow peuvent être incohérents. Selon Balinski et Laraki, lorsque les candidats sont jugés sur la base de mesures communes du rang de nomination, le vote identifie toujours le vainqueur, neutralise le paradoxe Condorcet-Arrow et réduit la probabilité d'une élection. Par conséquent, un vote majoritaire devrait permettre aux électeurs d'identifier les candidats qu'ils veulent vraiment. Cela prend en compte les opinions de tous les électeurs sur tous les candidats et donne aux électeurs une totale liberté d'exprimer leurs opinions. [Michel Balinski, Rida Laraki, 2010]

Le jugement majoritaire agrège les informations relatives aux opinions des électeurs en identifiant, pour chaque candidat, sa "mention majoritaire". Qu'est-ce que la mention majoritaire ? L'électeur vote en évaluant tous les candidats, à partir d'une échelle de valeur (par exemple : Excellent, Très bien, Bien, Assez bien, Passable, Insuffisant, À rejeter). Le candidat le mieux évalué par une majorité remporte l’élection. Prenons donc un candidat donné. Un certain nombre d'électeurs lui auront attribué la mention "à rejeter", un certain nombre d'autres électeurs lui auront attribué la mention "insuffisant", d'autres encore la mention "passable", et ainsi de suite jusqu'à la mention "excellent". La mention majoritaire, c'est la mention "globale" du candidat. Rien n'est plus simple que de la calculer. Il suffit de mettre toutes les mentions attribuées au candidat par les électeurs dans l'ordre, de "à rejeter" vers "excellent", et de prendre celle du milieu.

#### Principe de débat sur carte

Le principe du débat cartographique est une méthode de discussion qui implique l'utilisation de cartes pour faciliter la communication et l'échange d'informations entre les participants. Cette approche peut être utilisée pour discuter de problèmes locaux, de problèmes environnementaux, de projets de développement ou d'autres problèmes d'importance géographique. Dans les débats sur les cartes, les participants utilisent des cartes pour visualiser les informations et les perspectives des autres participants. Ils peuvent utiliser des marqueurs pour mettre en évidence des domaines d'intérêt ou des problèmes spécifiques, ou pour suggérer des idées pour résoudre des problèmes.

#### Dialoguea

C’est une plateforme de débat numérique en ligne. Elle est utilisée depuis novembre 2015 dans divers lycées de l’académie de Montpellier, en particulier en Philosophie et en SVT. La plateforme actuelle, à laquelle les élèves se connectent via un serveur web, permet déjà une certaine structuration du débat, et son intérêt pédagogique est déjà perceptible dans les expériences réalisées.

#### mieuxvoter.fr

C’est une plateforme qui propose des informations et des outils pour aider les citoyens à comprendre et à utiliser des méthodes de vote alternatives pour les élections. Ces méthodes de vote comprennent notamment le vote par approbation, le vote pondéré et le majoritaire. Le site fournit également des ressources pour les électeurs, les candidats et les organisations qui souhaitent promouvoir des systèmes de vote plus justes et plus efficaces.

Par des actions de plaidoyer et des campagnes de mobilisation, Mieux Voter agit pour faire connaître le jugement majoritaire au plus grand nombre et accompagne les collectivités publiques, les entreprises, les associations et les particuliers dans son utilisation.

### Proposition des solutions

Selon les analyses effectuées, on peut intégrer une nouvelle fonctionnalité dans l’AREN, c’est le système de vote majoritaire pour faciliter la prise de décision collectif, et une fonctionnalité permettant d’effectuer un débat sur une carte, pour faciliter l’argumentation liées à des enjeux géographiques.

## Chapitre 6 : Conception du projet

### Dictionnaire de données

Le dictionnaire de données est l’inventaire des entités et de leurs attributs respectifs tirés des collectes des données, il regroupe aussi la structure de toutes les données que l’on va modéliser.

Tableau 6.1 : Dictionnaire de données

Tableau 6.1 : Dictionnaire de données

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| RUBRIQUE | DESCRIPTION | STRUCTURE | | |
| TYPE | TAILLE | FORMAT/VALEURS |
| email | Email | AN |  |  |
| username | Nom de l’utilisateur | AN |  |  |
| authority | Rôles de l’utilisateurs | AN |  | USER, ADMIN, SUPERADMIN |
| password | Mot de passe utilisateur | AN |  |  |
| Vm\_themes | | | | |
| authorId | Clé de l’utilisateur | AN |  |  |
| teamId | Clé du team (groupe d’utilisateur) | AN |  |  |
| title | Titre du theme | AN |  |  |
| description | Description du theme | AN |  |  |
| createdAt | Date de creation | Date |  |  |
| expiracyDate | Date limite pour pouvoir effectuer du vote | Date |  |  |
| Vm\_choices | | | | |
| themeId | Clé du vm\_theme | N |  |  |
| title | Titre du choix ou sous-theme | AN |  |  |
| description | Description du choix | AN |  |  |
| url | Lien | AN |  |  |
| img | Lien de l’image | AN |  |  |
| rejected | Nombre de la mention rejeté | N |  |  |
| insufficient | Nombre de la mention insufficient | N |  |  |
| pass | Nombre de la mention passable | N |  |  |
| acceptable | Nombre de la mention acceptable | N |  |  |
| good | Nombre de la mention bien | N |  |  |
| excellent | Nombre de la mention excellet | N |  |  |
| createdAt | Date de creation | Date |  |  |
| Vm\_votes | | | | |
| subThemeId | Clé du vm\_choices | AN |  |  |
| authorId | Cle de l’utilisateur qui effectue le vote | AN |  |  |
| opinion | Choix du vote | AN |  | REJECTED, INSUFFICIENT, PASS, ACCEPTABLE, GOOD, VERY\_GOOD, EXCELLENT |
| createdAt | Date de creation | Date |  |  |

### Les règles de gestions

Les exigences doivent être écrites de telle manière qu'elles orientent la création et la modification d'un système selon les règles métier (ou règles de gestion) appropriées au contexte et au domaine et dans lequel le système doit être utilisé.

Avant de modéliser le système, il est donc important de se renseigner sur les exigences qui régissent les entités de la passation de marché, les règles de gestion sont la formulation de ses spécificités en phrases qui seront ensuite appliquées lors de la modélisation. La liste des règles de gestion est :

RG1 : Seul l’administrateur peut lancer un nouveau sujet à voter

RG2 : Seul l’administrateur peut lancer un nouveau débat

RG3 : Tous les utilisateurs invités peuvent participer à l’argumentation sur un débat.

RG4 : Tous les utilisateurs authentifiés peuvent voter dans le vote majoritaire.

RG5 : l’utilisateur peut consulter les débats existants.

### Diagramme UML du projet

#### Diagramme de cas d’utilisation

En langage UML, les diagrammes de cas d'utilisation modélisent le comportement d'un système et permettent de capturer les exigences du système.

Les diagrammes de cas d'utilisation décrivent les fonctions générales et la portée d'un système. Ces diagrammes identifient également les interactions entre le système et ses acteurs. Les cas d'utilisation et les acteurs dans les diagrammes de cas d'utilisation décrivent ce que le système fait et comment les acteurs l'utilisent, mais ne montrent pas comment le système fonctionne en interne.

Les diagrammes de cas d'utilisation illustrent et définissent le contexte et les exigences d'un système entier, ou des parties essentielles d'un système. Vous pouvez modéliser un système complexe avec un seul diagramme de cas d'utilisation, ou créer de nombreux diagrammes de cas d'utilisation pour modéliser les composants du système. Vous développerez des diagrammes de cas d'utilisation essentiellement dans les premières phases d'un projet et vous vous y référerez tout au long du processus de développement. [[3]](#_Bibliographie)

La figure 6.1 nous montre le diagramme de cas d’utilisation du projet avec l’outil Visual paradigme.

Dans cette application, il y a deux types d’utilisateurs :

* L’administrateur : C’est celui qui a le pouvoir de lance un nouveau débat et un sondage, de définir les rôles de chaque utilisateur.
* L’utilisateur : C’est l’utilisateur simple ne peut que participer à l’argumentation sur des débat lancés et ne peut que participer sur le sondage.

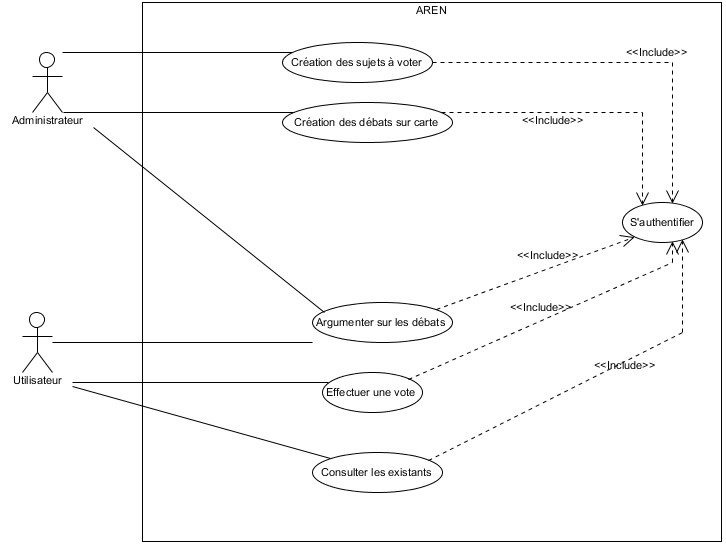


Figure 6.1 : Diagramme de cas d’utilisation

#### Diagramme d’activités

Un diagramme d'activité fournit une vue du comportement d'un système en décrivant la séquence d'actions d'un processus. Les diagrammes d'activité sont similaires aux organigrammes de traitement de l'information, car ils montrent les flux entre les actions dans une activité. Les diagrammes d'activité peuvent, cependant, aussi montrer les flux parallèles simultanés et les flux de remplacement.

Dans les diagrammes d'activité, vous utilisez des nœuds d'activité et des bords d'activité pour modéliser le flux de commande et de données entre les actions. [[4]](#_Bibliographie)

##### Cas d’utilisation « s’authentifier »

Avant d’effectuer toutes opérations dans la plateforme AREN, un utilisateur doit s’authentifier.

La figure 6.2 montre le diagramme d'activité du cas d'utilisation « s’authentifier »

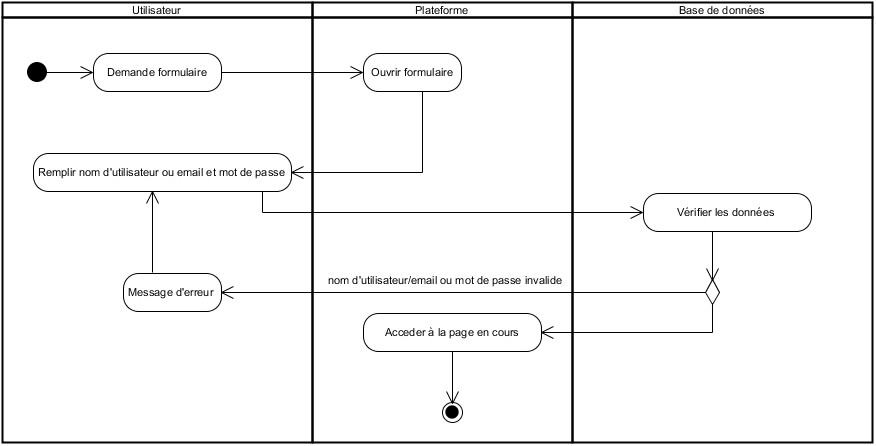


Figure 6.2 : Diagramme d'activité du cas d'utilisation « s’authentifier »

Tableau 6.2: Description textuelle du cas d'utilisation "s'authentifier"

|  |  |
| --- | --- |
| Ligne | Description |
| Cas d’utilisation | S’authentifier |
| Précondition | Avoir un compte existant |
| Scénario nominal | Ouvrir formulaire d’authentification  Saisir nom utilisateur ou adresse email et mot et mot de passe Validation des données  Accès aux pages |
| Scénario alternatif | Ouvrir formulaire d’authentification  Saisir nom utilisateur ou email et mot et mot de passe Validation des données  Données erronées  Affichage d’un message d’erreur |
| Postcondition | Ouverture d’un formulaire ou mode édition activé |

##### Cas d’utilisation « Création vote majoritaire »

La figure 6.3 illustre le digramme d’activité du cas d’utilisation "Création du vote majoritaire"

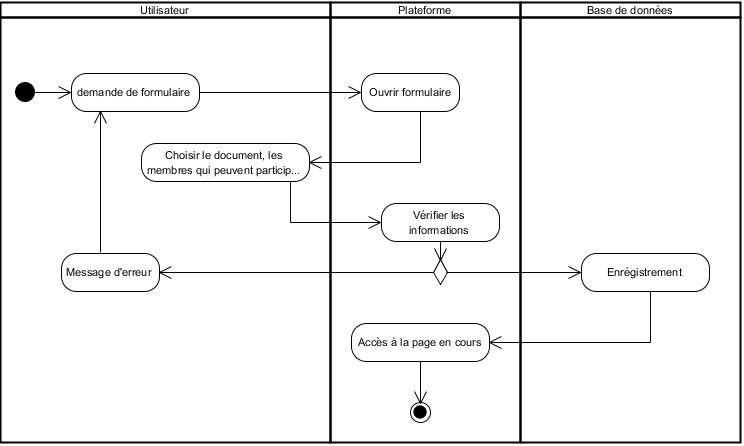


Figure 6.3 : Diagramme d'activité du cas d'utilisation "Création du vote majoritaire"

Tableau 6.3: Description textuelle du cas d'utilisation " Création du vote majoritaire "

|  |  |
| --- | --- |
| Ligne | Description |
| Cas d’utilisation | Création du vote majoritaire |
| Précondition | Être connecté en tant qu’administrateur |
| Scénario nominal | Mode édition active  Edition  Enregistrement avec succès |
| Scenario alternatif | Affichage d’un message d’erreur |
| Postcondition | Ajout terminé avec succès |

##### Cas d’utilisation « Création des débats sur carte »

La figure 6.4 illustre le digramme d’activité du cas d’utilisation " Création des débats sur carte"

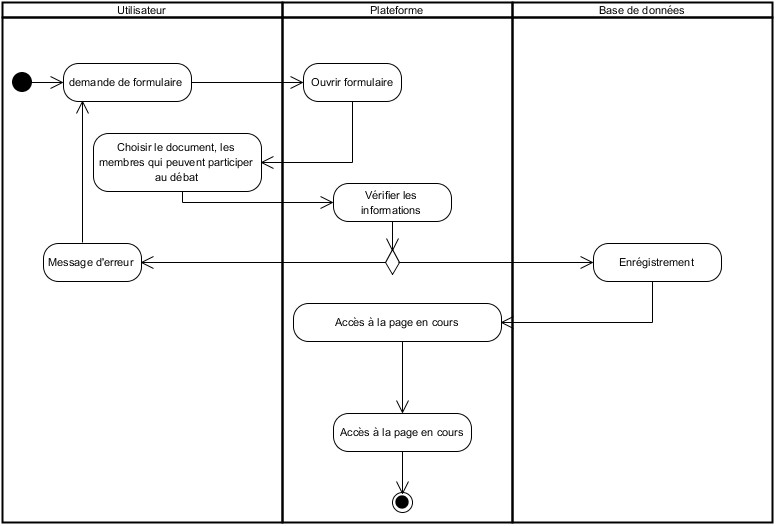


Figure 6.4 : Diagramme d'activité du cas d'utilisation " Création des débats sur carte"

Tableau 6.4: Description textuelle du cas d'utilisation " Création des débats sur carte"

|  |  |
| --- | --- |
| Ligne | Description |
| Cas d’utilisation | Création des débats sur carte |
| Précondition | Être connecté en tant qu’administrateur |
| Scénario nominal | Mode édition active  Edition  Enregistrement avec succès |
| Scenario alternatif | Affichage d’un message d’erreur |
| Postcondition | Ajout terminé avec succès |

##### Cas d’utilisation « Argumenter sur les débats »

La figure 6.5 illustre le digramme d’activité du cas d’utilisation "Argumenter sur les débats"

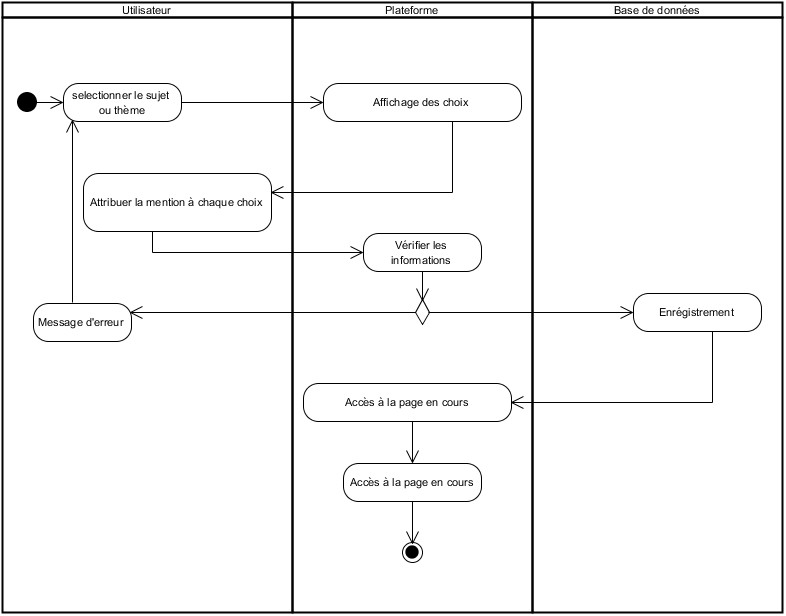


Figure 6.5 : Diagramme d'activité du cas d'utilisation " Argumenter sur les débats "

Tableau 6.5: Description textuelle du cas d'utilisation " Argumenter sur les débats "

|  |  |
| --- | --- |
| Ligne | Description |
| Cas d’utilisation | Argumenter sur les débats |
| Précondition | Être connecté |
| Scénario nominal | Ouvrir un débat existant  Sélectionner un argument existant ou un texte ou encore argumenter directement.  Validation des données |
| Scénario alternatif | Affichage d’un message d’erreur |
| Postcondition | Accès à la page en cours |

##### Cas d’utilisation « Effectuer une vote »

La figure 6.6 illustre le digramme d’activité du cas d’utilisation "Effectuer une vote"

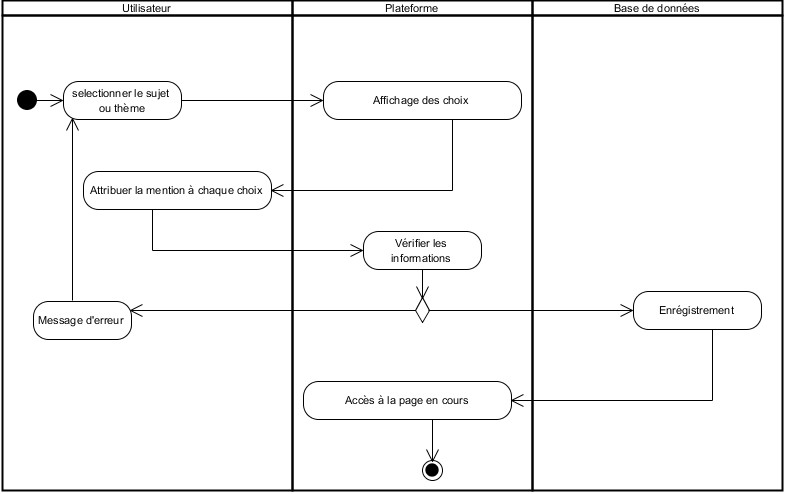


Figure 6.6 : Diagramme d'activité du cas d'utilisation " Effectuer une vote"

Tableau 6.6: Description textuelle du cas d'utilisation " Effectuer un vote "

|  |  |
| --- | --- |
| Ligne | Description |
| Cas d’utilisation | Effectuer un vote |
| Précondition | Être connecté |
| Scénario nominal | Ouvrir un sondage existant  Sélectionner la mention sur le choix ou le candidat à voter.  Validation des données |
| Scénario alternatif | Affichage d’un message d’erreur |
| Postcondition | Accès à la page en cours |

##### Cas d’utilisation « Consulter les existants »

La figure 6.7 illustre le digramme d’activité du cas d’utilisation "Consulter les existants"

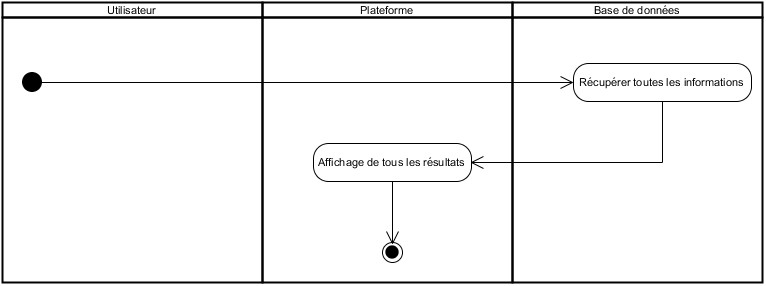


Figure 6.7 : Diagramme d'activité du cas d'utilisation " Consulter les existants "

Tableau 6.7: Description textuelle du cas d'utilisation " Consulter les existants "

|  |  |
| --- | --- |
| Ligne | Description |
| Cas d’utilisation | Consulter les existants |
| Précondition | Être connecté |
| Scénario nominal | Ouvrir sur le menu à consulter  Affichage des résultats |
| Postcondition | Affichage des résultats |

#### Diagramme de classe

Le diagramme de classes a pour objet de mettre en évidence les classes d’un système avec les relations qui les associent, notamment, des compositions et des généralisations. [BADIR, Hassan.]

La figure 6.7 montre le diagramme de classe

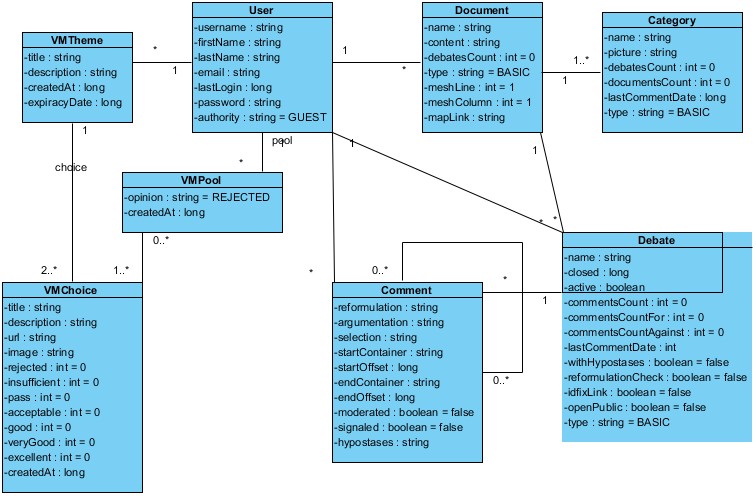


Figure 6.8 Le diagramme de classe de l'application

# PARTIE 3 : REALISATION DU PROJET

## Chapitre 7 : Spécification des outils utilisés

### Introduction

Pour mener à bien un projet et optimiser l’utilisation des ressources (temps et personnels), il est nécessaire de bien choisir les outils de mise en œuvre. Dans le cas du projet AREN, nous effectuons une maintenance d’une application existante. De ce fait, les outils de réalisations nous ont tous été prescrites.

### Langage de modélisation

UML est un langage de modélisation orienté objet qui a été mise en place par Grady Booch, Ivar Jacobson et Jim Rumbaugh pour répondre à la demande de l’Object Management Group (OMG) concernant la mise en place d’une notation de modélisation standard. La version 1.0 du langage c’est-à-dire le UML 1.0 est sortie en 1996.La version 2.0 qui est la plus utilisée n’est apparue qu’en 2005.

UML utilise des représentations graphiques ou diagrammes pour appuyer et illustrer les diverses étapes de la modélisation d’un système. UML propose 13 diagrammes reparties en 2 groupes dont :

• 6 diagrammes Structuraux ou statiques et

• 7 diagrammes comportementaux ou dynamique.

Les principaux avantages de l’utilisation d’UML pour la modélisation d’un logiciel ou d’un système sont :

• UML est une notation normalisée donc il permet aux développeurs utilisant différente technologie d’avoir une norme pour communiquer.

• UML permet d’avoir une modélisation stable

• UML donne un aperçu du logiciel à développer c’est-à-dire, qu’après avoir effectué tous les diagrammes, il est très facile de passer au développement du logiciel spécifié.

• La majorité des outils UML offrent aujourd’hui des ORM permettant de générer le schéma de la base de données correspondant au système.

### Langages de programmations

Le « core » du projet AREN est écrite en langage java et javascript. La partie serveur c’est-à-dire le backend ou le cerveau de l’application est écrite en JAVA tandis que Javascript gère le coté présentation.

#### Le langage java

Historiquement, le langage Java est un langage généraliste de programmation synthétisant les principaux langages existants lors de sa création en 1995 par **James Gosling** et **Patrick Naughton,** employé de la Société Sun Microsystems, désormais appartient à l’Oracle après un rachat effectué en 2009. Il permet une programmation orientée-objet modulaire et reprend une syntaxe très proche de celle du langage C. Outre son orientation objet, le langage Java a l’avantage d’être modulaire (on peut écrire des portions de code génériques, c.-à-d. utilisables par plusieurs applications), rigoureux (la plupart des erreurs se produisent à la compilation et non à l’exécution) et portable (un même programme compilé peut s’exécuter sur différents environnements). En contrepartie, les applications Java ont le défaut d’être plus lentes à l’exécution que des applications programmées en C ou C++ par exemple.

Par contre, les applications Java sont lentes à l’exécution par rapport aux applications programmés par autres langages compilés tels que C, C++, c#... . La cause en est que, les codes sources écrit, sous la forme de classes, dans des fichiers dont l’extension est .java sont compilés par le compilateur du java en un langage appelé bytecode et enregistre le résultat dans un fichier dont l’extension est .class. Le bytecode ainsi obtenu n’est pas directement utilisable. Il doit être interprété par la machine virtuelle de Java qui transforme alors le code compilé en code machine compréhensible par le système d’exploitation. C’est la raison pour laquelle Java est un langage portable : le bytecode reste le même quel que soit l’environnement d’exécution.

La figure ci-dessous illustre ce fonctionnement :

Figure 7.1 : Cursus d’exécution d’un programme java

#### Le langage javascript

JavaScript désigne un langage de développement informatique, et plus précisément un langage de script orienté objet. On le retrouve principalement dans les pages Internet. Il permet, entre autres, d'introduire sur une page web ou HTML des petites animations ou des effets.

Créé en 1995 par Brendan Eich, en même temps que la technologie Java, le langage JavaScript se distingue des langages serveurs par le fait que l'exécution des tâches est opérée par le navigateur lui-même, sur l'ordinateur de l'utilisateur, et non sur le serveur web. Il s'active donc généralement sur le poste client plutôt que côté serveur.

Le langage JavaScript est principalement employé pour améliorer l'ergonomie d'un site Internet et/ou d'une interface applicative utilisateur. Il sert également à intégrer des effets esthétiques, toutefois rarement indispensables. Son principal intérêt réside dans son mode de fonctionnement : le langage JavaScript offre en effet la possibilité d'exécuter un code sans être contraint de recharger une page web. En cela, il joue un rôle dans l'amélioration de la rapidité de chargement des pages, un critère d'ergonomie et de référencement de plus en plus important.

### Framework d’affichage

Comme mentionnées dans les paragraphes précédents, Javascript est un langage largement utiliser pour dynamiser la présentation d’un application web **[Sen, Koushik, et al., 2013].**

Dans le cas d’un application web, il existe plusieurs outils permettant de faciliter l’utilisation du Javascript pour l’écriture des partie clientes : les « frameworks ».

Dans le cadre du projet AREN, le langage Javascript va être appuyé par le framework VueJs pour faciliter le développement et dans ce même optique, le framework CSS Materialize est aussi utilisé pour la mise ne forme.

#### Framework VueJs

Vue est un framework évolutif écrit en javascript pour construire des interfaces utilisateur. À la différence des autres frameworks monolithiques, Vue a été conçu et pensé pour pouvoir être adopté de manière incrémentale. Le cœur de la bibliothèque se concentre uniquement sur la partie vue, et il est vraiment simple de l’intégrer avec d’autres bibliothèques ou projets existants. D’un autre côté, Vue est tout à fait capable de faire tourner des applications web monopages quand il est couplé avec des outils modernes et des bibliothèques complémentaires.

#### Materialize

Materialize CSS est un framework CSS orienté front-end, qui s'appuie sur les principes de Material Design de Google. Il vous permet de créer des sites web responsives afin de gagner du temps lors de vos différentes mises en page d'interfaces utilisateurs. Il fournit un système de grille réactif, ainsi que des composants préconçus qui sont adaptés aux mobiles. Ce framework dispose d'un grand nombre d'éléments designs et prêts-à-être intégrés. Il est d'ailleurs très facile à utiliser car il est basé sur une grille à 12 colonnes qui rend les mises en page beaucoup plus simples, comme certains autres Frameworks tel que Bootstrap.

### Le Système de gestion de base de données.

Comme l’application AREN est un outil de communication et de débat. Plusieurs données sont utilisées et analysées pour faciliter la prise de décision. Ces données doivent être conservées et stockées pour des éventuelles réutilisations. Ainsi, l’application utilise un SGBD notamment le PostgreSQL pour la gestion de ses données.

#### La base de données PostgreSQL

PostgreSQL est un système de gestion de base de données relationnelle orienté objet puissant et open source qui est capable de prendre en charge en toute sécurité les charges de travail de données les plus complexes. Alors que MySQL donne la priorité à l'évolutivité et aux performances, PostgreSQL donne la priorité à la conformité et à l'extensibilité SQL.

#### Avantages d’utiliser PostgreSQL

* Il fonctionne sur les principaux systèmes d'exploitation : Linux, UNIX (AIX, BSD, HP-UX, SGI IRIX, Mac OS X, Solaris, Tru64) et Windows.
* Il est extrêmement respectueux des standards, se conformant au plus près à la norme ANSI-SQL 2008. Il supporte plus d'une douzaine de langages de programmation, dont Java, Perl, Python, Ruby, Tcl, C / C ++. Son propre PL / pgSQL est similaire à PL / SQL d'Oracle.
* Open-source, il est développé par une communauté mondiale regroupant des milliers de développeurs et plusieurs dizaines d'entreprises.

### L’ORM utilisé

ORM signifie Object-Relational Mapping. C’est un ensemble de classes permettant de manipuler les tables d’une base de données relationnelle comme s’il s’agissait d’objets. Un ORM est une couche d’abstraction d’accès à la base de données qui donne l’illusion de ne plus travailler avec des requêtes, mais de manipuler des objets. [O'Neil, E.J, 2008].

Pour faciliter l’interaction du serveur avec la base de données, nous avons utilisé Hibernate comme ORM.

Hibernate est une solution open source de type ORM (Object Relational Mapping) qui permet de faciliter le développement de la couche persistance d'une application. Hibernate permet donc de représenter une base de données en objets Java et vice versa. [[5]](#_Bibliographie)

Hibernate facilite la persistance et la recherche de données dans une base de données en réalisant lui-même la création des objets et les traitements de remplissage de ceux-ci en accédant à la base de données. La quantité de code ainsi épargnée est très importante d'autant que ce code est généralement fastidieux et redondant.

Hibernate est très populaire notamment à cause de ses bonnes performances et de son ouverture à de nombreuses bases de données.

Les bases de données supportées sont les principales du marché : DB2, Oracle, MySQL, PostgreSQL, Sybase, SQL Server, Sap DB, Interbase,…

## Chapitre 8. Mise en œuvre et implémentation

### Architecture logicielle [Julie V. et Yann-Gaël G., 2008]

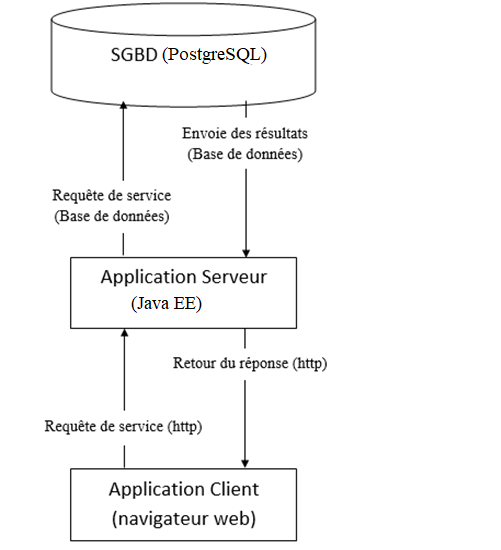
L’architecture logiciel d’une application est la représentation structurée des divers modules constituant l’application. Ella expose très clairement les interactions entre les composants de l’application et ainsi facilite la compréhension du fonctionnement de ces derniers.

Une architecture logicielle contient :

• Les divers composants de l’application

• Les propriétés externes de ces composants

• Les relations entre les composants.

La figure 8.1 montre l’architecture logiciel de l’application.

A l’aide de son navigateur, le client envoie des requêtes pour demander telles ou telles informations. Cette requête sera transmise au serveur Apache comme une demande d’affichage d’une page web. Le serveur Apache lit aussi les paramètres passés avec la demande de la page web. Ainsi le serveur Apache envoie à son tour des requêtes de services de Base de Données vers le serveur pour avoir les données demandées par le client. Enfin le serveur de base de données envoie les données souhaitées par le client vers le serveur Apache, ce dernier s’occupe de la synchronisation entre la page demandée et les données obtenue et envoie le tout vers le client.

### Architecture matérielle de l’application [Julie V. et Yann-Gaël G., 2008]

L’architecture matériel décrit l’interaction entre les divers matériels mise en œuvre pour le fonctionnement de l’application. Notre application nécessite certains dispositifs pour bien marcher.

La figure 8.2 montre le strict minimum pour le bon fonctionnement de notre application.

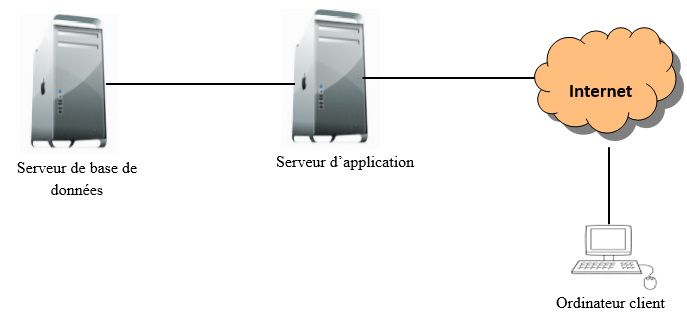


Figure 8.2 : Architecture matériel de l’application

Le client est connecté à son point d’accès internet pour pouvoir envoyer et recevoir des réponses venant du serveur de la base de données. Le serveur d’application assure la liaison entre l’ordinateur client et le serveur de la base de données par le biais des requêtes et des réponses https.

## Chapitre 9 : Présentation du logiciel

### Introduction

Dans ce chapitre qu’on va montrer certaines interfaces graphiques essentiels sur les fonctionnalités récemment développées dans l'application web.

### Création du vote majoritaire

Le vote majoritaire a été conçu par deux chercheurs français du Centre national de la recherche scientifique (CNRS), Michel Balinski et Rida Laraki1, pour répondre aux exigences de Arrow. L’idée est dans le cheminement philosophique de Condorcet et de Borda : il faut donner aux électeurs encore plus de liberté, c’est-à-dire non seulement comparer ou classer les candidats mais juger les mérites de chacun. Cela suppose l’existence d’une échelle d’évaluation qui sert à mesurer les mérites des candidats. Nous l’appelons un langage commun.

La figure 9.1 nous montre l’interface graphique de la page permettant de créer un vote majoritaire

Figure 9.1 : Interface graphique de la page permettant de créer un vote majoritaire

Il suffit de sélectionner un groupe contenant un certain utilisateur, ce qui permet aux membres de participer au scrutin. Puis le titre, suivi de la description complète du vote à faire afin que les utilisateurs puissent comprendre facilement en ce qui concerne le vote. Sans oublier de préciser la date limite du vote, et surtout les choix sur lesquels nous choisissons.

Il est possible d’ajouter plusieurs choix selon le besoin.

### Effectuer un vote

Le principe du vote majoritaire est simple. Le ou les candidats qui obtiennent la majorité des suffrages exprimés sont élus. Il s'agit donc de confier le soin de représenter l'ensemble d'une circonscription aux candidats qui arrivent en tête sans tenir compte des suffrages recueillis par ses concurrents.

La figure 9.2 nous montre l’interface graphique de la page permettant d’effectuer un vote

Figure 9.2 : Interface graphique de la page permettant d’effectuer un vote

Il suffit de sélectionner la mention de notre choix sur le candidat ou choix présent avant de le valider. Dans cette capture, l’électeur a attribué la mention "à rejeter" pour les deux candidats. Au cas où nous n’avons pas sélectionné aucune mention, la valeur par défaut sera « à rejeter ».

### Création du document d’un débat sur carte

La figure 9.3 nous montre l’interface graphique de la page permettant de créer un nouveau document pour un débat sur carte.

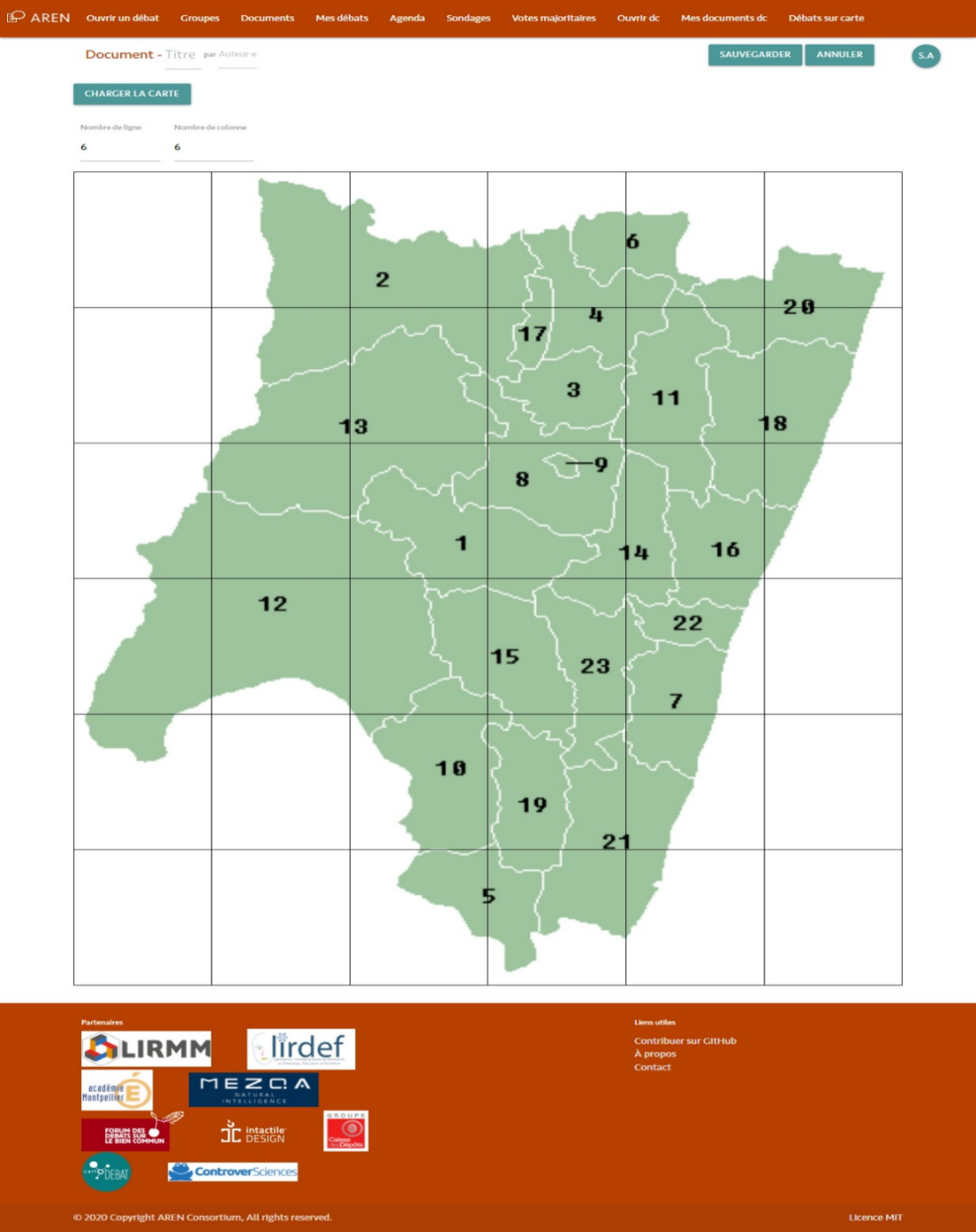


Figure 9.3 : Interface graphique de la page permettant de créer un nouveau document pour un débat sur carte.

Après avoir chargé la carte au format image, on indique le nombre de maille pour la carte à l’aide du champ « nombre de ligne » et « nombre de colonne ».

### Sélection des participants au débat sur carte

La figure 9.4 nous montre l’interface graphique de la page montrant la première étape de la création du débat sur carte.



Figure 9.4 : Interface graphique de la page montrant la première étape de la création du débat sur carte.

La première étape de la création du débat sur carte consiste à sélectionner le(s) groupe(s) contenant des utilisateurs choisis, qui pourront par la suite participer aux argumentations sur le débat.

### Sélection du document pour débat sur carte

La figure 9.5 nous montre l’interface graphique de la page montrant la deuxième étape de la création du débat sur carte.

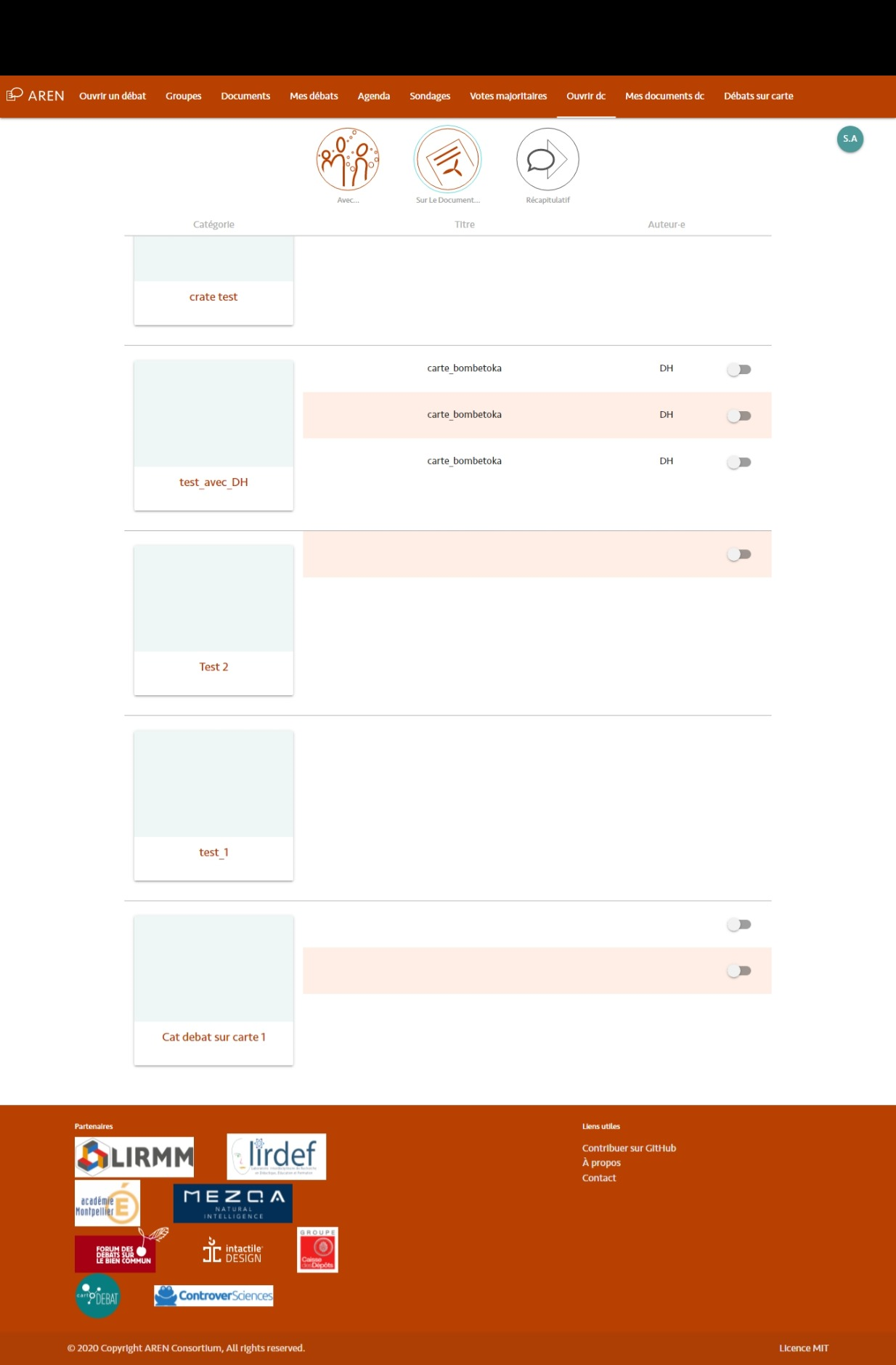


Figure 9.5 : Interface graphique de la page montrant la deuxième étape de la création du débat sur carte.

La deuxième étape de la création du débat sur carte consiste à sélectionner le document sur lequel le débat en cours va être utiliser.

### Liste des débats sur carte

La figure 9.6 nous montre l’interface graphique de la page montrant la liste des débats déjà publiés dans la plateforme.

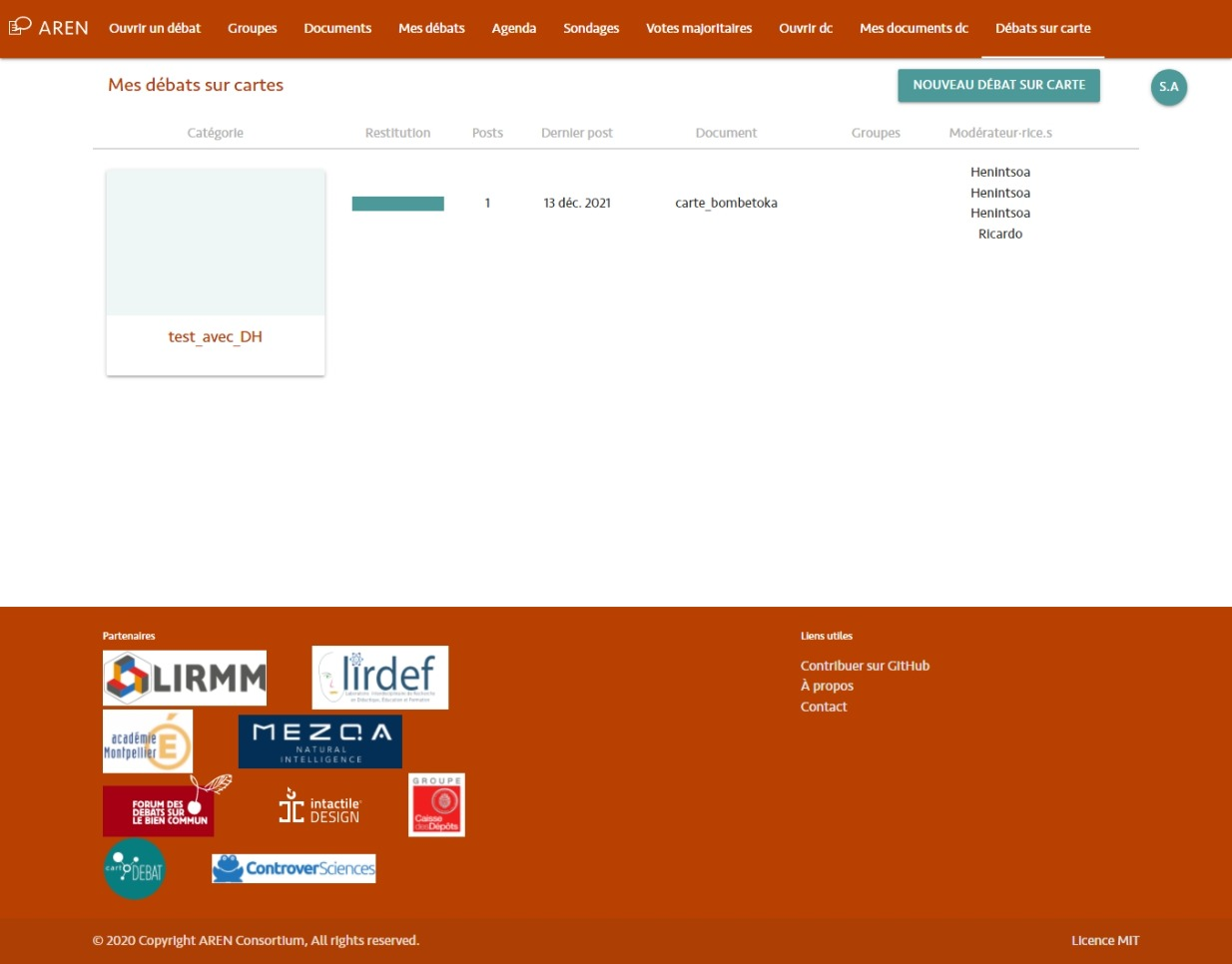


Figure 9.6 : Interface graphique de la page montrant la liste des débats déjà publiés dans la plateforme.

La figure ci-dessus nous montre toute la liste des débats présent dans la plateforme. En cliquant sur « aller au débat », on nous mène vers la figure 9.7.

### Affichage d’un débat

La figure 9.7 nous montre l’interface graphique de la page montrant les détails d’un débat avec une carte.

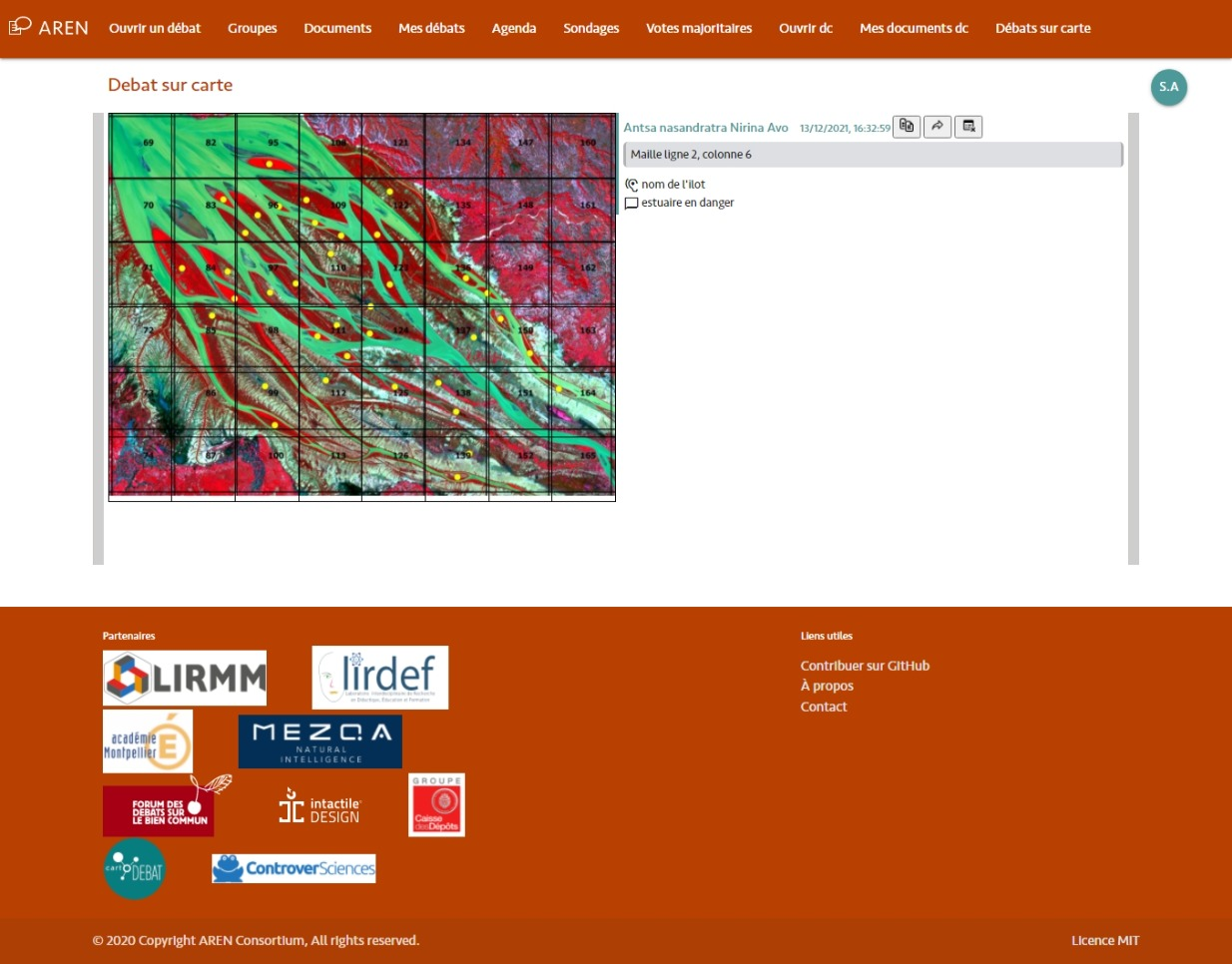


Figure 9.7 : Interface graphique de la page montrant les détails d’un débat avec une carte

Dans cette page, le sujet du débat est représenté par une carte. On peut effectuer l’argumentation en cliquant sur une maille de cette carte (sous forme de carreaux). Mais aussi, il est possible d’effectuer une argumentation à partir de celles des autres participants. La figure 9.8 nous montre le formulaire de commentaire ou d’argumentation. Par ailleurs, il y a des options de partage de lien vers une autre plateforme, en particulier Linfluent.

### Formulaire d’argumentation

La figure 9.8 nous montre l’interface graphique de la page montrant le formulaire d’argumentation.

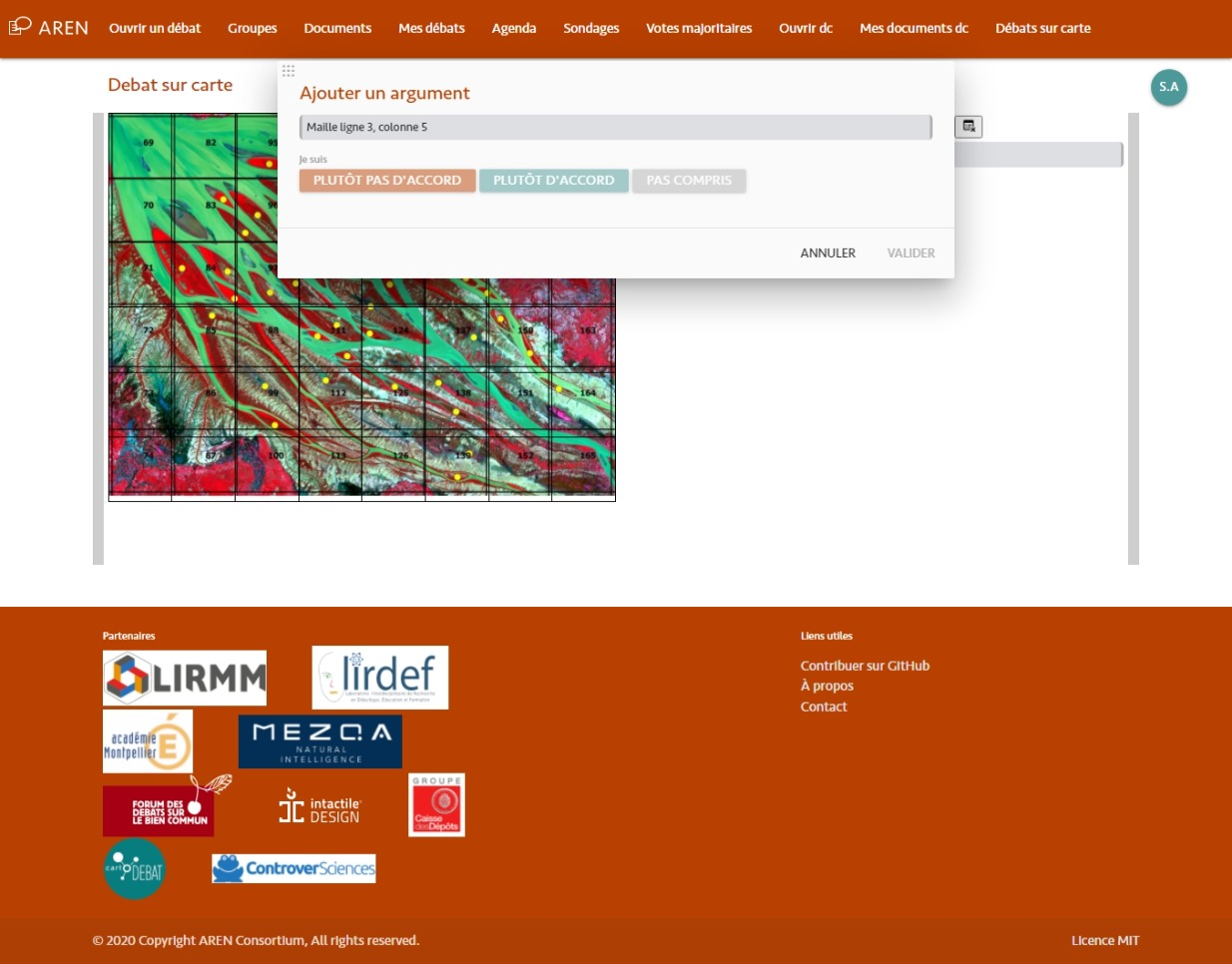


Figure 9.8 : Interface graphique de la page montrant le formulaire d’argumentation.

# Conclusion et perspective

En guise de conclusion, l’intégration au sein de la société DINIKA m’a donné l’occasion de faire face à la réalité professionnelle et l’approfondissement de nos connaissances théoriques et pratique sur la programmation informatique ainsi que la base des données.

D’après les analyses, la plateforme AREN favorise le développement des compétences argumentatives et les esprit critique des participants aux débats. De plus, la collecte des données statistiques des débats numériques est effectivement améliorée.

Comme la plateforme est encore actuellement en cours de développement, la société DINIKA vise encore d’aller plus loin sur la plateforme en effectuant une modélisation mathématique et développement d’une nouvelle intelligence artificiel dans AREN.

.

# Bibliographie

**[GAUDEL Marie-Claude et al., 1996] :** GAUDEL, Marie-Claude, MARRE, Bruno, SCHLIENGER, Françoise, *et al.* *Précis de génie logiciel*. Masson, 1996.

**[COLLIGNON, Alain et SCHÖPFEL, Joachim, 2016]** : COLLIGNON, Alain et SCHÖPFEL, Joachim. Méthodologie de gestion agile d’un projet. Scrum–les principes de base. I2D-Information, données documents, 2016, vol. 53, no 2, p. 12-15.

**[Permana, Putu A., 2015]** : PERMANA, Putu Adi Guna. Scrum method implementation in à software development project management. *International Journal of Advanced Computer Science and Applications*, 2015, vol. 6, no 9, p. 198-204.

**[Hasina R., Jérome Q, 2019]** : Hasina RAKOTONIRAINY, Jérôme QUEST, 2019, EMIT cours Genie logiciel, Un « bon » logiciel.

**[CÔTÉ, Daniel et ST-DENIS, Richard, 2006]** : CÔTÉ, Daniel et ST-DENIS, Richard. Compilateur du langage java--. Rapport technique, Université de Sherbrooke, 2006.

**[****HONDERMARCK, Olivier, 2009]** : HONDERMARCK, Olivier. JavaScript: le guide complet. MA éditions, 2009.

**[O'Neil, E.J, 2008] :** O'Neil, E. J. (2008, June). Object/relational mapping 2008: hibernate and the entity data model (edm). In Proceedings of the 2008 ACM SIGMOD international conference on Management of data (pp. 1351-1356).

**[MOMJIAN, Bruce, 2001]:** MOMJIAN, Bruce PostgreSQL: introduction and concepts. New York : Addison-Wesley, 2001.

**[TSAY, Jason, DABBISH, Laura, et HERBSLEB, 2014]**  : TSAY, Jason, DABBISH, Laura, et HERBSLEB, James. Influence of social and technical factors for evaluating contribution in GitHub. In : *Proceedings of the 36th international conference on Software engineering*. 2014. p. 356-366.

**[BALINSKI, Michel et LARAKI, Rida, 2011-2012]** : BALINSKI, Michel et LARAKI, Rida. Jugement majoritaire versus vote majoritaire : (via les présidentielles 2011-2012). Revue française d’économie, 2012, vol. 27, no 4, p. 11-44.

**[BADIR, Hassan**] : BADIR, Hassan. Les diagrammes de classes UML et la conception des bases de données.

**[Mooreeau C., 2016]** : MOREAU, Charlotte. " Analyse des interets et des enjeux d’adopter les méthodes agiles pour innover au sein d’une entreprise, cas pratique chez colruyt group. 2016.

**[Nevenka k, Saso k,2015]***:* KIROVSKA, Nevenka et KOCESKI, Saso. Usage of Kanban methodology at software development teams. *Journal of applied economics and business*, 2015, vol. 3, no 3, p. 25-34.

**[Michel Balinski, Rida Laraki, 2016]** : Michel Balinski et Rida Laraki. Le jugement majoritaire : une nouvelle théorie du vote, École Polytechnique et CNRS, Paris, 2016.

**[Michel Balinski, Rida Laraki, 2010]** : Michel Balinski et Rida Laraki. Majority Judgement : Measuring Ranking and Electing, MIT Press, 2010, et « Judge : Don’t Vote », Cahier du laboratoire d’Économétrie de l’École polytechnique, 27, 2010.

**[CAELEN, Jean., 2002]** : CAELEN, Jean. Modèles formels de dialogue. Actes des 2èmes assises du GdR I, 2002, vol. 3, p. 31-58.

**[Susanne Salmon-Ait, 2001] :** Susanne Salmon-Ait. Référence et dialogue finalisé : de la linguistique à un' modèle opérationnel, 2001

# Webographie

“EMIT | Accueil.” http://www.emit.mg/#close (accessed Nov. 13, 2021).

“EMIT | Personnel.” http://www.emit.mg/Personnel (accessed Nov. 13, 2021).

“Diagrammes de cas d'utilisation - Documentation IBM” https://www.ibm.com/docs/fr/rational-soft-arch/9.5?topic=diagrams-use-case (accessed Dec. 1, 2021).

“Diagrammes d'activité - Documentation IBM” https://www.ibm.com/docs/fr/rsar/9.5?topic=diagrams-activity (accessed Dec. 1, 2021).

“Présentation du projet AREN (lirmm.fr)” [Présentation du projet AREN (lirmm.fr)](https://www.lirmm.fr/aren/page1.php) (accessed Dec. 14, 2021).

différents modes de scrutins [Les différents modes de scrutins / Les modalités d'élections / Les élections en France / Archives élections / Archives - Ministère de l'Intérieur (interieur.gouv.fr)](https://www.interieur.gouv.fr/Archives/Archives-elections/Les-elections-en-France/Les-modalites-d-elections/Les-differents-modes-de-scrutins#:~:text=Le%20principe%20du%20scrutin%20majoritaire,suffrages%20recueillis%20par%20ses%20concurrents.)

“Le jugement majoritaire” <https://mieuxvoter.fr/le-jugement-majoritaire> (accessed Dec. 20, 2021).

# Table des matières

[Curriculum Vitae 0](#_Toc130758527)

[Avant-propos I](#_Toc130758528)

[Remerciements II](#_Toc130758529)

[Liste des figures III](#_Toc130758530)

[Liste des tableaux V](#_Toc130758531)

[Liste des abréviations VI](#_Toc130758532)

[Glossaire VIII](#_Toc130758533)

[Sommaire IX](#_Toc130758534)

[Introduction générale 1](#_Toc130758535)

[PARTIE 1 : PRESENTATION GENERALE 0](#_Toc130758536)

[Chapitre 1 : Présentation de l’Ecole Management et d’Innovation Technologique (EMIT) 3](#_Toc130758537)

[1.1. Historique [emit.mg] 3](#_Toc130758538)

[1.2. Missions 3](#_Toc130758539)

[1.3. Formations existantes 3](#_Toc130758540)

[1.4. Organigramme de l’EMIT 4](#_Toc130758541)

[Chapitre 2 : Présentation de Datascience INnovation Intelligence Knowledge Acquisition 6](#_Toc130758542)

[2.1. Création 6](#_Toc130758543)

[2.2. Siege social 6](#_Toc130758544)

[2.3. Mission et vision 6](#_Toc130758545)

[2.4. Les services offerts 6](#_Toc130758546)

[2.5. Les clients 7](#_Toc130758547)

[2.6. Les projets en cours 7](#_Toc130758548)

[Chapitre 3 : Présentation du projet 8](#_Toc130758549)

[3.1. Formulation du contexte et de l’origine du projet 8](#_Toc130758550)

[3.2. Problématique 8](#_Toc130758551)

[3.3. Objectifs et besoins de l’utilisateur 8](#_Toc130758552)

[3.4. Moyens personnels et techniques 9](#_Toc130758553)

[PARTIE 2 : ANALYSE ET CONCEPTION DU PROJET 0](#_Toc130758554)

[Chapitre 4 : Méthodes et notations utilisés 12](#_Toc130758555)

[4.1. Introduction 12](#_Toc130758556)

[4.2. La méthode de conduite de projet 12](#_Toc130758557)

[4.3. Méthodologie choisie 21](#_Toc130758558)

[Chapitre 5 : Analyse du projet 22](#_Toc130758559)

[5.1. Définition et contexte de l’AREN 22](#_Toc130758560)

[5.2. But de l’AREN 22](#_Toc130758561)

[5.3. Analyse des besoins 23](#_Toc130758562)

[5.4. Analyse de l’existant dans l’AREN 23](#_Toc130758563)

[5.5. Etat de l’art 23](#_Toc130758564)

[5.6. Proposition des solutions 25](#_Toc130758565)

[Chapitre 6 : Conception du projet 26](#_Toc130758566)

[6.1. Dictionnaire de données 26](#_Toc130758567)

[6.2. Les règles de gestions 28](#_Toc130758568)

[6.3. Diagramme UML du projet 28](#_Toc130758569)

[PARTIE 3 : REALISATION DU PROJET 0](#_Toc130758570)

[Chapitre 7 : Spécification des outils utilisés 37](#_Toc130758571)

[7.1. Introduction 37](#_Toc130758572)

[7.2. Langage de modélisation 37](#_Toc130758573)

[7.3. Langages de programmations 38](#_Toc130758574)

[7.4. Framework d’affichage 39](#_Toc130758575)

[7.5. Le Système de gestion de base de données. 40](#_Toc130758576)

[7.6. L’ORM utilisé 40](#_Toc130758577)

[Chapitre 8. Mise en œuvre et implémentation 42](#_Toc130758578)

[8.1. Architecture logicielle [Julie V. et Yann-Gaël G., 2008] 42](#_Toc130758579)

[8.2. Architecture matérielle de l’application [Julie V. et Yann-Gaël G., 2008] 43](#_Toc130758580)

[Chapitre 9 : Présentation du logiciel 44](#_Toc130758581)

[9.1. Introduction 44](#_Toc130758582)

[9.2. Création du vote majoritaire 44](#_Toc130758583)

[9.3. Effectuer un vote 46](#_Toc130758584)

[9.4. Création du document d’un débat sur carte 48](#_Toc130758585)

[9.5. Sélection des participants au débat sur carte 49](#_Toc130758586)

[9.6. Sélection du document pour débat sur carte 50](#_Toc130758587)

[9.7. Liste des débats sur carte 51](#_Toc130758588)

[9.8. Affichage d’un débat 52](#_Toc130758589)

[9.9. Formulaire d’argumentation 53](#_Toc130758590)

[Conclusion et perspective 55](#_Toc130758591)

[Bibliographie VIII](#_Toc130758592)

[Webographie X](#_Toc130758593)

[Table des matières XI](#_Toc130758594)

[Annexes XIV](#_Toc130758595)

[Annexe 1 : Code du model theme dans vote majoritaire XIV](#_Toc130758596)

[Annexe 2 : Code du model vote dans vote majoritaire XVII](#_Toc130758597)

[Annexe 3 : Code du service vote dans vote majoritaire XX](#_Toc130758598)

# Annexes

## Annexe 1 : Code du model theme dans vote majoritaire

package fr.lirmm.aren.model.vm;  
  
import com.fasterxml.jackson.annotation.JsonIdentityInfo;  
import com.fasterxml.jackson.annotation.ObjectIdGenerators;  
import fr.lirmm.aren.model.AbstractEntity;  
import fr.lirmm.aren.model.Team;  
import fr.lirmm.aren.model.User;  
import org.hibernate.annotations.Type;  
  
import javax.persistence.\*;  
import javax.validation.constraints.Size;  
import java.io.Serializable;  
import java.time.ZonedDateTime;  
import java.util.HashSet;  
import java.util.LinkedHashSet;  
import java.util.Set;  
  
*/\*\*  
 \** ***@author*** *ANDRIAMBOLAHARIMIHANTA Havana on 24/06/2021  
 \** ***@project*** *aren-1  
 \*/*@Entity  
@Table(name = "vm\_themes")  
@JsonIdentityInfo(generator = ObjectIdGenerators.PropertyGenerator.class, property = "id", scope = VMTheme.class)  
public class VMTheme extends AbstractEntity implements Serializable {  
  
 */\*\*  
 \*  
 \*/* @JoinColumn(name = "author", referencedColumnName = "id")  
 @ManyToOne  
 private User author;  
  
 @JoinColumn(name = "team", referencedColumnName = "id")  
 @ManyToOne  
 private Team team;  
  
 @Size(max = 255)  
 @Column(name = "title")  
 private String title;  
  
 @Lob  
 @Type(type = "org.hibernate.type.TextType")  
 @Column(name = "description")  
 private String description;  
  
 @Column(name = "createdAt")  
 private ZonedDateTime createdAt=ZonedDateTime.*now*();  
  
 @Column(name = "expiracyDate")  
 private ZonedDateTime expiracyDate;  
  
 @OneToMany(mappedBy = "themeId")  
 private Set<VMChoice> choices = new HashSet<>();  
  
 */\*\*  
 \*  
 \** ***@return*** *\*/* public User getAuthor() {  
 return author;  
 }  
  
 */\*\*  
 \*  
 \** ***@param*** *author  
 \*/* public void setAuthor(User author) {  
 this.author = author;  
 }  
  
 */\*\*  
 \*  
 \** ***@return*** *\*/* public Team getTeam() {  
 return team;  
 }  
  
 */\*\*  
 \*  
 \** ***@param*** *team  
 \*/* public void setTeam(Team team) {  
 this.team = team;  
 }  
  
 */\*\*  
 \*  
 \** ***@return*** *\*/* public String getTitle() {  
 return title;  
 }  
  
 */\*\*  
 \*  
 \** ***@param*** *title  
 \*/* public void setTitle(String title) {  
 this.title = title;  
 }  
  
 */\*\*  
 \*  
 \** ***@return*** *\*/* public String getDescription() {  
 return description;  
 }  
  
 */\*\*  
 \*  
 \** ***@param*** *description  
 \*/* public void setDescription(String description) {  
 this.description = description;  
 }  
  
 */\*\*  
 \*  
 \** ***@return*** *\*/* public ZonedDateTime getCreatedAt() {  
 return createdAt;  
 }  
  
 */\*\*  
 \*  
 \** ***@param*** *createdAt  
 \*/* public void setCreatedAt(ZonedDateTime createdAt) {  
 this.createdAt = createdAt;  
 }  
  
 */\*\*  
 \*  
 \** ***@return*** *\*/* public ZonedDateTime getExpiracyDate() {  
 return expiracyDate;  
 }  
  
 */\*\*  
 \*  
 \** ***@param*** *expiracyDate  
 \*/* public void setExpiracyDate(ZonedDateTime expiracyDate) {  
 this.expiracyDate = expiracyDate;  
 }  
  
 */\*\*  
 \*  
 \** ***@return*** *\*/* public Set<VMChoice> getChoices() {  
 return choices;  
 }  
  
 */\*\*  
 \*  
 \** ***@param*** *choices  
 \*/* public void setChoices(Set<VMChoice> choices) {  
 this.choices = choices;  
 }  
  
 @Override  
 public String toString(){  
 return "title : "+this.title+"\n"  
 +"Description : "+this.description+"\n"  
 +"Expiration : "+this.expiracyDate+"\n";  
 }  
}

## Annexe 2 : Code du model vote dans vote majoritaire

Voici le code écrit pour le model vote dans le vote majoritaire de la plateforme AREN

package fr.lirmm.aren.model.vm;  
  
import com.fasterxml.jackson.annotation.JsonIdentityInfo;  
import com.fasterxml.jackson.annotation.ObjectIdGenerators;  
import fr.lirmm.aren.model.AbstractEntity;  
import fr.lirmm.aren.model.User;  
  
import javax.persistence.\*;  
import java.io.Serializable;  
import java.time.ZonedDateTime;  
  
*/\*\*  
 \** ***@author*** *ANDRIAMBOLAHARIMIHANTA Havana on 25/06/2021  
 \** ***@project*** *aren-1  
 \*/*@Entity  
@Table(name = "vm\_votes")  
@JsonIdentityInfo(generator = ObjectIdGenerators.PropertyGenerator.class, property = "id", scope = VMVote.class)  
public class VMVote extends AbstractEntity implements Serializable {  
 public enum Opinion {  
 *REJECTED*,  
 *INSUFFICIENT*,  
 *PASS*,  
 *ACCEPTABLE*,  
 *GOOD*,  
 *VERY\_GOOD*,  
 *EXCELLENT* }  
  
 @ManyToOne  
 @JoinColumn(name = "subThemeId")  
 private VMChoice subThemeId;  
  
 @ManyToOne  
 @JoinColumn(name = "authorId")  
 private User authorId ;  
  
 @Enumerated(EnumType.*STRING*)  
 @Column(name = "opinion", nullable = false)  
 private Opinion opinion=Opinion.*REJECTED*;  
  
 @Column(name = "createdAt")  
 private ZonedDateTime createdAt=ZonedDateTime.*now*();  
  
 */\*\*  
 \*  
 \** ***@return*** *\*/* public VMChoice getSubThemeId() {  
 return subThemeId;  
 }  
  
 */\*\*  
 \*  
 \** ***@param*** *subThemeId  
 \*/* public void setSubThemeId(VMChoice subThemeId) {  
 this.subThemeId = subThemeId;  
 }  
  
 */\*\*  
 \*  
 \** ***@return*** *\*/* public User getAuthorId() {  
 return authorId;  
 }  
  
 */\*\*  
 \*  
 \** ***@param*** *authorId  
 \*/* public void setAuthorId(User authorId) {  
 this.authorId = authorId;  
 }  
  
 */\*\*  
 \*  
 \** ***@return*** *\*/* public Opinion getOpinion() {  
 return opinion;  
 }  
  
 */\*\*  
 \*  
 \** ***@param*** *opinion  
 \*/* public void setOpinion(Opinion opinion) {  
 this.opinion = opinion;  
 }  
  
 */\*\*  
 \*  
 \** ***@return*** *\*/* public ZonedDateTime getCreatedAt() {  
 return createdAt;  
 }  
  
 */\*\*  
 \*  
 \** ***@param*** *createdAt  
 \*/* public void setCreatedAt(ZonedDateTime createdAt) {  
 this.createdAt = createdAt;  
 }  
}

## Annexe 3 : Code du service vote dans vote majoritaire

package fr.lirmm.aren.service.vm;  
  
import fr.lirmm.aren.model.vm.VMChoice;  
import fr.lirmm.aren.model.vm.VMVote;  
import fr.lirmm.aren.service.AbstractService;  
  
import javax.enterprise.context.ApplicationScoped;  
import javax.ws.rs.NotFoundException;  
import java.util.List;  
  
*/\*\*  
 \** ***@author*** *ANDRIAMBOLAHARIMIHANTA Havana on 25/06/2021  
 \** ***@project*** *aren-1  
 \*/*@ApplicationScoped  
public class VMVoteService extends AbstractService<VMVote> {  
 public VMVoteService(){  
 super(VMVote.class) ;  
 }  
  
 @Override  
 protected void afterCreate(VMVote entity){  
 this.updateExternaleTables(entity);  
 }  
  
 public void updateExternaleTables(VMVote vmVote){  
 super.transactionBegin();  
  
 List<VMChoice> results = getEntityManager().createQuery("SELECT c "  
 + "FROM VMChoice c "  
 + "WHERE c.id = :id", VMChoice.class)  
 .setParameter("id", vmVote.getSubThemeId().getId())  
 .getResultList();  
  
 if (results.isEmpty()) {  
 throw new NotFoundException();  
 }  
  
 VMChoice choice=results.get(0) ;  
 switch(vmVote.getOpinion()){  
 case *REJECTED*:  
 choice.setRejected(choice.getRejected()+1);  
 break ;  
 case *INSUFFICIENT*:  
 choice.setInsufficient(choice.getInsufficient()+1);  
 break;  
 case *PASS*:  
 choice.setPass(choice.getPass()+1);  
 break;  
 case *ACCEPTABLE*:  
 choice.setAcceptable(choice.getAcceptable()+1);  
 break ;  
 case *GOOD*:  
 choice.setGood(choice.getGood()+1);  
 break ;  
 case *VERY\_GOOD*:  
 choice.setVeryGood(choice.getVeryGood()+1);  
 break ;  
 case *EXCELLENT*:  
 choice.setExcellent(choice.getExcellent()+1);  
 break ;  
 }  
  
 getEntityManager().createQuery("UPDATE VMChoice c SET "  
 + "c.rejected = :rejected, "  
 + "c.insufficient = :insufficient, "  
 + "c.pass = :pass, "  
 + "c.acceptable = :acceptable, "  
 + "c.good = :good, "  
 + "c.veryGood = :veryGood, "  
 + "c.excellent = :excellent "  
 +"WHERE c.id=:id"  
 )  
 .setParameter("rejected",choice.getRejected())  
 .setParameter("insufficient", choice.getInsufficient())  
 .setParameter("pass",choice.getPass())  
 .setParameter("acceptable",choice.getAcceptable())  
 .setParameter("good", choice.getGood())  
 .setParameter("veryGood",choice.getVeryGood())  
 .setParameter("excellent",choice.getExcellent())  
 .setParameter("id", vmVote.getSubThemeId().getId())  
 .executeUpdate() ;  
 super.commit();  
 }  
  
}

**RESUME**

Le stage que nous avons effectué au sein de la société DINIKA nous a conduit vers une expérience encore plus professionnelle. On a conçu et élaborer des nouvelles fonctionnalités dans la plateforme AREN.

Le principal but de ce projet est d’offrir des opportunités de développement des compétences argumentatives et les esprit critique des participants aux débats ainsi que de développer une vision plus riche de la nature des connaissances, de leurs modes de production et de validation

La plateforme AREN possède actuellement certaines fonctionnalités comme le pouvoir d’effectuer un vote majoritaire, du débat sur carte qui puisse encore être amélioré. Comme la plateforme AREN est déjà en cours de développement avant de notre intégration, on n’a qu’à continuer à utiliser le langage java comme langage de programmation.

**Mots clés :** AREN, débat, vote, argumenter, connaissances

**ABSTRACT**

The internship that we carried out within the company DINIKA led us to an even more professional experience. We have designed and developed new functionalities in the AREN platform.

The main goal of this project is to offer opportunities for the development of argumentative skills and critical thinking of participants in debates as well as to develop a richer vision of the nature of knowledge, its modes of production and validation.

The AREN platform currently has some features like the power to vote majority vote, card debate which could be further improved. As the AREN platform is already under development before our integration, we just have to continue to use the java language as a programming language.

**Keywords:** AREN, debate, vote, argue, knowledge

1. <https://www.lirmm.fr/aren/#prest> consulté de 07 Novembre 2020 [↑](#footnote-ref-1)