|  |  |
| --- | --- |
|  |  |

**UNIVERSITE DE FIANARANTSOA**

**ECOLE NATIONALE D’INFORMATIQUE**

**MEMOIRE DE FIN D’ETUDES POUR L’OBTENTION DU**

**DIPLOME DE MASTER PROFESSIONNEL**

**Mention : Informatique**

**Parcours : Informatique générale**

**Intitulé**

**CONCEPTION ET REALISATION D’UNE APPLICATION WEB POUR LA GESTION DE LA RELATION HUMAINE PAR LA COMMUNICATION NON VERBALE**

Présenté le 8 Septembre 2021

Par ZAFINDRATAFA ONJANIAINA Stephanot Karoza

**Membres du Jury :**

**Président :** Monsieur MAHATODY Thomas, Maître de Conférences

**Examinateur :** Monsieur RAMAMONJISOA Bertin Olivier, Professeur Titulaire

**Rapporteurs :** Monsieur RABARIVONY Zo, lead développeur

Monsieur GILANTE Gesazafy, Assistants d’Enseignement Supérieur et de Recherche

**Année universitaire 2019 - 2020**

CURRICULUM VITAE

**ETAT CIVIL**

Nom et Prénoms : ZAFINDRATAFA ONJANIAINA Stephanot Karoza

Date et Lieu de naissance : 09-05-1997 à Ambatomainty Ikalamavony

Situation matrimoniale : Célibataire

Adresse : Porte N°02 bâtiment 14 Caserne Tsaramandroso

Tél : +261 34 99 263 66

Email : [stephanotk@gmail.com](mailto:stephanotk@gmail.com)

**FORMATIONS ET DIPLOMES**

* 2019-2020 :
* Etudiant en deuxième année de formation en Master professionnel à l’ENI de l’université de Fianarantsoa**.**
  + Parcours :Informatique générale
* Obtention de diplôme d’étude en langue française niveau B2
* 2018-2019 : Etudiant en première année de formation en Master professionnel à l’ENI de l’université de Fianarantsoa**.**
  + - Parcours :Informatique générale
* 2015-2018 : Obtention de diplôme de Licence professionnelle en Informatique à l’ENI de l’université de Fianarantsoa.
  + - Parcours :Informatique générale
  + 2014-2015 : Obtention » de Baccalauréat série D au Lycée Raherivelo Ramamonjy

**STAGES ET EXPERIENCES PROFESSIONNELLES**

* 2019-2020 :
  + Stage de fin d’étude au sein de l’entreprise MORALOGIQ Manakambahiny Antananarivo, thème : « Conception et réalisation d’une application web pour la gestion de la relation humaine par la communication non verbale ».
  + Stage de 3 mois au sein de l’entreprise MyAgency.
* 2018-2019 :
  + Projet fin d’année en Master I, thème : « Conception et réalisation d’une application web/mobile QR code de gestion de présence à l’ENI ».
* 2017-2018 :
  + Stage au sein de l’entreprise RDJ Ingeno consulting Antananarivo d’une durée de 3 mois avec pour thème : « Conception et réalisation d’une application pour la gestion de service à la personne ».
  + Elaboration d’un projet au sein de l’ENI pour la gestion de réservation de train développé sous JAVA et ASP.NET.
  + Elaboration d’un projet au sein de l’ENI développement web mobile avec React native
  + Mise en place d’un serveur VOIP Asterisk.
  + Mise en place d’un détecteur d’intrusion (IPS) snort.
  + Mise en place d’un serveur pour la haute disponibilité : Load balancing et fail over avec Windows server 2012.
* 2016-2017 :
  + Stage au sein du ministère du tourisme Tsimbazaza Antananarivo thème : Conception et réalisation d’une application pour la gestion d’annuaire des établissements touristiques de Madagascar.
  + Elaboration de projet au sein de l’ENI pour la gestion de prestation des employés en PHP et en VB.NET.
* 2015-2016 :
  + Initiation à l’utilisation des logiciels de gestion d’entreprise à l’ENI avec Qt Creator et développée en C.

**CONNAISSANCES EN INFORMATIQUE**

* Gestion de projet : AGILE, SCRUM
* Langage de modélisation : UML
* Méthodes de conception : 2TUP, MERISE
* Langages de programmation : C, C++, Java, VB.NET, Python, Dart
* SGBD : MySQL, Oracle, MongoDB
* Systèmes d’exploitation : Linux et Windows
* Technologies web: JavaScript, PHP, XML, ASP, JSP, Angular JS, NodeJS.
* Frameworks: Symfony, Laravel, Codeigniter, React, Flutter
* Réseaux et protocoles : TCP/IP, LAN, WAN, MAN, internet, routage, câblage, configuration sécurité
* Outils virtuels : GNS3, Virtual Box

**CONNAISSANCES LINGUISTIQUES**

|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| Domaines | Comprendre à l'audition | | | | | Lire | | | | | Ecrire et rédiger | | | | | Parler et communiquer oralement | | | | |
| Langues |
| Niveau | TB | B | AB | P | NS | TB | B | AB | P | NS | TB | B | AB | P | NS | TB | B | AB | P | NS |
| Français |  | X |  |  |  |  | X |  |  |  |  | X |  |  |  |  | X |  |  |  |
| Anglais |  |  | X |  |  |  |  | X |  |  |  |  | X |  |  |  |  | X |  |  |

TB : Très bien, B : Bien, AB : Assez-Bien, P : Passable, NS : Niveau scolaire

**DIVERS**

Sport : basket ball

SOMMAIRE GENERAL

[CURRICULUM VITAE I](#_Toc81804687)

[SOMMAIRE GENERAL IV](#_Toc81804688)

[REMERCIEMENTS VII](#_Toc81804689)

[LISTE DES FIGURES VIII](#_Toc81804690)

[LISTE DES TABLEAUX X](#_Toc81804691)

[LISTE DES ABREVIATIONS XI](#_Toc81804692)

[INTRODUCTION GENERALE 1](#_Toc81804693)

[PARTIE I : PRESENTATIONS 2](#_Toc81804694)

[Chapitre 1 Présentation de l’Ecole Nationale d’Informatique 3](#_Toc81804695)

[1.1. Information d’ordre générale 3](#_Toc81804696)

[1.2. Missions et historiques 3](#_Toc81804697)

[1.3. Organigramme institutionnel de l’ENI 5](#_Toc81804698)

[1.4. Domaines et spécialisation 6](#_Toc81804699)

[1.5. Architecture des formations pédagogiques 7](#_Toc81804700)

[1.6. Relation de l’ENI avec les entreprises et organismes 9](#_Toc81804701)

[1.7. Partenariat au niveau international 10](#_Toc81804702)

[1.8. Débouchés professionnels des diplômes 12](#_Toc81804703)

[1.9. Ressources humaines 14](#_Toc81804704)

[Chapitre 2 Présentation de MORALOGIQ 15](#_Toc81804705)

[2.1. Historique et missions 15](#_Toc81804706)

[2.2. Situation juridique 15](#_Toc81804707)

[2.3. Organigramme 15](#_Toc81804708)

[2.4. Services 15](#_Toc81804709)

[2.5. Contacts 16](#_Toc81804710)

[Chapitre 3 Description du projet 17](#_Toc81804711)

[3.1. Formulation 17](#_Toc81804712)

[3.2. Objectif et besoins d’utilisateur 17](#_Toc81804713)

[3.3. Moyen nécessaire à la réalisation du projet 17](#_Toc81804714)

[3.4. Résultats attendus 20](#_Toc81804715)

[3.5. Chronogramme des Travaux 21](#_Toc81804716)

[PARTIE II : ANALYSE ET CONCEPTION 22](#_Toc81804717)

[Chapitre 4 Analyse préalable 23](#_Toc81804718)

[4.1. Analyse de l’existant 23](#_Toc81804719)

[4.2. Critique de l’existant 23](#_Toc81804720)

[4.3. Conception avant-projet 24](#_Toc81804721)

[Chapitre 5 Analyse conceptuelle 45](#_Toc81804722)

[5.1. Étapes de l’élaboration du backlog de produit 45](#_Toc81804723)

[5.2. Dictionnaire des données 59](#_Toc81804724)

[5.3. Règle de gestion 64](#_Toc81804725)

[5.4. Capture des besoins fonctionnels 65](#_Toc81804726)

[5.5. Analyse 65](#_Toc81804727)

[5.6. Capture des besoins techniques 72](#_Toc81804728)

[5.7. Conception préliminaire 72](#_Toc81804729)

[Chapitre 6 Conception détaillée 85](#_Toc81804730)

[6.1. Architecture du système 85](#_Toc81804731)

[6.2. Diagramme de séquence de conception 87](#_Toc81804732)

[6.3. Diagramme de classe de conception 88](#_Toc81804733)

[6.4. Diagramme de classe de conception global 95](#_Toc81804734)

[6.5. Diagramme de paquetage 95](#_Toc81804735)

[6.6. Diagramme de déploiements 96](#_Toc81804736)

[PARTIE III : REALISATION 97](#_Toc81804737)

[Chapitre 7 Mise en place de l’environnement de développement 98](#_Toc81804738)

[7.1. Installation et configuration des outils 98](#_Toc81804739)

[7.2. Architecture de l’application 100](#_Toc81804740)

[Chapitre 8 Développement de l’application 101](#_Toc81804741)

[8.1. Création de la base de données 101](#_Toc81804742)

[8.2. Codage de d’application 106](#_Toc81804743)

[8.3. Présentation de l’application 117](#_Toc81804744)

[CONCLUSION XI](#_Toc81804745)

[BIBLIOGRAPHIE XII](#_Toc81804746)

[WEBOGRAPHIE XIII](#_Toc81804747)

[GLOSSAIRE XIV](#_Toc81804748)

[ANNEXES XV](#_Toc81804749)

[TABLE DES MATIERES XVI](#_Toc81804750)

[RESUME XX](#_Toc81804751)

[ABSTRACT XX](#_Toc81804752)

REMERCIEMENTS

Avant tout, je tiens à remercier Dieu tout puissant de m’avoir donné la force, le courage et surtout la santé, afin de mener à bien tout ce travail. J’adresse également mes vifs remerciements à :

* Monsieur **RAFAMANTANANTSOA Fontaine**, Professeur HDR à l’Université de Fianarantsoa et Président de l’Université de Fianarantsoa de m’avoir accueilli au sein de l’Université de Fianarantsoa ;
* Monsieur **RAMAMONJISOA Andriantiana Bertin Olivier**, Professeur Titulaire à l’Université de Fianarantsoa et Directeur de l’Ecole Nationale d’Informatique de Fianarantsoa, pour avoir accepté mon inscription au sein de son établissement et d’avoir bien voulu accepter d’examiner ce travail ;
* Monsieur **RAKOTONDRANDRIA Elie**, fondateur de la société Moralogiq, de m’avoir donné l’opportunité d’intégrer son entreprise.
* Monsieur **GILANTE Gesazafy,** Assistants d’Enseignement Supérieur et de Recherche, mon encadreur pédagogique, pour son organisation et son soutien au bon déroulement ;
* Monsieur **RABARIVONY Zo Harisoa,** Lead développeur chez Moralogiq, mon encadreur professionnel, pour son soutien et ses conseils ainsi que l’aide précieux qu’il a donné qui a abouti à l’accomplissement de ce mémoire ;
* Tous les membres de jury ici présent d’avoir pu consacrer leur temps pour l’examen de ce mémoire malgré leur lourde responsabilité ;
* Tout le personnel de l’entreprise Moralogiq*;*
* Tous les professeurs et enseignants de l’Ecole Nationale de l’Informatique pour leurs cours et les connaissances illimitées qu’ils nous ont transmises.

Je tiens pareillement à exprimer mes remerciements à toute ma famille plus particulièrement mes parents et mes proches qui m’ont soutenu moralement ainsi que financièrement durant le déroulement du stage.

LISTE DES FIGURES

[Figure 1. organigramme de l'ENI 6](#_Toc81776975)

[Figure 2. Organigramme du société MORALOGIQ 15](#_Toc81776976)

[Figure 3 Chronogramme des travaux 21](#_Toc81776977)

[Figure 4 Tableau de bord de Trello 39](#_Toc81776978)

[Figure 5 Espace de travail de JIRA 40](#_Toc81776979)

[Figure 6 Manifeste agile 42](#_Toc81776980)

[Figure 7 Étapes de SCRUM 42](#_Toc81776981)

[Figure 8 Diagrammes de cas d'utilisation 67](#_Toc81776982)

[Figure 9 PU centré sur l'architecture 73](#_Toc81776983)

[Figure 10 PU itératif et incrémental 74](#_Toc81776984)

[Figure 11 Systèmes d'information soumis à deux natures de contraintes 75](#_Toc81776985)

[Figure 12 processus de développement en Y 76](#_Toc81776986)

[Figure 13 Séquence de système pour le cas d'utilisation s'authentifier 78](#_Toc81776987)

[Figure 14 Diagramme de séquence système du cas d'utilisation gérer utilisateur 78](#_Toc81776988)

[Figure 15 Diagramme de séquence du cas d'utilisation gérer codification 79](#_Toc81776989)

[Figure 16 Diagramme de séquence du cas d'utilisation s'inscrire 79](#_Toc81776990)

[Figure 17 Diagramme de séquence de cas d'utilisation d'items secondaires 80](#_Toc81776991)

[Figure 18 Diagramme de séquence de cas d'utilisation approuver une analyse 81](#_Toc81776992)

[Figure 19 Diagramme de séquence du cas d'utilisation uploader vidéo 81](#_Toc81776993)

[Figure 20 Diagramme de séquence du cas d'utilisation gérer lecture d'une vidéo 82](#_Toc81776994)

[Figure 21 Diagramme de séquence du cas d'utilisation consulter galerie de vidéo 83](#_Toc81776995)

[Figure 22 Diagrammes de séquence du cas d'utilisation gérer analyses 83](#_Toc81776996)

[Figure 23 Modèle de domaine 84](#_Toc81776997)

[Figure 24 Architecture MVC 86](#_Toc81776998)

[Figure 25 Diagramme de séquence de conception du cas d’utilisation s’authentifier 87](#_Toc81776999)

[Figure 26 Diagramme de séquence de conception du cas d'utilisation uploader vidéo 88](#_Toc81777000)

[Figure 27 Diagramme de classe de conception du cas d'utilisation s'authentifier 89](#_Toc81777001)

[Figure 28 Diagramme de classe de conception de gérer utilisateurs 90](#_Toc81777002)

[Figure 29 Diagramme de classe de conception de gérer codification 90](#_Toc81777003)

[Figure 30 Diagramme de classe de conception de gérer items secondaires 91](#_Toc81777004)

[Figure 31 Diagramme de classe de conception de upload vidéo 91](#_Toc81777005)

[Figure 32 Diagramme de classe de conception du cas d’utilisation approuver analyse 92](#_Toc81777006)

[Figure 33 Diagramme de classe de conception de gérer lecture d'une vidéo 93](#_Toc81777007)

[Figure 34 Diagramme de classe de conception de consulter galerie 93](#_Toc81777008)

[Figure 35 Diagramme de classe de conception de gérer analyse 94](#_Toc81777009)

[Figure 36 Diagramme de classe de conception globale 95](#_Toc81777010)

[Figure 37 Diagrammes de paquetage 95](#_Toc81777011)

[Figure 38 Diagramme de déploiement 96](#_Toc81777012)

[Figure 39 Installation de visual paradigm 98](#_Toc81777013)

[Figure 40 Interface de phpStorm 100](#_Toc81777014)

[Figure 41 Architecture 3-tiers 100](#_Toc81777015)

[Figure 42 Page d'authentification 117](#_Toc81777016)

[Figure 43 Page de gestion d'utilisateur 117](#_Toc81777017)

[Figure 44 page de galerie photo 118](#_Toc81777018)

[Figure 45 page de gestion de codification 119](#_Toc81777019)

[Figure 46 page d'item secondaire 119](#_Toc81777020)

LISTE DES TABLEAUX

[Tableau 1. Organisation du système de formation pédagogique de l'école 7](#_Toc81777039)

[Tableau 2 Architecture des études correspondant au système LMD 8](#_Toc81777040)

[Tableau 3 Listes des formations existantes à l'ENI 8](#_Toc81777041)

[Tableau 4 Débouchés professionnels éventuels des diplômes 13](#_Toc81777042)

[Tableau 5 Moyens humain 18](#_Toc81777043)

[Tableau 6. Inventaire des moyens matériels 19](#_Toc81777044)

[Tableau 7 Inventaire des moyens logiciels 20](#_Toc81777045)

[Tableau 8 Inventaire des matériels de l'existant 23](#_Toc81777046)

[Tableau 9 Comparaison des solutions proposées 24](#_Toc81777047)

[Tableau 10 Comparaison entre MERISE et Processus unifié 25](#_Toc81777048)

[Tableau 11 Comparaison des outils de conception 27](#_Toc81777049)

[Tableau 12 Tableau comparatif entre MySQL et PostgreSQL 28](#_Toc81777050)

[Tableau 13 Comparatif des IDE 32](#_Toc81777051)

[Tableau 14 Comparaison entre Git et SVN 33](#_Toc81777052)

[Tableau 15 Comparatif des langages de programmation 34](#_Toc81777053)

[Tableau 16 Comparaison entre les frameworks PHP 37](#_Toc81777054)

[Tableau 17 Backlog du produit 58](#_Toc81777055)

[Tableau 18 Priorisation des cas d'utilisation 68](#_Toc81777056)

LISTE DES ABREVIATIONS

**2TUP**  : Two Track Unified Process

**AUF**  : Agence Universitaire de la Francophonie

**B2B**  : Business to Business

**BPMN**  : Business Process Model and Notation

**CARI**  : Colloque Africain sur la Recherche en Informatique

**CCNA**  : Cisco Certified Network Associate Routing and Switching

**CHU**  : Centre Hospitalier Universitaire

**CNH**  : Commission Nationale d’Habilitation

**CNRE**  : Centre National de Recherche sur l’Environnement

**CNRIT**  : Centre National de Recherches Industrielle et Technologique

**CPU**  : Central Processing Unit

**CSS** : Cascading Style Sheets

**CUR**  : Centre Universitaire Régional

**DD** : Disque Dur

**ESPA** : Ecole Supérieur Polytechnique d’Antananarivo

**FID**  : Fonds d’Intervention pour le développement

**FTM**  : Foibe Tao-Tsaritanin’i Madagasikara

**HDD**  : Hard Disk Drive

**HP**  : Hewlett Packard

**HTML**  : Hypertext Markup Language

**IES** : Institut Européen de Synergologie

**INPG** : Institut National Polytechnique de Grenoble

**INSA**  : Institut National des Sciences Appliquées de Lyon

**IPS**  : Intrusion Prevention System

**IRD**  : Institut de Recherche pour le développement

**JS**  : JavaScript

**JSP**  : Java Server Page

**LMD**  : Licence Master Doctorat

**MAEF**  : Mathématiques Appliquées à l’Economie et à la Finance

**MAN**  : Metropolitan Area Network

**MEF**  : Ministère de l’Economie et des Finances

**MEN**  : Ministère de l’Education Nationale

**MeSupRES**  : Ministère de l’Enseignement supérieur et de recherche scientifique

**MFB** : Ministère des Finances et du Budget

**MIC**  : Madagascar Informatique Corporation

**MININTER**  : Ministère de l’Interieur

**MVC**  : Modèle Vue Contrôleur

**OS**  : Operating System

**PHP** : Hypertext Preprocessor

**PRESUP**  : Programme de renforcement de l’Enseignement Supérieur

**PU**  : Processus Unifié

**RAM**  : Random Access Memory

**SARL** : Société à Responsabilité Limitée

**SGBD**  : Système de Gestion de Base de Donnée

**SSII**  : Société de Services en Ingénierie Informatique

**TCP/IP**  : Transmission Control Protocol / Internet Protocol

**TVA**  : Taxe sur valeur ajoutée

**UML**  : Unified Modeling Language

**UP**  : Unified Process

**UPST** : Université Paul Sabatier de Toulouse

**VB**  : Visual Basic

**VOIP**  : Voice Over IP

**VP**  : Visual Paradigm

**WAN** : Wide Area Network

INTRODUCTION GENERALE

A delà de la parole, l'expression du visage et la gestuelle représente un code de communication fiable pour les hommes. L'étude de ce code a été inventée et se nomme la synergologie.

La synergologie est une marque de commerce relative à l’étude de la communication non verbale, des messages transmis par les mouvements corporels inconscients. Le corps émet des messages inconsciemment dans une communication inter personnel et celui-ci est resté inaperçu si la personne ne sait pas comment la décoder. Comment est-ce qu'on va interpréter ces gestes et mettre à disposition de tous cette capacité d’analyser les micros geste inconscient?

C'est à partir de cette question que s'est traduite l'idée de ce projet synergologie, qui consiste à concevoir et à réaliser une application web pour la gestion de la relation humaine par communication non verbale.

La réussite de ce projet comme n'importe quel autre dépend de la méthode adoptée.

En général, il faut commencer par étudier le système existant sur lequel travailler et chercher ses avantages et ses éventuels inconvénients, ensuite explorer diverses solutions afin de retenir la meilleure et celle adaptée au système, et enfin mettre en place des améliorations ou un nouveau système après avoir procédé par des tests justifiant son bon fonctionnement.

Pour mener à bien ce projet, une méthode sera adoptée. On va également devoir choisir un langage de programmation, un système de gestion de base de données, un système de gestion de version et un outil de gestion de projet.

Nous allons donc développer ces principes en commençant par les présentations générales suivies de la partie analyse et conception, et enfin la réalisation.

# PRESENTATIONS

## Présentation de l’Ecole Nationale d’Informatique

### Information d’ordre générale

L’Ecole Nationale d’Informatique, en abrégé ENI, est un établissement d’enseignement supérieur rattaché académiquement et administrativement à l’Université de Fianarantsoa. Le siège de l’Ecole se trouve à Tanambao- Antaninarenina à Fianarantsoa.

Les coordonnées pour la prise de contact avec l’Ecole sont les suivantes :

Ecole Nationale d’Informatique (ENI)

Antaninarenina Tanambao

BP 1487 FIANARANTSOA (301)

Téléphones : 034 05 733 36 ou 033 42 302 02

Adresse mél : eni@eni-univ-fianar.mg. Site : [www.eni-univ-fianar.mg](http://www.eni-univ-fianar.mg)

### Missions et historiques

L’ENI se positionne sur l’échiquier socio-éducatif malgache comme étant le plus puissant secteur de diffusion et de vulgarisation des connaissances et des technologies informatiques.

Cette Ecole Supérieure peut être considérée aujourd’hui comme la vitrine et la pépinière des élites informaticiennes du pays.

L’Ecole s’est constituée de façon progressive au sein du Centre Universitaire Régional (CUR) de Fianarantsoa.

De façon formelle, l’ENI était constituée et créée au sein du (CUR) par le décret N° 83-185 du 24 Mai 1983, comme étant le seul établissement Universitaire Professionnalisé au niveau national, destiné à former des techniciens et des Ingénieurs de haut niveau, aptes à répondre aux besoins et exigences d’Informatisation des entreprises, des sociétés et des organes implantés à Madagascar.

L’ENI a pour conséquent pour mission de former des spécialistes informaticiens compétents et opérationnels de différents niveaux notamment :

En fournissant à des étudiants des connaissances de base en informatique ;

En leur transmettant le savoir-faire requis, à travers la professionnalisation des formations dispensées et en essayant une meilleure adéquation des formations par rapport aux besoins évolutifs des sociétés et des entreprises.

En initiant les étudiants aux activités de recherche dans les différents domaines des Technologies de l’information et de la communication (TIC).

L’implantation de cette Ecole Supérieure de technologie de pointe dans un pays en développement et dans une Province (ou Faritany) à tissu économique et industriel faiblement développé ne l’a pourtant pas défavorisée, ni empêchée de former des spécialistes informaticiens de bon niveau, qui sont recherchés par les entreprises, les sociétés et les organismes publics et privés sur le marché de l’emploi.

La filière de formation d’Analystes Programmeurs a été mise en place à l’Ecole en 1983, et a été gelée par la suite en 1996, tandis que la filière de formation d’ingénieurs a été ouverte à l’Ecole en 1986.

Dans le cadre du Programme de renforcement en l’Enseignement Supérieur (PRESUP) la filière de formation des Techniciens Supérieurs en Maintenance des Systèmes des informatiques a été mise en place en 1986 grâce à l’appui matériel et financier de la Mission Française de coopération auprès de l’Ambassade de France à Madagascar.

Une formation pour l’obtention de la certification CCNA et / ou NETWORK + appelée « CISCO Networking Academy » a été créée à l’Ecole en 2002-2003 grâce au partenariat avec CISCO SYSTEM et l’Ecole Supérieure Polytechnique d’Antananarivo (ESPA). Cependant, cette formation n’avait pas duré longtemps.

L’organisation de stage en entreprise continue non seulement à renforcer la professionnalisation des formations dispensées, mais elle continue surtout à accroître de façon exceptionnelle les opportunités d’embauche pour les diplômés de l’Ecole:

* Génie Logiciel et base de Données.
* Administration des Système et réseaux.

La mise en place à l’Ecole de ces deux options de formation devait répondre au besoin de basculement vers le système Licence – Master – Doctorat (LMD).

Mais la filière de formation des Techniciens Supérieurs en Maintenance des Systèmes Informatiques a été gelée en 2009.

En vue de surmonter les difficultés de limitation de l’effectif des étudiants accueillis à l’Ecole, notamment à cause du manque d’infrastructures, un système de « Formation Hybride » a été mise en place à partir de l’année 2010. Il s’agit en effet d’un système de formation semi-présentielle et à distance avec l’utilisation de la visioconférence pour la formation à distance.

Le système de formation hybride a été ainsi créé à Fianarantsoa ainsi qu’Université de Toliara.

### Organigramme institutionnel de l’ENI

Cet organigramme de l’Ecole est inspiré des dispositions du décret N° 83-185 du 24 Mai 1983.

Sur cet organigramme, l’Ecole placée sous la tutelle académique et administrative de l’Université de Fianarantsoa, est dirigée par un Directeur élu par les Enseignants – Chercheurs permanents de l’Etablissement et nommé par un décret pris en Conseil des Ministres pour un mandat de 3 ans.

Le Conseil de l’Ecole est l’organe délibérant de l’Ecole.

Le Collège des Enseignants regroupant tous les enseignants-chercheurs de l’Ecole est chargé de résoudre les problèmes liés à l’organisation pédagogique des enseignements. Il propose et coordonne les programmes d’activités pédagogiques.

Le Conseil Scientifique propose les orientations pédagogiques et scientifiques de l’établissement, en tenant compte notamment de l’évolution du marché de travail et de l’adéquation des formations dispensées par rapport aux besoins des entreprises. Il coordonne les programmes de recherche à mettre en œuvre à l’Ecole.

Le Secrétariat principal coordonne les activités des services administratifs (Scolarité, Comptabilité, et Intendance).

Conformément aux textes en vigueur régissant les Etablissements malgaches d’Enseignement Supérieur, qui sont barrés sur le système LMD, les Départements de Formation pédagogique ont été ainsi remplacés par des Mentions et des parcours. Et les chefs des Départements ont été ainsi remplacés par des responsables des mentions et les responsables des parcours.

L’organigramme de l’Ecole Nationale d’Informatique est présenté par la **Erreur ! Source du renvoi introuvable.**.

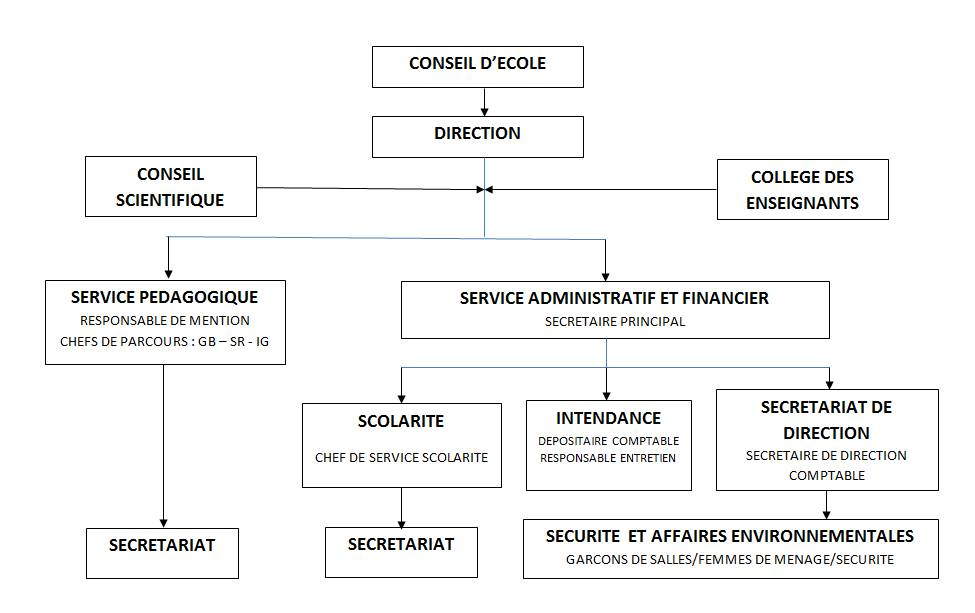


Figure 1. Organigramme de l'ENI

### Domaines et spécialisation

Les activités de formation et de recherche organisées à l’ENI portent sur les domaines suivants :

* Génie logiciel et Base de Données ;
* Administration des Systèmes et Réseaux ;
* Informatique Générale ;
* Modélisation informatique et mathématique des Systèmes complexes.

D’une manière plus générale, les programmes des formations sont basés sur l’informatique de gestion et sur l’informatique des Systèmes et Réseaux. Et les modules de formation intègrent aussi bien des éléments d’Informatique fondamentale que des éléments d’Informatique appliquée.

Le **Erreur ! Source du renvoi introuvable.** décrit l’organisation du système de formation pédagogique de l’Ecole.

Tableau 1. Organisation du système de formation pédagogique de l'école

|  |  |
| --- | --- |
| Formation Théorique | Formation pratique |
| * Enseignement théorique * Travaux dirigés * Travaux pratiques * Conférences | * Etude de cas * Travaux de réalisation * Projets/ Projets tutoriels * Voyages d’Etudes * Stages en entreprise |

### Architecture des formations pédagogiques

Le recrutement des étudiants à l’ENI se fait uniquement par voie de concours d’envergure nationale en première année de Licence Professionnelle.

Les offres de formation organisées à l’Ecole ont été validées par la Commission Nationale d’Habilitation (CNH) auprès du Ministères de l’Enseignement Supérieur et de la Recherche Scientifique selon les dispositions de l’Arrêté N° 31.174/2012-MENS en date du 05 Décembre 2012.

Au sein de l’ENI, il existe une seule mention (INFORMATIQUE) et trois parcours :

* Génie logiciel et Base de Données,
* Administration des Systèmes et Réseaux,
* Informatique Générale.

L’architecture des études à trois niveaux conforment au système Licence-Master-Doctorat (LMD) permet les comparaisons et les équivalences académiques des diplômes au niveau international.

* L =Licence (Bac + 3) = L1, L2, L3 = 6 semestres S1 à S6
* M = Master (Bac + 5) = M1, M2 = 4 semesters S7 à S10

Le diplôme de licence est obtenu en 3 années des études après Baccalauréat. Et le diplôme de Master est obtenu en 2 ans après obtention du diplôme de LICENCE.

Le MASTER PROFESSIONNEL est un diplôme destiné à la recherche d’emploi au terme des études.

Le MASTER RECHERCHE est un diplôme qui remplace l’ancien Diplôme d’Etudes Approfondies (DEA), et qui permet de s’inscrire directement dans une Ecole Doctorale.au terme des études.

* D = Doctorat (Bac +8)

Le Doctorat est un diplôme qu’on peut obtenir en 3 ans après l’obtention du diplôme de MASTER RECHERCHE.

Le Tableau 2 présente l’architecture des études correspondant au système LMD.

Tableau 2 Architecture des études correspondant au système LMD



La licence peut avoir une vocation générale ou professionnelle.

Le master peut avoir une vocation professionnelle ou de recherche.

Tableau 3 Listes des formations existantes à l'ENI

|  |  |  |
| --- | --- | --- |
|  | FORMATION | |
|  | LICENCE PROFESSIONNELLE | MASTER |
| Condition d’admission | Par voie de concours  Formation Professionnelle : 100 candidats  Formation hybride : 150 candidats |  |
| Condition d’Accès | Bac de série C, D ou Technique | Etre titulaire de licence professionnelle |
| Durée de Formation | 3 ans | 2 ans |
| Diplôme délivré | Diplôme de Licence Professionnelle | Diplôme de Master Professionnel  Diplôme de Master Recherche |

L’accès en première année de MASTER se fait automatiquement pour les étudiants de l’Ecole qui ont obtenu le diplôme de Licence Professionnelle.

Le Master Recherche permet à son titulaire de poursuivre directement des études en doctorat et de s’inscrire directement dans une Ecole Doctorale.

Les Ecoles Doctorales jouissent d’une autonomie de gestion par rapport aux Etablissements de formation universitaire.

Il convient de signaler que par arrêté ministériel N° 21.626/2012 – MESupRES publié le 9 Août 2012 par la Commission Nationale d’habilitation (CNH), l’Ecole Doctorale « Modélisation – Informatique » a été habilitée pour l’Université de Fianarantsoa.

Depuis l’année universitaire 2010-2011, l’ENI s’est mise à organiser des formations hybrides en informatique dans les différentes régions (Fianarantsoa, Toliara) en raison de l’insuffisance de la capacité d’accueil des infrastructures logistiques. En effet, le système de formation hybride semi - présentielle utilise la visioconférence pour la formation à distance.

Bien qu’il n’existe pas encore au niveau international de reconnaissance écrite et formelle des diplômes délivrés par l’ENI, les étudiants diplômés de l’Ecole sont plutôt bien accueillis dans les instituts universitaires étrangers (Canada, Suisse, France…).

### Relation de l’ENI avec les entreprises et organismes

Les stages effectués chaque année par les étudiants mettent l’Ecole en rapport permanent avec plus de 300 entreprises et organismes publics, semi-publics et privés, nationaux et internationaux.

L’Ecole dispose ainsi d’un réseau d’entreprises, de sociétés et d’organismes publics et privés qui sont des partenaires par l’accueil en stage de ses étudiants, et éventuellement pour le recrutement après l’obtention des diplômes par ces derniers.

Les compétences que l’Ecole cherche à développer chez ses étudiants sont l’adaptabilité, le sens de la responsabilité, du travail en équipe, le goût de l’expérimentation et l’innovation.

En effet, la vocation de l’ENI est de former des techniciens supérieurs de niveau LICENCE et des ingénieurs de type généraliste de niveau MASTER avec des qualités scientifiques, techniques et humaines reconnues, capables d’évoluer professionnellement dans des secteurs d’activité variés intégrant l’informatique.

Les stages en milieu professionnel permettent de favoriser une meilleure adéquation entre les formations à l’Ecole et les besoins évolutifs du marché de l’emploi. Les principaux débouchés professionnels des diplômés de l’Ecole concernent les domaines suivants :

L’informatique de gestion d’entreprise :

* les technologies de l’information et de la communication (TIC),
* la sécurité informatique des réseaux,
* l’administration des réseaux et des systèmes,
* les services bancaires et financiers, notamment le Mobile Banking,
* les télécommunications et la téléphonie mobile,
* les Big Data,
* le commerce, la vente et l’achat, le Marketing,
* l’ingénierie informatique appliquée,
* l’écologie et le développement durable.

Parmi les sociétés, entreprises et organismes partenaires de l’Ecole, on peut citer : ACCENTURE Mauritius, Air Madagascar, Ambre Associates, Airtel, Agence Universitaire de la Francophonie ( AUF) , B2B, Banque Centrale, BFG-SG, BIANCO, BLUELINE, Bureau national de gestion des Risques et des catastrophes (BNGRC), CEDIIFianarantsoa, Data Consulting, Central Test, Centre National Antiacridien, CNRE, CHU, CNRIT, COLAS, Direction Générale des Douanes, DLC, FID, FTM, GNOSYS, IBONIA, INGENOSIA, INSTAT, JIRAMA, JOUVE, MADADEV, MEF, MEN, MESupRES, MFB, MIC, Min des postes/Télécommunications et du Développement Numérique, NEOV MAD, Ny Havana, Madagascar National Parks, OMNITEC, ORANGE, OTME, PRACCESS, QMM Fort-Dauphin, SMMC, SNEDADRS Antsirabe, Sénat, Société d’Exploitation du Port de Toamasina (SEPT), SOFTWELL, Strategy Consulting, TELMA, VIVETEC, Société LAZAN’I BETSILEO, WWF …

L’organisation de stage en entreprise continue non seulement à renforcer la professionnalisation des formations dispensées, mais elle continue surtout à accroître de façon exceptionnelle les opportunités d’embauche pour les diplômés de l’Ecole.

### Partenariat au niveau international

Entre 1996 et 1999, l’ENI avait bénéficié de l’assistance technique et financière de la Mission Française de Coopération et d’action culturelle dans le cadre du Programme de Renforcement de l’Enseignement Supérieur (PRESUP) consacré à l’Ecole a notamment porté sur :

* une dotation en logiciels, micro-ordinateurs, équipements de laboratoire de maintenance et de matériels didactiques :
* la réactualisation des programmes de formation assortie du renouvellement du fonds de la bibliothèque ;
* l’appui à la formation des formateurs ;
* l’affectation à l’Ecole d’Assistants techniques français.

De 2000 à 2004, l’ENI avait fait partie des membres du bureau de la Conférence Internationale des Ecoles de formation d’Ingénieurs et Technicien d’Expression Française (CITEF).

Les Enseignants-Chercheurs de l’Ecole participent régulièrement aux activités organisées dans le cadre du Colloque Africain sur la Recherche en Informatique (CARI)

L’ENI avait également signé un accord de coopération interuniversitaire avec l’Institut de Recherche en Mathématiques et Informatique Appliquées (IREMIA) de l’Université de la Réunion, l’Université de Rennes 1, l’INSA de Rennes, l’Institut National Polytechnique de Grenoble (INPG).

A partir du mois de Juillet 2001, l’ENI avait abrité le Centre de Réseau Opérationnel (Network Operating Center) du point d’accès à Internet de l’Ecole ainsi que de l’Université de Fianarantsoa. Grâce à ce projet américain qui a été financé par l’USAID Madagascar, l’ENI de l’Université de Fianarantsoa avait été dotées d’une ligne spécialisée d’accès permanent au réseau Internet.

L’ENI avait de même noué des relations de coopération avec l’Institut de Recherche pour le Développement (IRD).

L’objet du projet de coopération avait porté sur la modélisation environnementale du Corridor forestier de Fandriana jusqu’à Vondrozo (COFAV). Dans ce cadre, un atelier scientifique international avait été organisé à l’ENI en Septembre 2008. Cet atelier scientifique avait eu pour thème de modélisation des paysages.

Et dans le cadre du programme scientifique PARRUR, l’IRD avait financé depuis 2010 le projet intitulé « Forêts, Parcs et Pauvreté dans le Sud de Madagascar (FPPSM). Des étudiants en DEA et des Doctorants issus de l’ENI avaient participé à ce Programme.

Par ailleurs, depuis toujours la même année 2010, l’ENI de Fianarantsoa avait été sélectionnée pour faire partie des organismes partenaires de l’Université de Savoie dans le cadre du projet TICEVAL relatif à la certification des compétences en TIC ;

Le projet TICEVAL avait été financé par le Fonds Francophone des Inforoutes pour la période allant de 2010 à 2012, et il avait eu pour objectif de généraliser la certification des compétences en Informatique et Internet du type C2i2e et C2imi.

Dans le cadre du projet TICEVAL, une convention de coopération avec l’Université de Savoie avait été signée par les deux parties concernées. La mise en œuvre de la Convention de 13 Coopération avait permis d’envoyer des étudiants de l’ENI à Chambéry pour poursuivre des études supérieures en Informatique.

Enfin et non des moindres, l’ENI avait signé en Septembre 2009 un protocole de collaboration scientifique avec l’ESIROI – STIM de l’Université de la Réunion.

Comme l’ENI constitue une pépinière incubatrice de technologie de pointe, d’emplois et d’entreprises, elle peut très bien servir d’instrument efficace pour renforcer la croissance économique du pays, et pour lutter contre la Pauvreté.

De même que le statut de l’Ecole devrait permettre de renforcer la position concurrentielle de la Grande Ile sur l’orbite de la modélisation grâce au développement des nouvelles technologies.

### Débouchés professionnels des diplômes

Le chômage des jeunes diplômés universitaires fait partie des maux qui gangrènent Madagascar. L’environnement socio-politique du pays depuis 2008 jusqu’ à ce jour a fait que le chômage des diplômés est devenu massif par rapport aux établissements de formation supérieure existants.

Cependant, les formations proposées par l’Ecole permettent aux diplômés d’être immédiatement opérationnels sur le marché du travail avec la connaissance d’un métier complet lié à l’informatique aux TIC.

L’Ecole apporte à ses étudiants un savoir-faire et un savoir-être qui les accompagnent tout au long de leur vie professionnelle. Elle a une vocation professionnalisante.

Les diplômés en LICENCE et en MASTER issus de l’ENI peuvent faire carrière dans différents secteurs.

L’Ecole bénéficie aujourd’hui de 38 années d’expériences pédagogiques et de reconnaissance auprès des sociétés, des entreprises et des organismes. C’est une Ecole Supérieure de référence en matière informatique.

Par conséquent, en raison de fait que l’équipe pédagogique de l’Ecole est expérimentée, les enseignants-chercheurs et les autres formateurs de l’Ecole sont dotés d’une grande expérience dans l’enseignement et dans le milieu professionnel.

L’Ecole est fière de collaborer de façon régulière avec un nombre croissant d’entreprises, de sociétés et d’organismes publics et privés à travers les stages des étudiants. Les formations dispensées à l’Ecole sont ainsi orientées vers le besoin et les attentes des entreprises et des sociétés.

L’Ecole fournit à ses étudiants de niveau LICENCE et MASTER des compétences professionnelles et métiers indispensables pour les intégrer sur le marché du travail.

L’Ecole s’efforce de proposer à ses étudiants une double compétence à la fois technologique et managériale combinant l’informatique de gestion ainsi que l’administration des réseaux et systèmes.

Tableau 4 Débouchés professionnels éventuels des diplômes

|  |  |
| --- | --- |
| LICENCE | * Analyste * Programmeur * Administrateur de Site Web/ de portail Web * Assistant Informatique et Internet * Chef de projet Web ou MultiMedia * Développeur Informatique ou MultiMedia * Intégrateur Web ou Web Designer * Hot Liner/ Hébergeur Internet * Agent de référencement * Technicien/Supérieur de help desk sur Informatique * Responsable de sécurité web * Administrateur de réseau * Administrateur de cybercafé |
| MASTER | * Administrateur de cyber café * Administrateur de réseau et système * Architecture de système d’information * Développeur d’application /web /java/Python/ IOS /Android * Ingénieur réseau * Webmaster /web designer * Concepteur Réalisateur d’applications * Directeur du système de formation * Directeur de projet informatique * Chef de projet informatique * Responsable de sécurité informatique * Consultant fonctionnel ou freelance |

### Ressources humaines

Les ressources humaines affectées à l’Ecole Nationale d’Informatique sont composées de :

* Directeur de l’Ecole : Professeur RAMAMONJISOA Bertin Olivier
* Responsable de Mention : Monsieur RABETAFIKA Louis Haja
* Responsable de Parcours « Génie Logiciel et Base de Données » : Monsieur RALAIVAO Jean Christian
* Responsable de Parcours « Administration Systèmes et Réseaux » : Monsieur SIAKA
* Responsable de Parcours « Informatique Générale » : Monsieur GILANTE Gesazafy
* Nombre d’Enseignants permanents : 12 dont un (01) Professeur Titulaire, un (01) Professeur, Cinq (05) Maîtres de Conférences et cinq (05) Assistants d’Enseignement Supérieur et de Recherche.
* Nombre d’Enseignants vacataires : 10

Personnel Administratif : 23

## Présentation de MORALOGIQ

### Historique et missions

La société Moralogiq est une entreprise au service du numérique spécialisé dans la transformation numérique à Madagascar. Elle est fondée en décembre 2020.

Moralogiq propose de rendre la transformation digitale accessible à tous.

### Situation juridique

1. Raison sociale : MORALOGIQ
2. Statut : S.A.R. L
3. Numéro d’Identification Fiscale : 4 004 321 595
4. Numéro Statistique : 62011 11 2020 0 12100
5. Siège : Manakambahiny Antananarivo

### Organigramme

L’organigramme de Moralogiq est représenté sur la figure 2.

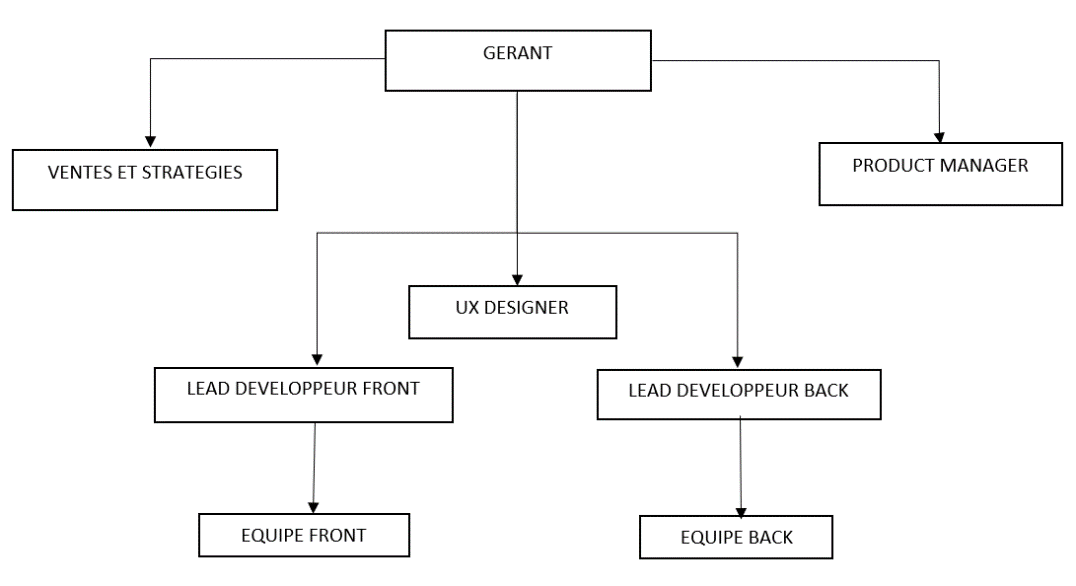


Figure 2. Organigramme du société MORALOGIQ

### Services

Elle propose les services suivants :

* Elaboration de terme de référence,
* Rédaction des spécifications techniques / cahiers des charges,
* Création de Site vitrine, Site catalogue, Site e-commerce,
* Hébergement de site
* Développement d’Application Web, Application Desktop, Application mobile
* Externalisation de données

### Contacts

Personne habilité à représenter la société :

Nom  : RAKOTOARISON Patrick

Numéro de téléphone : +261 34 76 924 63

Adresse électronique : [kotoarison@gmail.com](mailto:kotoarison@gmail.com)

Site web : [www.moralogiq.fr](http://www.moralogiq.fr)

## Description du projet

La description du projet est primordiale avant d’entamer l’analyse et conception. Ce chapitre nous permettra de décrire le projet, spécifier les objectifs du projet, organiser le projet.

### Formulation

Après avoir enseigné de façon traditionnelle, l’institut européen de synergologie s’intéresse maintenant à la nouvelle technologie de l’information et de la communication. Pour ce faire, l’analyse et la conception sont entamées par l’équipe de Moralogiq.

Ainsi le travail consiste à donner vie à ce projet et de satisfaire le client d’où le projet de conception et réalisation d’une application web pour la gestion de la relation humaine par la communication non verbale ou bien la synergologie.

La synergologie est une pseudoscience qui opère dans la relation humaine par communication non verbale ; elle s’intéresse au micro geste du corps humain et de donnée une signification logique.

### Objectif et besoins d’utilisateur

#### Objectif du projet

L’objectif est de concevoir et réaliser une application web pour la gestion de la relation humaine par communication non verbale.

#### Besoin de l’utilisateur

Les utilisateurs souhaiteront avoir une application qui contiendra :

* S’authentifier
* La gestion des utilisateurs
* La gestion d’accès aux plateformes
* La gestion des vidéos uploadées depuis YouTube
* La gestion d’un arbre de codification
* La notification des activités dans la plateforme
* La gestion des items secondaires

### Moyen nécessaire à la réalisation du projet

#### Moyens humains

Vu que la société travaille en agile, l’équipe doit au moins contenir un product owner, un scrum master, une équipe de développement et de tests qui sont représentés sur le tableau 5.

Tableau 5 Moyens humain

|  |  |  |
| --- | --- | --- |
| Fonction | Nombre | Commentaire |
| Product Owner | 1 | Représente le client |
| Chef de projet | 1 | Dirige le projet |
| Développeur | 5 | 2 développeurs back end  1 développeur front end  2 intégrateurs |
| Designer | 1 | S’occupe de la conception visuel |
| Testeur | 1 | Tester l’application |

#### Moyens matériels

Tableau 6. Inventaire des moyens matériels

|  |  |  |  |
| --- | --- | --- | --- |
| Désignation | Caractéristique | Quantité | Utilité lors du projet |
| Ordinateur portable | Marque : LAPTOP ONKYO  CPU : intel core i7-3820QM  RAM : 32GB  DD : 500 GB SSD + 128 GB SSD, | 2 | Réalisation du back |
| Ordinateur portable | Marque : Asus ROG  CPU : corei7  RAM : 8 GB  DD : 500 Go HDD | 1 | Réalisation du front |
| Ordinateur portable | Marque : Asus ROG  CPU : corei7  DD : 500 Go HDD | 1 | Intégration |
| Ordinateur portable | MSI corei7 1To HDD | 1 | Design |
| Serveur VPS | CPU : Intel (R) Xeon ES-2650 V3 RAM : 4GB | 1 | Serveur pré prod |
| Imprimante | Imprimante HP ink Tank 410 series, | 1 | Impression des documents utilisé pour la mise en œuvre du projet |
| Internet | WIFIBER PRO prémium ORANGE | 1 | La connexion internet |
| Internet | FIBRE OPTIQUE TELMA | 1 | La connexion internet |

#### Moyens logiciels

Le tableau 7 représente les logiciels utilisés pour la mise en œuvre du projet.

Tableau 7 Inventaire des moyens logiciels

|  |  |  |
| --- | --- | --- |
| Designation | Version | Utilité durant le stage |
| PhpMyAdmin | 5.1.1 | Administration de la base de données |
| MariaDB |  |  |
| phpStorm | 2020.1 | IDE |
| Visual Paradigm | 2016 | Conception |
| Composer | 2 | Gestionnaire de paquet PHP |
| postman | **8.11.1** | Client http |
| adobeXD | 43.0.12.14 | Conception visuel de l’application |
| Just Color picker | 5.5 | Intégration palette couleur |
| slack | 4.19.3 | Communication |
| gitlab | Néant | Version |
| Jira | Néant | Gestion de projet |
| Trello | néant | Stockage des informations nécessaires au projet |
| Google meet | Néant | Daily meeting |
| Gantt Project | 3.1.3102 | Réalisation chronogramme de travaux |

### Résultats attendus

Les résultats attendus sont :

* Une application avec une interface intuitive
* Les données doivent être structurées
* L’accès à l’application doit être rapide et sécuriser
* La gestion de l’arbre de classification mentale doit faire partie de l’application
* L’application doit pouvoir évoluer c’est-à-dire que l’application doit être facile à maintenir.
* L’application doit avoir une documentation
* L’application doit pouvoir uploader une vidéo YouTube dans le serveur
* L’application doit permettre d’analyser une vidéo full HD
* L’application doit permettre une exportation de script d’analyse

### Chronogramme des Travaux

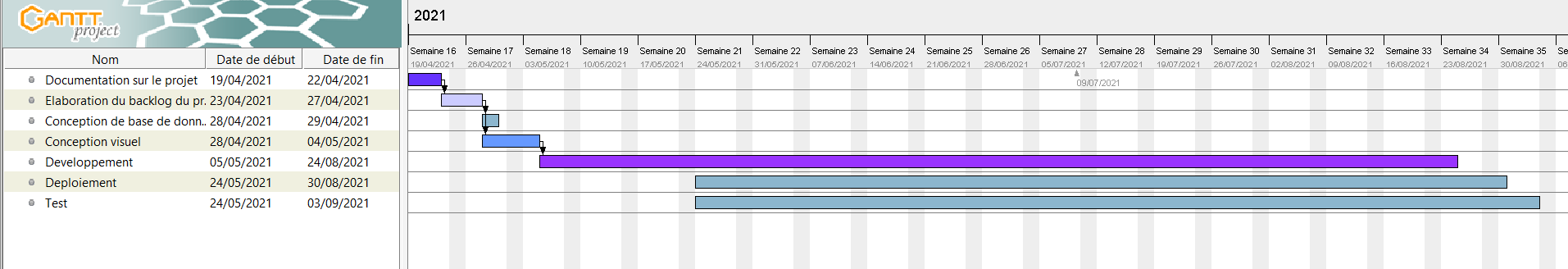
Pour bien organiser les tâches à réaliser, nous avons suivi le planning représenté par le calendrier présenté dans la figure 3. 

Figure Chronogramme des travaux

# ANALYSE ET CONCEPTION

## Analyse préalable

La conception et la réalisation de ce projet nécessitent une analyse de la  
situation existante et une analyse conceptuelle afin de recueillir les  
informations nécessaires sut l’existant et les attentes de l’utilisateur.

### Analyse de l’existant

Cette partie est d’une importance capitale. Elle va nous permettre de décrire toutes les  
facettes des fonctionnalités de notre système, tout en tenant compte d’une question pionnière à laquelle on répondra subséquemment : que fera notre système ainsi découvrir les critiques  
qu’on peut fournir à l’ancien système.

#### Organisation actuelle

A l’heure actuelle, pour devenir un spécialiste du langage corporel, l’IES (Institut européen de synergologie) propose plusieurs formules de formation :

* Une formation certifiant réalisable en présentiel
* Une formation certifiant à distance en Visio formation niveaux1 et 2
* Des formations intra entreprises

La formation se fait de façon traditionnelle. On visualise la vidéo. On apprend par cœur la classification mentale. Ensuite on transcrit les codes en texte un à un.

#### Inventaire des moyens matériels et logiciels

Le tableau 8 récapitule les moyens matériels et logiciels existant actuellement :

Tableau 8 Inventaire des matériels de l'existant

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Désignation | Caractéristique | Nombres | Usages | Type |
| Ordinateur | Core i5  RAM: 16Go  HDD: 500Go | 20 | Pour les étudiants | Matériel |
| Site web | néant | 1 | Diffusion d’information | Logiciel |
| Google meet | néant | néant | Meeting | Logiciel |

### Critique de l’existant

Nombreuses sont les critiques constatées lors de l’analyse de l’existant. Ces critiques découragent les clients d’adhérer à une formation chez l’IES.

* Perte de temps pour parcourir la base de donnée
* L’institut n’a pas encore de plateforme
* Les données récoltées ne sont pas exploitables
* Perte de temps pendant la correction des analyses
* Pas de système d’historisation

Malgré les critiques, il y a aussi des points forts de l’application existant comme :

* La classification mentale qui aide les étudiants.
* La formation présentielle

### Conception avant-projet

#### Comparaison de solutions

Afin de remédier aux problèmes, nous avons proposé 2 solutions :

Solution 1 : commander logiciel d’analyse vidéo auprès un entreprise concepteur de logiciel

Solution 2 : développer et concevoir un logiciel sur mesure

Le tableau 9 compare les 2 solutions proposées

Tableau 9 Comparaison des solutions proposées

|  |  |  |
| --- | --- | --- |
| Solutions | Avantages | Inconvénients |
| Solution 1 | Gains de temps | Cout élevé (des milliers d’euros)  Perte de temps dans la recherche d’une entreprise |
| Solution 2 | Sur mesure  Adapté aux besoins  Maintenable  Donnée sécurisée | Temps de développement peut être élevé |

La solution qui convient le plus à notre situation est la solution 2 qui est le développement d’un logiciel sur mesure, car les avantages cités sont meilleurs que les 2 autres solutions.

#### Choix et justification

À l’issue de la formulation des besoins, on termine par l’approfondissement des solutions. Et dans cette phase on procède aux choix des outils (moyen logiciel).

La phase consiste à étudier parmi un ensemble de solutions potentielles, la solution la plus ajustée sur des aspects fondamentaux.

Afin de bien développer notre projet avec rapidité et efficacité, nous devons d’abord choisir des outils pour nous aider, comme un serveur, un IDE, et même un outil de modélisation.

##### Méthode

Le tableau 10 montre la comparaison entre MERISE et le Processus unifié.

Tableau 10 Comparaison entre MERISE et Processus unifié

|  |  |  |
| --- | --- | --- |
| Méthodes | Avantages | Inconvénients |
| MERISE | Le modèle en nombre limité (06) est progressivement élaboré, enrichi, et constitue des supports de communication et de participation pour les utilisateurs.  MERISE permet de modéliser le métier de l’entreprise indépendamment des techniques, aux niveaux conceptuel et organisationnel.  Cycle de vie séquentielle | Méthode non orientée objet  MERISE n’exprime pas certaines notions (agrégation et composition)  Merise s’appuie sur la décomposition des modules pour la règle de gestion alors que ceux-là sont difficilement extensibles et exploitables par de nouveaux systèmes. |
| Processus unifié | Piloté par le cas d’utilisation.  Démarche centrée sur l’architecture.  Démarche itérative et incrémentale.  Cycle de vie incrémentale et itérative.  Utilise le langage de modélisation UML.  Exprime certaines notions (agrégation, composition)  Ne préconise aucune démarche spécifique. | Les niveaux d’abstraction ne sont pas nettement distingués dans UP  UP ne se préoccupe pas de normalisation (2e ou 3e forme normale).  La mise en pratique de l’UP nécessite un apprentissage et passe par une période d’adaptation. |

On a choisi la méthode PU parce qu’elle utilise le langage de modélisation UML et aussi parce qu’elle permet d’avoir un cycle de vie itérative et incrémentale.

##### Outils de modélisation

Pour mener à bien ce projet, on aura besoin d’un outil de conception pour faire la  
modélisation du système ainsi que pour anticiper les scénarios des cas d’utilisation à travers des diagrammes.

Le tableau 11 illustre la comparaison des 3 outils de conception tels que : ArgoUML, Modelio et Visual Paradigm.

Tableau 11 Comparaison des outils de conception

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Nom | Plateforme | Ergonomie | Types de graphiques pris en charge | Autre |
| ArgoUML | Multi plateforme (java) | Facile d’utilisation pour débutant mais pas pour les professionnels | Cas d’utilisation, Classe, Etat, Activité, Séquence, Collaboration, Déploiement | Génération de code en langage Java |
| Modelio | Windows, Linux | Programme académique Modelio pour intégrer l’outil dans le programme d’étude Successeur d’obecteering | Cas d’utilisation, Classe, Objet, Etat, Activité, Séquence, Composant, Déploiement, Package, BPMN, Communication | Supporte tout l’UML2 Intègre BPMN Génération de code HTML, MS WORD, Open document Modélisation et application de design Patterns |
| Visual Paradigm for UML (VP) | Multi plateforme(Java) | Dispose de différentes fonctions facilitant la conception des diagrammes aux débutants | Cas d’utilisation, Classe, Objet, Etat, Activité, Séquence, Communication, Composant, Déploiement, Package, Structure composite, Chronogramme | Génération de code source à partir des diagrammes Réalisation du mapping objet relationnel automatiquement à partir du diagramme de classe |

D’après le tableau comparatif, l’outil qui nous convient le mieux est visual paradigm en raison qu’il est multiplateforme et qu’il intègre tous les diagrammes de l’UML.

##### Langage de modélisation utilisé

Pour modéliser notre application, nous avons choisi la notation UML.

La modélisation est le pilier de toute activité qui conduit au déploiement de logiciels de qualité. Les modèles sont construits pour spécifier la structure et le comportement attendus d’un système, pour visualiser et contrôler son architecture, pour mieux comprendre son organisation et ainsi déceler les possibilités de simplification et de réutilisation et, enfin, pour gérer les risques encours.

Unified Modeling Language ou UML est un langage de modélisation objet officiellement approuvé en 1997 par l’OMG (Objet Management Group). UML « unifie » des méthodes de conception logicielle orientées objet telles que Booch, OMT et OOSE qui coexistaient jusque-là sans beaucoup de compatibilité entre elles. UML les rassemble non pas en proposant une synthèse, mais en créant un « langage de modélisation », c’est-à-dire une notation unique pour faciliter la conception de programmes. C’est un langage graphique de modélisation des données et des traitements. UML n’est pas à l’origine des concepts objets, mais en constitue une étape majeure, car il unifie les différentes approches et en donne une définition plus formelle.

##### Système de gestion de base de données

La plupart des SGBD actuels sont construits autour du modèle relationnel pour simplifier la description des données. Dans le modèle relationnel, la structure de données est formée par un système de relation. Les classes d’objets et les liens sont représentés sous forme de relation. Nombreux de ces SGBD sont disponibles sur le marché, partant des plus petits jusqu’aux plus grands destinés spécialement aux professionnels. Chacun de ces SGBD possède des avantages et des inconvénients.

Le tableau 12 résume le comparatif des SGBD

Tableau 12 Tableau comparatif entre MySQL et PostgreSQL

|  |  |  |
| --- | --- | --- |
| SGBD | Avantages | Inconvénients |
| MySQL | • Plus léger que les autres SGBD  • Multiplateforme, OpenSource et gratuit  • Possède une plus grande base d’utilisateurs  • Solution très courante en hébergement public  • Très bonne intégration dans l'environnement Apache/PHP  • Version cluster depuis la version 4  • Facilité de déploiement et de prise en main.  • Plusieurs moteurs de stockage adaptés aux différentes problématiques, configurable au niveau table. | •Pas de vue matérialisée,  •Support incomplet des triggers et procédures stockées,  •Ne supporte pas une énorme quantité |
| PostgreSQL | • OpenSource et gratuit  • Fiable et relativement performant, tout en restant simple d'utilisation  • Très riche fonctionnellement, notions d'héritage de tables, multitude de modules  • Simple d'utilisation et d'administration  • Héritage de tables | • Les tables sont obligatoirement transactionnelles  • La modification du fichier de sécurité pg\_hba.conf nécessite un reboot pour être prise en compte  • Pas possible de requêter sur plusieurs bases à la fois : chaque base nécessite sa connexion  • Sauvegardes peu évoluées  • Supporte les bases de moyenne importance  • Pas de services Web  • Pas de requêtes récursives  • Solutions de réplication pas encore totalement packagées  • Solution en cluster pas finalisée |

On a choisi MySQL en raison de sa stabilité et le d’innombrable de communauté d’utilisateur qu’il possède.

* MySQL

MySQL est un système de gestion de base de données (SGBD). Il est distribué sous une double licence GPL et propriétaire. Il fait partie des logiciels de gestion de base de données les plus utilisés au monde1, autant par le grand public (applications web principalement) que par des professionnels.

MySQL est un serveur de bases de données relationnelles SQL développé dans un souci de performances élevées en lecture, ce qui signifie qu'il est davantage orienté vers le service de données déjà en place que vers celui de mises à jour fréquentes et fortement sécurisées.

Il est multi-thread et multi-utilisateur. C'est un logiciel libre développé sous double licence selon qu'il est distribué avec un produit libre ou avec un produit propriétaire. Dans ce dernier cas, la licence est payante, sinon c'est la licence publique générale GNU (GPL) qui s'applique. Un logiciel qui intègre du code MySQL ou intègre MySQL lors de son installation devra donc être libre ou acquérir une licence payante.

Cependant, si la base de données est séparée du logiciel propriétaire qui ne fait qu'utiliser des API tierces (par exemple en C# ou php), alors il n'y a pas besoin d'acquérir une licence payante MySQL comme dans le cas qui se présente lors de la réalisation de ce projet. MySQL fonctionne sur de nombreux systèmes d'exploitation différents, incluant Linux, Mac OS X, Solaris, Windows.

* Langages

MySQL supporte deux langages informatiques, le langage de requête SQL et le SQL/PSM (Persistent Stored Modules), une extension procédurale standardisée au SQL, introduit dans la version 5 de MySQL, permet de combiner des requêtes SQL et des instructions procédurales (boucles, conditions...), dans le but de créer des traitements complexes destinés à être stockés sur le serveur de base de données (objets serveur), comme par exemple des procédures stockées ou des déclencheurs (rudimentaires).

* Transactions

MySQL est un SGBD transactionnel. Il est capable de préparer des modifications sur les données d'une base et de les valider ou de les annuler d'un bloc. Cela garantit l'intégrité des informations stockées dans la base.

Lors d'une transaction, les blocs de données contenant les lignes de données modifiées par cette transaction sont verrouillés. Les autres utilisateurs, en fonction du niveau d'isolation choisi, doivent attendre ou non la fin de la transaction pour pouvoir les modifier à nouveau. Les verrouillages s'effectuent au niveau des lignes, pages, extensions, tables ou base de données. MySQL ne verrouille que les ressources dont il a besoin (par défaut les enregistrements) et en fonction des besoins peut verrouiller à un niveau plus élevé (pages ou objet). Cela évite aux utilisateurs d'attendre la fin d'une transaction pour mettre à jour des lignes de données qui n'ont pas été touchées par une modification et permet de diminuer la quantité de ressources consommées.

* Procédures stockées

Il est possible de définir des procédures stockées. Une procédure stockée est une suite d'instructions qui vont avoir des effets sur la base de données ou qui renvoient une ou plusieurs valeurs.

Les procédures stockées sous MySQL peuvent prendre en paramètre et/ou retourner des entiers, des chaines de caractère, des dates, des curseurs, des tables, des tables virtuelles et tout autre type défini dans MySQL par défaut ou par les utilisateurs. Les principaux avantages de l'utilisation des procédures stockées sont :

* La transaction de traitements complexes,
* La sécurité accrue,
* La réutilisation des plans compilés (contrairement aux vues),
* La possibilité de gérer les dépendances entre le code SQL et les objets du moteur.

##### Serveur

Pour notre environnement de développement local, nous avons installé séparément les serveurs sur Linux : PHP, MySQL, PhpMyAdmin.

Le fait de les séparer nous fait bénéficier en performance pique sous Linux, l’environnement de développement est très léger et plus rapide. Et on n’aura pas besoin d’apache.

##### IDE

Le choix d’un bon environnement de développement intégré est primordial avant de  
commencer à développer une application. Les principaux IDE pour PHP sont PHPStorm, NetBeans et Eclipse.

Le tableau 13 montre la comparaison entre PHPStorm

Tableau 13 Comparatif des IDE

|  |  |  |  |
| --- | --- | --- | --- |
| Critères | PHPStorm | NetBeans | Eclipse |
| Editeur | JetBrain | NetBeans | Eclipse |
| Supporte PHP | OUI | OUI | OUI |
| Edition de code et réfactoring | OUI | OUI | OUI |
| Intégration des frameworks PHP | OUI | OUI | OUI |
| Modélisation | OUI | OUI | OUI |
| Interface | Complexe | Intuitive | Complexe |
| Fonctionnalité | Excellent | Excellent | Excellent |
| Performance | Moins gourmand en ressources | Gourmand en ressources | Gourmand en ressources |

D’après ce tableau, le choix se penche vers PHPStorm en raison de sa consommation en ressource et son excellence en matière d’intégration des Framework PHP.

##### Système de gestion de version

Un logiciel de gestion de versions est un logiciel qui permet de stocker un ensemble de fichiers en conservant la chronologie de toutes les modifications qui ont été effectuées dessus. Il permet notamment de retrouver les différentes versions d'un lot de fichiers connexes [23].Il est indispensable lorsqu’on travaille à plusieurs sur un projet afin d’en assurer l’efficacité. Il existe plusieurs outils de gestion de version tels que CVS, SVN, Mercurial, Git.… Mais nous allons faire un petit zoom sur SVN et Git. Le tableau 14 va montrer leurs différences.

Tableau Comparaison entre Git et SVN

|  |  |  |
| --- | --- | --- |
|  | **SVN** | **Git** |
| **Licence** | Logiciel Libre | Logiciel Libre |
| **Disponibilité** | Linux, Windows, MACOSX | Linux, Windows, MACOSX |
| **Versionning** | Centralisé | Décentralisé |
| **Dépôt** | Un dépôt central dans lequel les copies de travail sont créées | Des copies de dépôt, Présentes localement, dans lesquelles il est possible de travailler |
| **Droit d’accès** | Basé sur le chemin | Pour le répertoire complet |
| **Suivi des modifications** | Enregistre des données | Enregistre des données |
| **Journal de modifications des données** | Complet seulement dans le dépôt. Les copies de travail ne contiennent que la version la plus récente. | Le dépôt et les copies de travail contiennent l’historique complet. |
| **Quelques commandes** | *svnadmin create* : créer un nouveau dépôt  *svn import* : importer un projet dans le dépôt  *svn checkout* : lire tout un projet  *svn update* : lire/mettre à jour depuis le dépôt  *svn commit* : écrire/modifier le  dépôt (nouvelle révision)  *svn status* : état de la copie  locale *svn add* : ajouter un fichier  *svn rm* : enlever un fichier *svn mv* : déplacer un fichier *svn mkdir* : créer un répertoire | *git init* : créer un dépôt local git *git add* : ajouter un élément git commit : propager sa version sur le dépôt *git clone* : copier un dépôt existant (/distant)  *git pull* : tirer les modifications du dépôt distant  vers le dépôt local *git push* : pousser les modifications du dépôt local vers le dépôt distant |

Nous avons choisi Git comme système de gestion de version parce qu’il est performant et est décentralisé, très pratique pour des développements en équipe.

##### Choix du langage de programmation

Un langage de programmation est une notation conventionnelle destinée à formuler des algorithmes [24]. On va voir dans le tableau 15 un comparatif entre les différents langages de programmation destiné au Web.

**Comparaison avec les concurrents :**

Il existe plusieurs langages de programmation qui pourraient être choisis pour réalisercette application … Cependant, une comparaison doit être établie afin de savoir le langage lemieux adapté à notre projet. Le tableau 15 montre l’étude comparative sur ces trois langages.Il est possible de développer une application web utilisant des Frameworks sur beaucoupde langage de programmation.

En général, quatre (3) solutions dominent le marché du web en ce moment :

* PHP
* JSP
* C#

Tableau 15 Comparatif des langages de programmation

|  |  |  |
| --- | --- | --- |
|  | Avantages | Incovénients |
| PHP | ● Gratuit  ● Variété de Framework Open Source  ● Facilité de déploiement  ● ORM :  ● Courbe d’apprentissage  ● L’hébergement est supporté presque partout  ● Portabilité | ● N’est pas adapté pour des traitements asynchrones |
| JSP | ● Portable  ● Plus grande stabilité de code  ● Sécurisé  ● Langage fortement typé  ● Soutenu par Oracle | ● L’hébergement de JSP n’est pas disponible chez tous les hébergeurs  ● Il est plus demandant au niveau du microprocesseur |
| C# | ● Il est compilé ? rapide d’exécution  ● Le langage est soutenu par Microsoft | ● L’hébergement se fait exclusivement sous Windows server.  ● À la merci des choix technologiques de Microsoft. |

En tenant compte du langage de programmation pour le développement de l’application et ses avantages sur tous les langages comparés tel qu’elle est gratuite, compatible sur presque tout l’hébergement, cela réduit considérablement la marge de choix.

o **PHP :**

L’HyperText PreProcessor, plus commun sous son sigle PHP est un langage interprété (un langage de script) exécuté du côté serveur et non du côté client. Un script principalement utilisé pour produire des pages Web dynamiques via serveur http, mais pouvant également fonctionner comme n’importe quel langage interprété de façon locale, en exécutant les programmes en lignes de commande. PHP est donc un langage de script HTML/XHTML côté serveur. Le code est inclus dans les pages HTML/XHTML à la recherche de code PHP à interpréter. Le client reçoit uniquement le résultat du script (une page HTML/XHTML « générée »).

Les versions actuelles de PHP (à partir de PHP 5) permettent de programmer orienté objet, en créant des classes contenant des attributs et des méthodes, des instances de classes.

* **Justification :**

On constate à travers ce tableau que ces langages sont à peu près similaires, mais on observe également que PHP a des avantages par rapport à ces trois langages.

* Rapide, Portable, dynamique
* Orienté objet : cela permet de créer des programmes modulaires et codes réutilisables ; - Indépendante de la plateforme ;
* Possède une capacité de se déplacer facilement d’un système informatique à un autre ;
* Possède une capacité à exécuter le même programme sur de nombreux Systèmes différents ;
* Robuste ;
* Fiable : PHP met beaucoup d’accent sur la vérification pour d’éventuelles erreurs, que des compilateurs PHP sont en mesure de détecter de nombreux problèmes qui seraient d’abord montrés au cours de délais d’exécution dans d’autres langues ;
* Multithread : multithread est une possibilité pour un programme à exécuter plusieurs tâches simultanément dans un programme. En PHP, la programmation multithread a été intégrée harmonieusement ;
* Facile à apprendre et à programmer.

Il est donc tout à fait naturel que notre choix soit fixé sur PHP.

Pour remédier au fait que PHP n’ait pas une structure prédéfinie, nous allons lui assigner un Framework.

##### Framework

L’utilisation de Framework présente plusieurs avantages : l’accélération du temps de développement permet de mieux structurer le code et d’avoir une bonne architecture de codage, la conservation d’un code homogène et une pérennité certaine des projets. Enfin, les applications sont plus facilement sécurisées.

Les Frameworks PHP contiennent nativement une innombrable quantité de fonctions et de classes censées faciliter le travail du développeur.

Parmi la multitude d’excellents Framework PHP nous avons choisi de comparer trois d’entre eux, adoptant tous la structure MVC (Model-View-Controller ou Modèle-Vue-Contrôleur) :

Modèle **:** c’est le comportement de l'application.

Vue **:** c’est l'interface manipulée par l'utilisateur.

Contrôleur **:** c’est le gestionnaire des évènements de mise à jour de la vue et de synchronisation des informations par le modèle.

Le tableau 16 nous montre cinq Framework les plus utilisés et renommés : Laravel, CodeIgniter et Symfony

Tableau 16 Comparaison entre les frameworks PHP

|  |  |  |
| --- | --- | --- |
| **Framework** | **Avantage** | **Inconvénients** |
|  | * Développement rapide d'applications * Tests unitaires (FAST sur HHVM) * Niveau élevé d'abstraction * Tonnes de fonctionnalités prêtes à l'emploi Paiement intégration avec stripe paquets de cryptage très forts | * Ne fonctionne pas sur les plans d'hébergement   partagé   * Fait beaucoup de requêtes sur votre base de données |
|  | * Haute performance, grâce à la mise en cache de code d'octet * Stable * Flexible * Sécurisé * Bien documenté, maintenu et pris en charge * Très bon support et très mature - Symfony propose de nombreux bundles, packages, logiciels tiers   commerciaux - Compatible PHP 5,   * Intègre l’architecture MVC et la méthode ORM, * Licence MIT (tout ce qui est réalisé par symfony peut être vendu), * Intègre Ajax, * Système de Template TWIG, * importante communauté active, * conçu pour une utilisation professionnelle. | * Bien que la documentation soit bonne, la courbe d'apprentissage est raide * Beaucoup de différences * entre les versions maintenues |
|  | * Très compatible avec les développeurs N'a besoin d'aucune dépendance ou support particulier. * Possibilité d'utiliser correctement les services d'hébergement Web normaux, en utilisant des bases de données standards telles que MySQL. * Surpasse la plupart des autres frameworks (non MVC). * Bonne documentation et LTS (support à long terme). * Simplicité de prise en main   Intégration de l’architecture MVC   * Nombreuses librairies, * Complètement orienté objet, * Extensible et modulaire, * Importante communauté très active (anglophone en majorité) - abondante documentation, * Souplesse et évolution constante. | * Pas d'espace de noms, cependant cela peut accélérer * Pas aussi convivial que les autres pour les tests unitaires * Peu de bibliothèques construites dans le cadre |

Pour la réalisation du projet, nous avons choisi Symfony 4 comme Framework PHP qui se marque par sa flexibilité et sa possession de plusieurs composants indépendants et réutilisables :

* C’est un Framework basé sur le type MVC,
* Il n’y a pas de conflits à chaque sortie de nouvelles versions de PHP,
* Une documentation claire et approfondie,
* Une communauté en ligne très active,
* Des bibliothèques et des aides intégrées,
* Une gestion facile des erreurs,
* La disposition des bibliothèques pour gérer les problématiques de sécurité.

##### Choix de l’outil de gestion de projet

Le terme logiciel de gestion de projet peut désigner différents types de logiciels ayant pour objectif de faciliter le travail de gestion de projet. Le travail des logiciels de gestion de projet est généralement d'automatiser des tâches de sauvegarde et/ou de la gestion du temps. Nous allons voir un à un les deux (02) principaux logiciels de gestion de projet : TRELLO, et JIRA.

* TRELLO :

Trello permet de créer des projets qui prennent la forme de tableaux en plusieurs colonnes personnalisables. Dans chaque colonne se trouvent les tâches sous forme de cartes que les collaborateurs font avancer de gauche à droite au fur et à mesure de leur avancement [23]. Il est gratuit. Il s'apparente plus à un gestionnaire de tâches qu'un outil de gestion de projet dans le sens où il fait l'impasse sur la gestion des couts, des temps, des budgets et les fonctions d'administration. La figure 4 représente le tableau de bord de Trello.

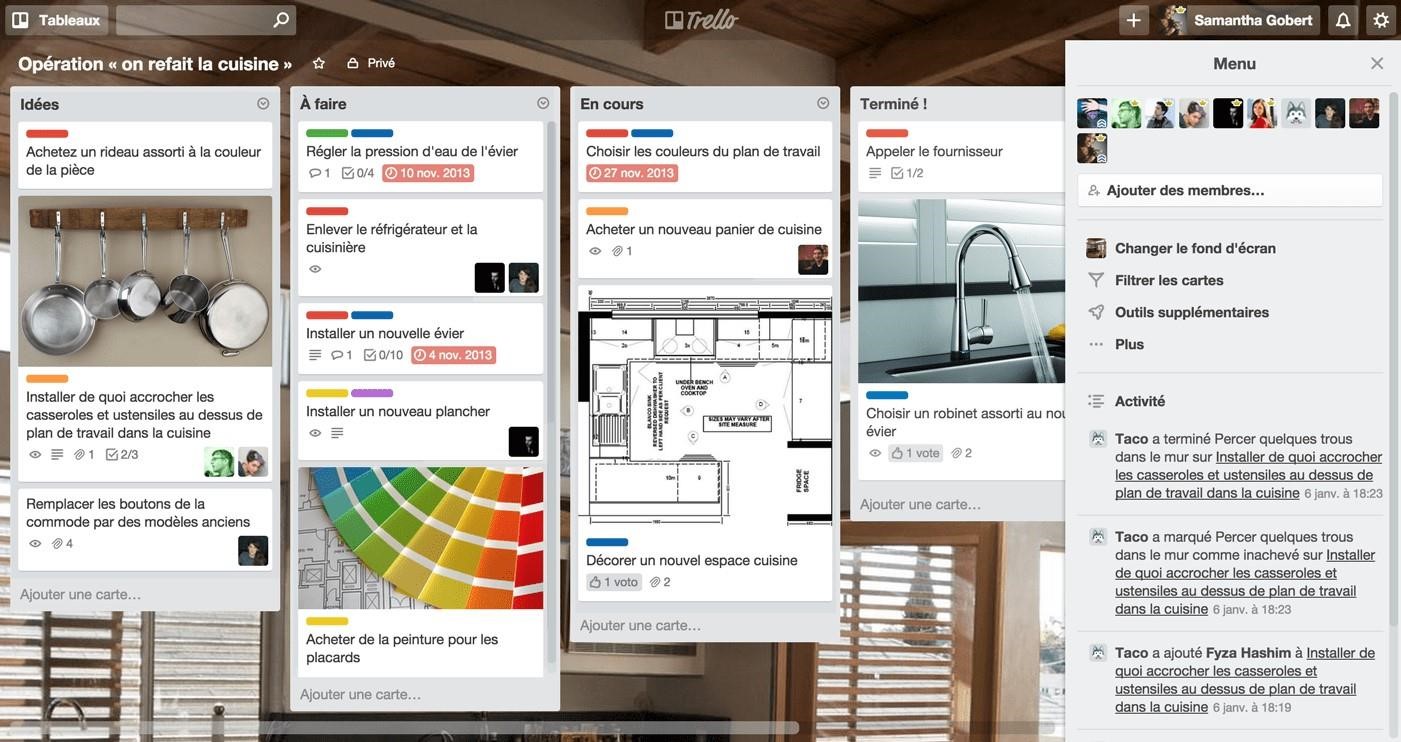


Figure 4 Tableau de bord de Trello

* JIRA :

Jira Software est conçu de sorte que chaque membre de votre équipe de développement puisse planifier, suivre et livrer d’excellents logiciels. Ce logiciel permet de planifier et de suivre le bon déroulement du projet.

La figure 5 présente l’espace de travail de JIRA.

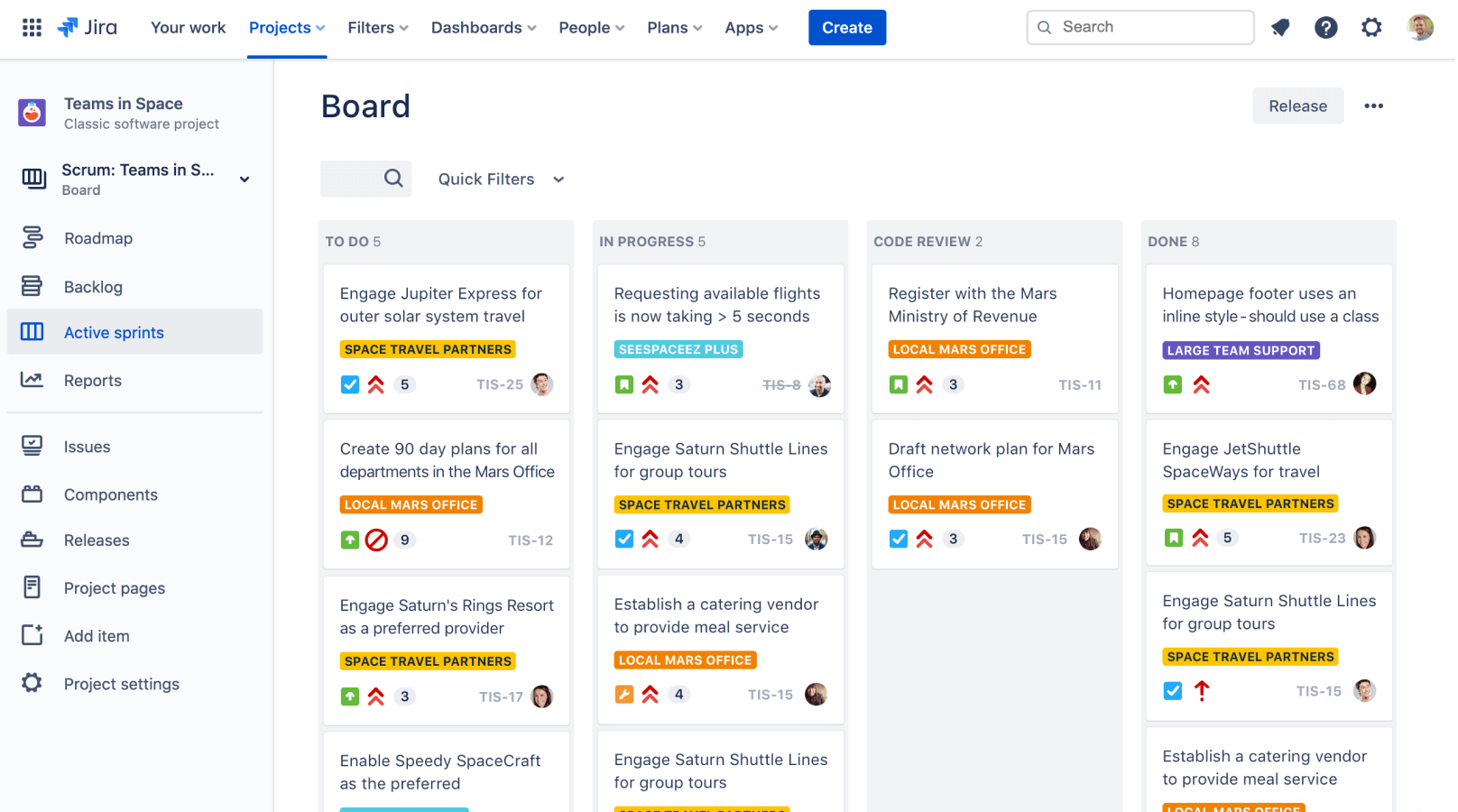


Figure 5 Espace de travail de JIRA

En fin du compte, après avoir considéré tous ces détails, nous avons choisi de combiner les 2 outils.

##### Méthode de gestion de projet

* Présentation de la méthode AGILE

Les méthodes agiles sont des méthodologies essentiellement dédiées à la gestion de projets informatiques. Elles reposent sur des cycles de développement itératifs et adaptatifs en fonction des besoins évolutifs du client. Elles permettent notamment d'impliquer l'ensemble des collaborateurs ainsi que le client dans le développement du projet [26].

Ces méthodes permettent généralement de mieux répondre aux attentes du client en un temps limité. Ces méthodes constituent donc un gain en productivité ainsi qu'un avantage compétitif tant du côté client que du côté du fournisseur.

Les méthodes agiles se reconnaissent toutes dans les valeurs suivantes.

* **L'équipe et la communication avant les outils et processus** : dans la vision agile, l'équipe est bien plus importante que les outils ou les procédures de fonctionnement. Il est préférable d'avoir une équipe soudée et dont les membres communiquent entre eux, composée de développeurs de niveaux différents, plutôt qu'une équipe composée d'experts qui travaillent de manière isolée. La communication est donc une notion fondamentale dans un contexte de développement agile.

* **L'application avant la documentation** : il est primordial que le projet fonctionne, c'est la priorité avant toute chose. La documentation technique et les autres outils (de tests, de reporting) constituent une aide précieuse, mais ne sont pas une fin en soi. Une documentation précise est utile comme moyen de communication. Il est parfois préférable de simplement commenter abondamment le code lui-même, et surtout de transférer la totalité des compétences et connaissances du métier à l'ensemble des collaborateurs de l'équipe.

* **La collaboration avant la négociation** : le client doit être impliqué dans le développement. Le fournisseur ne doit pas se contenter de négocier un contrat au début du projet, puis de refuser l'évolution des besoins du client. Le client doit collaborer avec l'équipe et fournir des comptes rendus réguliers sur l'adaptation du logiciel à ses attentes.

* **L'acceptation du changement et la flexibilité avant la planification** : la planification initiale et la structure du projet doivent être flexibles afin de permettre les évolutions attendues par le client. En effet, les premières livraisons du projet donnent très souvent suite à des demandes d'évolution.

La raison pour laquelle, on a préféré le manifeste agile qui met en premier plan la collaboration avec le client. Il consiste également à raccourcir les cycles de développement et à se focaliser sur le produit final.

Comparé avec les méthodes traditionnelles, le manifeste agile se marque par les quatre (4) points dans la figure 6 qui résume le manifeste agile.

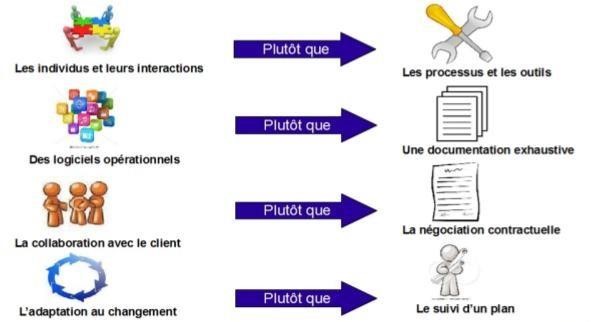


Figure 6 Manifeste agile

* La méthode scrum

La méthode Scrum est un cadre méthodologique de gestion de projet en Agile. Elle vise à accélérer les travaux et répondre plus vite et efficacement aux besoins du client

.elle est marquée par une succession d'étapes représentée par la figure 7. [27]

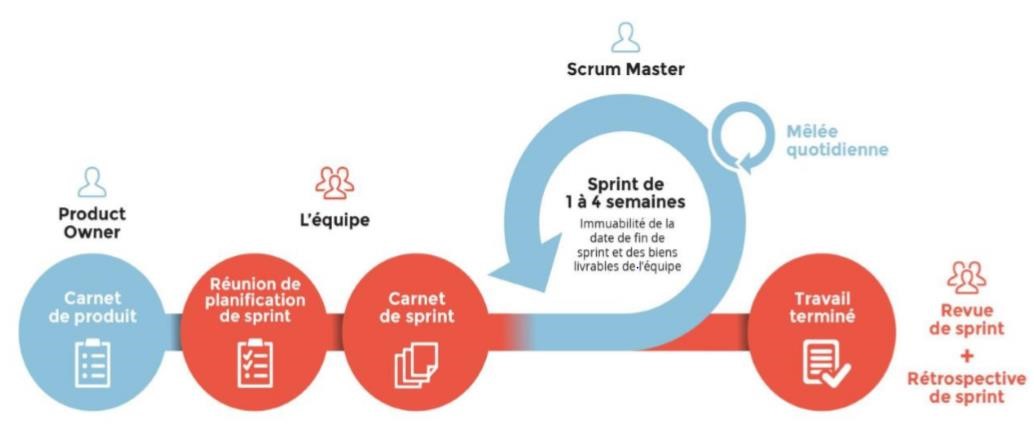


Figure 7 Étapes de SCRUM

D’après la figure, SCRUM est une méthode qui s’appuie sur le découpage des projets en itérations encore nommées « sprints ». Un sprint peut avoir une durée qui varie généralement entre une à quatre semaines. Avant chaque sprint, les tâches sont estimées en temps et en complexité à l'aide de certaines pratiques. Ces estimations permettent à la fois de planifier les livraisons, mais aussi d'estimer le cout de ces tâches auprès du client. Les fonctionnalités (encore appelées « user stories ») qui font l'objet d'un sprint constituent ce que l'on appelle un « sprint backlog » du produit éventuellement livrable à la fin du sprint.

Il est nécessaire de distinguer le sprint backlog du « product backlog » qui lui correspond à l'ensemble des fonctionnalités attendues pour le produit sur l'ensemble des sprints.

La méthode Scrum est aussi caractérisée par une « mêlée » quotidienne, dans laquelle les collaborateurs (chefs de projets, développeurs et responsables fonctionnels) indiquent tour à tour les tâches qu'ils ont effectuées la veille, les difficultés rencontrées et enfin ce sur quoi ils vont poursuivre leur travail le jour suivant. Cela permet d'évaluer l'avancement du projet, de mobiliser des ressources là où cela est le plus nécessaire, mais aussi de venir en aide aux collaborateurs rencontrant des difficultés lorsque celles-ci ont déjà été rencontrées auparavant par d'autres membres de l'équipe.

 **Rôles**

L’équipe Scrum est répartie en 3 rôles :

* **Le Product Owner** : c’est le responsable du produit, il représente les clients et les utilisateurs en transcrivant leurs besoins, définit et priorise les demandes produit.

* **Le Scrum Master**, qui n’est pas le chef de projet, mais il a pour charge de faciliter l’application de Scrum, sa mission est de tout mettre en œuvre pour que l’équipe travaille dans de bonnes conditions et se concentre sur l’objectif du projet. Il porte également une attention particulière au respect des différentes phases de Scrum.

* **L’équipe Scrum** : L’équipe se gère en toute autonomie et est en charge du développement du produit. Il n’y a pas de notion de hiérarchie, toutes les décisions sont prises ensemble. Elle regroupe les rôles habituellement nécessaires à un projet (architecte, concepteur, développeur, etc).

**Réunions ou meeting**

Les cérémonies Scrum sont donc:

* « **Sprint planning** » une fois par sprint : les équipes sélectionnent les tâches présentes dans le product backlog réalisable en un seul sprint (création du « sprint backlog »).

* « **Daily Scrum** » à chaque début de journée : chaque membre de l’équipe décrit ce qu’il a fait la veille, ce qu’il fera durant la journée et enfin si il a rencontré des problèmes.

* « **Sprint review** » une fois par sprint terminé : L’équipe présente ce qui a été accompli lors du sprint.

* « **Sprint retrospective** » une fois par sprint terminé : prise de recul sur ce qui a fonctionné et ce qui n’a pas bien fonctionné, afin d’optimiser les prochains sprints.

## Analyse conceptuelle

Après le recueil des besoins des utilisateurs et les différentes propriétés de l’application, on va entamer à une phase importante et indispensable dans le cycle de vie d’une application. C’est la phase de conception, elle a pour but d’expliquer toutes les étapes de l’élaboration et le déroulement de l’application.

### Étapes de l’élaboration du backlog de produit

Il y a cinq étapes qui faciliteront d’aboutir à l’élaboration du backlog de produit et qui permettent de répondre aux trois (03) questions suivantes : A qui sont-elles destinées ? Quelles seront les fonctionnalités à créer ? Dans quel ordre devraient-elles être livrées ?

Ces étapes sont :

Etape 1 : Vision globale du projet

Etape 2 : Liste détaillée des acteurs

Etape 3 : Thème ou regroupement , s de fonctionnalités

Etape 4 : Vision du premier release

Etape 5 : User stories

#### Vision globale du projet

« Offrir une application de gestion de la relation humaine par communication non verbale »

#### Liste détaillée des acteurs avec fiche

On entend par acteur un humain, une machine ou un système qui ne fait pas partie de la solution à réaliser, mais qui participe au fonctionnement général de la solution par une interaction.

Il est nécessaire d’identifier puis détailler tous les intervenants du système dans tous ses aspects. On peut citer, dans notre cas :

* Administrateur
* Architecte
* Modérateur
* Utilisateur

|  |
| --- |
| **Rôle** : Administrateur |
| **Description** : Gérer l'accès des utilisateurs à la plateforme  Voir détails profil  Sélectionner utilisateur  Supprimer utilisateur  Annuler suppression utilisateur  Valider suppression  Accéder à un profil  Gérer condition d’accès  Mettre en attente l’accès d'un utilisateur  Refuser l’accès d’un utilisateur  Changer le rôle d’un utilisateur  Changer de rôle par sélection multiple  Recherche utilisateur  Filtrer des utilisateurs  Ouvrir pièce jointe  Consulter historique  Valider pièce jointe  **Satisfaction** : Facilite la gestion des utilisateurs de l’application  **Instance** : 1 ou plusieurs  **Fréquence d’utilisation** : Journalière  **Niveau de connaissance technique** : Bon  **Niveau de connaissance métier** : Bon |

|  |
| --- |
| **Role** : Utilisateur |
| **Description**: S’inscrire  Modifier le profil  Créer un nouveau projet  Copier lien vidéo  Lire une vidéo téléchargée  Ralentir la vitesse de lecture  Mettre en pause la vidéo  Lire image par image la vidéo  Référencer un sujet  Créer une analyse  Pointer un sujet sur une image  Aller au profil du créateur d'item  Codifier (écrire code)  Éditer des codifications faites  Rapport item secondaire  Télécharger rapport item secondaire  Naviguer sur la galerie  Consulter analyses récentes  Trier liste dans la galerie  Filtre les analyses  Rechercher des analyses  Voir les détails de la vidéo  Gérer les détails de la vidéo  Naviguer dans les analyses d'une vidéo  Copier lien vers la page d'analyse  Éditer analyse vidéo  **Satisfaction** : Facilite la gestion des utilisateurs de l’application  **Instance** : 1 ou plusieurs  **Fréquence d’utilisation** : Journalière  **Niveau de connaissance technique** : Bon  **Niveau de connaissance métier** : Bon |

|  |
| --- |
| **Rôle** : Moderateur |
| **Description** : approuver une analyse  **Satisfaction** : S’occupe de l’approbation des analyses faites dans la plateforme  **Instance** : 1 ou plusieurs  **Fréquence d’utilisation** : Journalière  **Niveau de connaissance technique** : Bon  **Niveau de connaissance métier** : Bon |

|  |
| --- |
| **Rôle** : Architecte |
| **Description** : ajouter des items secondaires  naviguer dans les items secondaires  ajouter marqueurs d’item secondaire  **Satisfaction** : S’occupe de l’approbation des analyses faites dans la plateforme  **Instance** : 1 ou plusieurs  **Fréquence d’utilisation** : Journalière  **Niveau de connaissance technique** : Bon  **Niveau de connaissance métier** : Bon |

#### Thème et regroupement de fonctionnalités

La collision des thèmes forme la base spécifiant l’objectif central du produit. C’est de la vision du produit qu’on extrait les thèmes principaux.

Notre vision est d’offrir une plateforme complète de mise en relation des candidats à la recherche d’emploi et des recruteurs proposant des offres. Nous allons donc énoncer ci-après les fonctionnalités classées selon une notion de couleur afin de prioriser les tâches à faire.

Notion de couleur :

|  |
| --- |
|  |

* Rouge : Très important

|  |
| --- |
|  |

* Jaune : Important

|  |
| --- |
|  |

* Vert : Utile

Fonctionnalités :

* **Voir le détail profil**
  + *Description* : voir la page de détails profil
  + *Valeur commerciale* : élevée
  + *Critères de satisfaction* : Compte utilisateurs activé.
  + *Couleur de postit* : Rouge
  + *Utilisateurs* : Administrateurs
* **Sélectionner d’un utilisateur**
  + Description : édition d’un compte utilisateur
  + Valeur commerciale : élevée
  + Critère de satisfaction : compte utilisateur mise à jour
  + Couleur de postit : Rouge
  + Utilisateur : Administrateur
* **Suppression d’un utilisateur**
  + Description : suppression d’un utilisateur dans la plateforme
  + Valeur commerciale : élevée
  + Critère de satisfaction : utilisateur supprimé
  + Couleur de postit : Rouge
  + Utilisateur : Administrateur
* **Annuler suppression d’utilisateur**
  + *Description* : annuler la suppression pendant une demande de confirmation de supprimer
  + *Valeur commerciale* : élevée
  + *Critères de satisfaction* : la suppression est annulée
  + *Couleur de postit* : Rouge
  + *Utilisateurs* : Administrateurs
* **Valider suppression**
  + *Description* : valider la suppression pendant une demande de confirmation de supprimer
  + *Valeur commerciale* : élevée
  + *Critères de satisfaction* : la suppression avec succès
  + *Couleur de postit* : Rouge
  + *Utilisateurs* : Administrateurs
* **Accéder à un profil** 
  + *Description* : accéder au profil de l’utilisateur sélectionné
  + *Valeur commerciale* : élevée
  + *Critères de satisfaction* : obtention d’un visuel pour voir le résumé d’un compte utilisateur
  + *Couleur de postit* : Rouge
  + *Utilisateurs* : Administrateurs
* **Gérer la condition d’accès** 
  + *Description* : valider le document d’un utilisateur afin de l’accepter dans la plateforme
  + *Valeur commerciale* : élevée
  + *Critères de satisfaction* : Le document d’un utilisateur a comme stratus validité
  + *Couleur de postit* : Rouge
  + *Utilisateurs* : Administrateurs
* **Mettre en attente l’accès d’un utilisateur**
  + *Description* : mettre en attente l’accès d’un utilisateur à la plateforme
  + *Valeur commerciale* : élevée
  + *Critères de satisfaction* : L’accès de l’utilisateur à la plateforme est en attente
  + *Couleur de postit* : Rouge
  + *Utilisateurs* : Administrateurs
* **Refuser l’accès d’un utilisateur aux plateformes**
  + *Description* : refuser l’accès de l’utilisateur aux plateformes
  + *Valeur commerciale* : élevée
  + *Critères de satisfaction* : L’accès de l’utilisateur à la plateforme est refusé
  + *Couleur de postit* : Rouge
  + *Utilisateurs* : Administrateurs
* **Changer le rôle d’un utilisateur**
  + *Description* : Donner un rôle supérieur à l’utilisateur
  + *Valeur commerciale* : élevée
  + *Critères de satisfaction* : L’utilisateur obtient un nouveau rôle
  + *Couleur de postit* : Rouge
  + *Utilisateurs* : Administrateurs
* **Rechercher un utilisateur**
  + *Description* : Chercher un utilisateur à partir d’un mot-clé
  + *Valeur commerciale* : élevée
  + *Critères de satisfaction* : obtenir le résultat de la recherche
  + *Couleur de postit* : Rouge
  + *Utilisateurs* : Administrateurs
* **Filtrer la liste d’utilisateur** 
  + *Description* : Chercher un utilisateur à partir de plusieurs filtre en même temps
  + *Valeur commerciale* : élevée
  + *Critères de satisfaction* : obtenir le résultat de la recherche
  + *Couleur de postit* : Rouge
  + *Utilisateurs* : Administrateurs
* **Ouvrir la pièce jointe**
  + *Description* : ouvrir la pièce jointe pendant l’inscription
  + *Valeur commerciale* : élevée
  + *Critères de satisfaction* : la pièce jointe est lisible dans le navigateur
  + *Couleur de postit* : Rouge
  + *Utilisateurs* : Administrateurs
* **Valider la pièce jointe**
  + *Description* : valider la pièce jointe de l’utilisateur après l’avoir visualisé
  + *Valeur commerciale* : élevée
  + *Critères de satisfaction* : la pièce jointe est validée
  + *Couleur de postit* : Rouge
  + *Utilisateurs* : Administrateurs
* **Modifier information profil**
  + *Description* : changer les informations personnel ainsi que la photo de profil
  + *Valeur commerciale* : élevée
  + *Critères de satisfaction* : le profil est mis à jour
  + *Couleur de postit* : Rouge
  + *Utilisateurs* : Utilisateur
* **Uploader vidéo**
  + *Description* : téléchargement de la vidéo YouTube dans la plateforme
  + *Valeur commerciale* : élevée
  + *Critères de satisfaction* : la vidéo est uploadée dans le système
  + *Couleur de postit* : Rouge
  + *Utilisateurs* : Utilisateur
* **Uploader vidéo**
  + *Description* : téléchargement de la vidéo YouTube dans la plateforme
  + *Valeur commerciale* : élevée
  + *Critères de satisfaction* : la vidéo est uploadée dans le système
  + *Couleur de postit* : Rouge
  + *Utilisateurs* : Utilisateur
* **Lire la vidéo téléchargée**
  + *Description* : lecture de la vidéo uploadée dans le système
  + *Valeur commerciale* : élevée
  + *Critères de satisfaction* : la vidéo se met à être lue
  + *Couleur de postit* : Rouge
  + *Utilisateurs* : Utilisateur
* **Ralentir la vitesse** 
  + *Description* : lecture de la vidéo uploadée dans le système en mode ralenti c’est-à-dire avec une vitesse réduite
  + *Valeur commerciale* : élevée
  + *Critères de satisfaction* : la vidéo est lue avec une vitesse réduite
  + *Couleur de postit* : Jaune
  + *Utilisateurs* : Utilisateur
* **Mettre en pause la vidéo**
  + *Description* : mise en pause de la lecture de la vidéo
  + *Valeur commerciale* : élevée
  + *Critères de satisfaction* : la vidéo se met en pause
  + *Couleur de postit* : Jaune
  + *Utilisateurs* : Utilisateur
* **Lire image par image**
  + *Description* : lire la vidéo image par image
  + *Valeur commerciale* : élevée
  + *Critères de satisfaction* : la vidéo est lue image par image selon le nombre d’images par image de la vidéo
  + *Couleur de postit* : Jaune
  + *Utilisateurs* : Utilisateur
* **Référencer un sujet**
  + *Description* : énumérer les personnages dans une vidéo à analyser
  + *Valeur commerciale* : élevée
  + *Critères de satisfaction* : la vidéo a sa liste de personnages
  + *Couleur de postit* : Rouge
  + *Utilisateurs* : Utilisateur
* **Créer une analyse**
  + *Description* : analyser un vidéo en mettant en pause la vidéo sur une image et décrire en suite le geste du sujet à analyser qui a été référencé
  + *Valeur commerciale* : élevée
  + *Critères de satisfaction* : une analyse a été créer
  + *Couleur de postit* : Rouge
  + *Utilisateurs* : Utilisateur
* **Pointer un sujet sur une image**
  + *Description* : ajouter un pointeur sur une image afin de désigner le sujet à analyser
  + *Valeur commerciale* : élevée
  + *Critères de satisfaction* : un pointeur est sur une image
  + *Couleur de postit* : Rouge
  + *Utilisateurs* : Utilisateur
* **Aller au profil du créateur de l’analyse**
  + *Description* : Aller vers le profil du créateur de l’analyse
  + *Valeur commerciale* : élevée
  + *Critères de satisfaction* : page de profil de l’utilisateur créateur de l’analyse
  + *Couleur de postit* : Rouge
  + *Utilisateurs* : Utilisateur
* **Codifier**
  + *Description* : Saisir un code à partir d’un arbre qui contient la base de donnée à utiliser
  + *Valeur commerciale* : élevée
  + *Critères de satisfaction* : obtention d’un code avec les significations
  + *Couleur de postit* : Rouge
  + *Utilisateurs* : Utilisateur
* **Editer des codifications faites**
  + *Description* : Modifier la codification
  + *Valeur commerciale* : élevée
  + *Critères de satisfaction* : la codification est modifiée
  + *Couleur de postit* : Rouge
  + *Utilisateurs* : Utilisateur
* **Rapport item secondaire**
  + *Description* : Obtenir les rapports d’item secondaire sur une vidéo
  + *Valeur commerciale* : élevée
  + *Critères de satisfaction* : obtenir un fichier Word contenant les items secondaires
  + *Couleur de postit* : Jaune
  + *Utilisateurs* : Utilisateur
* **Télécharger le rapport item secondaire**
  + *Description* : Obtenir le fichier contenant le rapport d’item secondaire sur une vidéo
  + *Valeur commerciale* : élevée
  + *Critères de satisfaction* : télécharger le fichier Word contenant les items secondaires
  + *Couleur de postit* : Jaune
  + *Utilisateurs* : Utilisateur
* **Naviguer sur la galerie**
  + *Description* : aller sur la page contenant la galerie des vidéos
  + *Valeur commerciale* : élevée
  + *Critères de satisfaction* : liste des vidéos
  + *Couleur de postit* : Rouge
  + *Utilisateurs* : Utilisateur
* **Consulter les analyses récentes**
  + *Description* : lister les 10 dernière analyse faite dans la plateforme
  + *Valeur commerciale* : élevée
  + *Critères de satisfaction* : liste des 10 dernières analyses faites dans la plateforme
  + *Couleur de postit* : Jaune
  + *Utilisateurs* : Utilisateur
* **Trier la galerie des vidéos**
  + *Description* : trier la liste dans la galerie des vidéos
  + *Valeur commerciale* : élevée
  + *Critères de satisfaction* : la galerie est triée selon le critère de trie choisi et l’ordre de tri
  + *Couleur de postit* : Jaune
  + *Utilisateurs* : Utilisateur
* **Filtrer les analyses**
  + *Description* : filtrer les analyses
  + *Valeur commerciale* : élevée
  + *Critères de satisfaction* : résultat du filtre
  + *Couleur de postit* : Jaune
  + *Utilisateurs* : Utilisateur
* **Recherche d’analyse**
  + *Description* : taper une codification pour rechercher les analyses correspondantes
  + *Valeur commerciale* : élevée
  + *Critères de satisfaction* : résultat de la recherche
  + *Couleur de postit* : Jaune
  + *Utilisateurs* : Utilisateur
* **Détails de la vidéo**
  + *Description* : voir les détails de la vidéo dans la page analyse
  + *Valeur commerciale* : élevée
  + *Critères de satisfaction* : affichage de la page détails vidéo
  + *Couleur de postit* : Jaune
  + *Utilisateurs* : Utilisateur
* **Modifier les détails d’une vidéo**
  + *Description* : modifier les informations d’une vidéo
  + *Valeur commerciale* : élevée
  + *Critères de satisfaction* : les informations de la vidéo sont changées
  + *Couleur de postit* : Jaune
  + *Utilisateurs* : Utilisateur
* **Copier lien vers la page d'analyse**
  + *Description* : copier un lien vers la page d’analyse
  + *Valeur commerciale* : élevée
  + *Critères de satisfaction* : le lien est copié dans la presse papier
  + *Couleur de postit* : Jaune
  + *Utilisateurs* : Utilisateur
* **Approuver une analyse**
  + *Description* : approuver une analyse faite par un utilisateur
  + *Valeur commerciale* : élevée
  + *Critères de satisfaction* : le lien est copié dans la presse papier
  + *Couleur de postit* : Jaune
  + *Utilisateurs* : Modérateur
* **Ajouter des items secondaires**
  + *Description* : ajouter un item secondaire dans une analyse
  + *Valeur commerciale* : élevée
  + *Critères de satisfaction* : un item secondaire lié à une analyse
  + *Couleur de postit* : Jaune
  + *Utilisateurs* : Architecte
* **Ajouter un marqueur d’item secondaire**
  + *Description* : ajouter un marqueur d’item secondaire
  + *Valeur commerciale* : élevée
  + *Critères de satisfaction* : un marqueur est ajouté dans le timeline
  + *Couleur de postit* : Jaune
  + *Utilisateurs* : Architecte

#### Vision du premier release

La vision du premier release est « Assurer l’analyse vidéo ».

#### User stories

Une user story est un point de vue utilisateur guidé par les réponses aux trois (03) questions suivantes : Qui a fait la demande ou qui bénéficie de la demande? Quelle est la demande? Quelle valeur métier découle de la réalisation de ce besoin [6]?

Voici donc la présentation de la vision des utilisateurs basée autour de trois (03) axes (le rôle, besoin et valeur métier) :

**User story 1** : En tant qu’utilisateur déjà inscrit je veux me connecter afin d’accéder à la plateforme.

**User story 2** : En tant qu’utilisateur déjà inscrit je veux récupérer un mot de passe afin d’accéder à la plateforme.

**User story 3** : En tant que nouvel utilisateur je veux m’inscrire afin d’accéder à la plateforme

**User story 4** : En tant qu’utilisateur non inscrit je veux remplir mon profil afin d’accéder à la plateforme

**User story 5** : En tant qu’administrateur je veux procéder à des actions afin de gérer l’accès des utilisateurs à la plateforme

**User story 6** : En tant qu’administrateur je veux gérer les conditions d’accès afin de gérer les utilisateurs de la plateforme

**User story 7** : En tant qu’administrateur je veux accéder naviguer dans la liste afin de chercher des utilisateurs

**User story 8** : En tant qu’administrateur je veux procéder à des actions et obtenir des informations afin de gérer le profil d’un utilisateur

**User story 9** : En tant qu’utilisateur je veux intégrer une vidéo afin de commencer une analyse

**User story 10** : En tant qu’utilisateur je veux regarder une vidéo afin d’étudier les items

**User story 11** : En tant qu’utilisateur je veux obtenir des interactions afin d’analyser une vidéo

**User story 12** : En tant qu’utilisateur je veux procéder à une codification afin d’analyser une vidéo

**User story 13** : En tant qu’utilisateur je veux accéder au rapport afin de naviguer dans la vidéo

**User story 14** : En tant qu’utilisateur je veux modifier des informations relatives à l’ensemble d’une vidéo afin de les gérer au travers tous les items

**User story 15** : En tant que modérateur je veux accéder aux dernières activités sur la plateforme afin de les gérer de façon fluide

**User story 16** : En tant qu’utilisateur je veux naviguer dans la galerie afin de découvrir des analyses

**User story 17** : En tant qu’utilisateur je veux filtrer les analyses afin de trouver des analyses

**User story 18** : En tant qu’utilisateur je veux voir le détail d’une vidéo afin de découvrir les analyses associées

**User story 19** : En tant qu’architecte je veux voir naviguer parmi les items secondaires afin de voir les liens entre ceux-ci

#### Backlog de produit

Le tableau 15 illustre le backlog du produit

Tableau 17 Backlog du produit

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Backlog de produit – JIRA | | | | |
| User story | Titre | Importance | Estimation (jours) | Commentaire |
| 1 | Connexion | 1 | 2 | Design + Back + front |
| 2 | Récupération mot de passe | 2 | 1 | Design + Back + front |
| 3 | Inscription | 4 | 3 | Design + Back + front |
| 4 | Formulaire | 3 | 2 | Design + Back + front |
| 5 | Gestion d’accès | 6 | 5 | Design + Back + front |
| 6 | Condition d’accès | 7 | 4 | Design + Back + front |
| 7 | Recherche | 8 | 5 | Design + Back + front |
| 8 | Gestion profil utilisateur | 5 | 10 | Design + Back + front |
| 9 | Upload d’une vidéo | 9 | 5 | Design + Back + front |
| 10 | Lecture vidéo | 10 | 5 | Design + front |
| 11 | Analyse une vidéo | 13 | 7 | Design + Back + front |
| 12 | Codification | 11 | 5 | Design + Back + front |
| 13 | Rapport d’activité | 18 | 4 | Design + Back + front |
| 14 | Gestion information vidéo | 12 | 5 | Design + Back + front |
| 15 | Dernière activité | 16 | 3 | Design + Back + front |
| 16 | Galerie de vidéo | 14 | 6 | Design + Back + front |
| 17 | Filtre analyse | 19 | 3 | Design + Back + front |
| 18 | Détails vidéos | 15 | 4 | Design + Back + front |
| 19 | Item secondaire | 17 | 6 | Design + Back + front |

### Dictionnaire des données

Un dictionnaire des données est une collection de métadonnées ou des de données de référence nécessaire à la conception d’une base de données relationnelle. Il revêt une importance stratégique particulière, car il est le vocabulaire commun de l’organisation [15].

Il doit respecter les contraintes suivantes :

* Tous les noms doivent être mono values et non décomposables ;
* Il ne doit pas y avoir d’homonyme ni de synonyme ;
* Il ne doit pas y avoir d’homonyme ni de synonyme ;
* Les données y sont regroupées par entité ;
* Les identifiants sont complètement précisés ;
* Les commentaires doivent être pertinents.

Le but de cette partie est d’établir une spécification des besoins fonctionnels qui découlent de l’étude de l’existant précédent. Le tableau 18 présente le dictionnaire des données.

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| Nom | Description | Domaine de valeur | | Règle de gestion | format |
| Type | Taille |  |  |
| Author | Auteur de la vidéo | AN | 30 |  |  |
| Birthday | Date d’anniversaire de l’utilisateur | date | 10 |  | Yyyy-mm-dd |
| captureContentUrl | URL du capture | AN | 255 |  |  |
| captureFile | Fichier capture | AN | 255 |  |  |
| captureFilePath | Chemin vers le fichier capture | AN | 255 |  |  |
| Code | Code issue du carte mentale | AN | 255 |  |  |
| codeItemSecondaire | Code issue du carte mentale | AN | 40 |  |  |
| Color | Couleur de l’utilisateur | AN | 10 |  |  |
| confirmationToken | Token de confirmation | AN | 255 |  |  |
| descriptionAnalyse | Description de l’analyse | AN | 255 |  |  |
| descriptionCodification | Description du code dans la carte mentale | AN | 100 |  |  |
| descriptionItemSecondaire | Description de l’item secondaire | AN | 100 |  |  |
| Document | Piece jointe pendant l’inscription | AN | 255 |  |  |
| Email | Adresse email de l’utilisateur | AN | 100 |  |  |
| firstName | Prénom de l’utilisateur | AN | 50 |  |  |
| formationCenterName | Centre de formation choisi par l’utilisateur | AN | 20 |  |  |
| Image | Image de profil de l’utilisateur | AN | 255 |  |  |
| isMailVerified | Etat de verification mail | booleen | 1 |  |  |
| isPaid | Etat de payment | boolean | 1 |  |  |
| Keywords | Mot clé YouTube | AN | 50 |  |  |
| nameSujet | Nom du sujet | AN | 50 |  |  |
| nameCodification | Nom du codification | AN | 50 |  |  |
| Observation | Observation durant l’analyse | TEXTE |  |  |  |
| Password | Mot de passe de l’utilisateur | AN | 255 |  |  |
| pathYoutubeVideo | Chemin du vidéo dans le système | AN | 255 |  |  |
| phoneNumber | Numéro téléphone de l’utilisateur | AN | 15 |  |  |
| pointeurAnalyse | Coordonnée où il faut mettre un pointeur dans la capture pour l’analyse vidéo | AN | 8 |  |  |
| pointeurItemSecondaire | Coordonnée où il faut mettre un pointeur dans un item secondaire | AN | 8 |  |  |
| pointeurReferencement | Coordonnée qui référencie un sujet dans une capture | AN | 8 |  |  |
| Role | Rôle d’un utilisateur | AN | 30 |  |  |
| roleName | Nom du role d’utilisateur | AN | 30 |  |  |
| rolePseudo | Pseudo d’un role d’utilisateur | AN | 20 |  |  |
| shortDescription | Courte description d’une vidéo YouTube | AN | 255 |  |  |
| Society | Société | AN | 20 |  |  |
| statusAnalyse | Status d’une analyse vidéo | AN | 10 |  |  |
| statusUser | Status d’un utilisateur | AN | 10 |  |  |
| sujetName | Nom d’un sujet à analyser | AN | 10 |  |  |
| timeCapture | Timing d’un capture à partir d’un vidéo | AN | 8 |  | HH :MM :SS |
| Title | Titre de la vidéo | AN | 100 |  |  |
| typeEmetteur | Type d’emmeteur à analyser dans une vidéo | AN | 10 |  |  |
| url | url de la vidéo YouTube | AN | 255 |  |  |
| userCreatedAt | Date de creation d’utilisateur | date |  |  |  |
| userUpdatedAt | Date de mise à jour d’un utilisateur | date |  |  |  |
| verbalePrononce | Verbale prononcé pendant une analyse | TEXTE |  |  |  |
| videoID | Id video youtube | AN | 20 |  |  |
| viewCount | Nombre de vue d’une vidéo | Nombre | 3 |  |  |

### Règle de gestion

Une règle de gestion est une règle suivie par une société. Il peut s’agir d’une disposition légale, d’une exigence formulée par un client ou d’un article et règlement interne. Une règle de gestion est la traduction conceptuelle des objectifs choisis et des contraintes acceptées par l’entreprise. Elle est plus particulièrement liée aux traitements (règle d’action) ou aux données (règle de calcul) :

- Une règle d’action décrira les actions que doit accomplir l’entreprise.

- Les règles de calcul décrivent la façon dont doivent d’accomplir les actions.

**RG1** : Dans un centre de formation se trouvent plusieurs utilisateurs

**RG2** : Un utilisateur se trouve dans un centre de formation

**RG3** : Un utilisateur a plusieurs rôles

**RG4** : un rôle peut être associé à plusieurs utilisateurs

**RG5** : Un utilisateur effectue un ou plusieurs analyses

**RG6** : Une analyse est faite par un et un seul utilisateur

**RG7** : Un sujet est référencé sur une capture à l’aide d’un pointeur

**RG8** : Une analyse appartient à un et une seule capture

**RG9** : Une capture peut avoir plusieurs analyses

**RG10** : Une analyse contient plusieurs items secondaires

**RG11** : Un item secondaire appartient à une analyse

**RG12** : Une analyse ne peut avoir qu’un seul sujet analysé

**RG13** : Un sujet analysé peut être dans plusieurs analyses

**RG14** : Une vidéo a un ou plusieurs sujets à analyser

**RG15** : Un sujet analysé se trouve que dans un seul vidéo

**RG16** : Un vidéo a plusieurs captures

**RG17** : Une capture n’appartient qu’un et une seule vidéo

### Capture des besoins fonctionnels

Pour répondre aux besoins des utilisateurs et pour réaliser une application de qualité, nous avons retenu tous les besoins des utilisateurs à travers les spécifications des besoins fonctionnels et techniques. Les besoins fonctionnels ou besoins métiers représentent les actions que le système doit exécuter, il ne devient opérationnel que s’il les satisfait. Après une étude détaillée de système, cette partie est réservée à la description des exigences fonctionnelles des différents acteurs de l’application. Ces besoins se regroupent dans les diagrammes des cas d’utilisation. Cette application doit couvrir principalement les besoins fonctionnels suivants :

* La gestion des utilisateurs
* L’approbation d’une analyse
* La gestion d’une codification
* La gestion des items secondaires
* L’upload vidéo
* La gestion de lecture d’une vidéo
* La galerie des vidéos
* Gestion des analyses

### Analyse

#### Diagramme de cas d’utilisation

Le diagramme de cas d’utilisation a pour but de donner une vision globale sur les interfaces du futur de l’application. C’est le premier diagramme UML constitué d’un ensemble d’acteurs qui agit sur des cas d’utilisation et qui décrit sous la forme d’actions et des réactions, le comportement du système du point de vue utilisateur.

**Acteur** : un acteur est un utilisateur qui communique et interagit avec les cas d’utilisation du système. C’est une unité ayant un comportement comme une personne, le système ou une entreprise.

**Système** : cet élément fixe les limites du système en relation avec l’acteur qui utilisent (en dehors du système) et les fonctions qu’il doit fournir (à l’intérieur du système).

**Cas d’utilisation** : un cas d’utilisation représente un ensemble de séquence d’actions à réaliser par le système et produisant un résultat observable intéressant pour un acteur particulier représenter par des ellipses et limité par un rectangle pour représenter le système.

Identification des acteurs

**Administrateur** : c’est la personne qui gère les utilisateurs.

**Architecte** : c’est la personne qui s’occupe de la gestion de l’arbre de la classification mentale

**Modérateur** : c’est la personne qui s’occupe de l’approbation des analyses effectuée

**Utilisateur** : ce sont les personnes qui font l’analyse dans la plateforme

La figure 8 montre le diagramme de cas d’utilisation

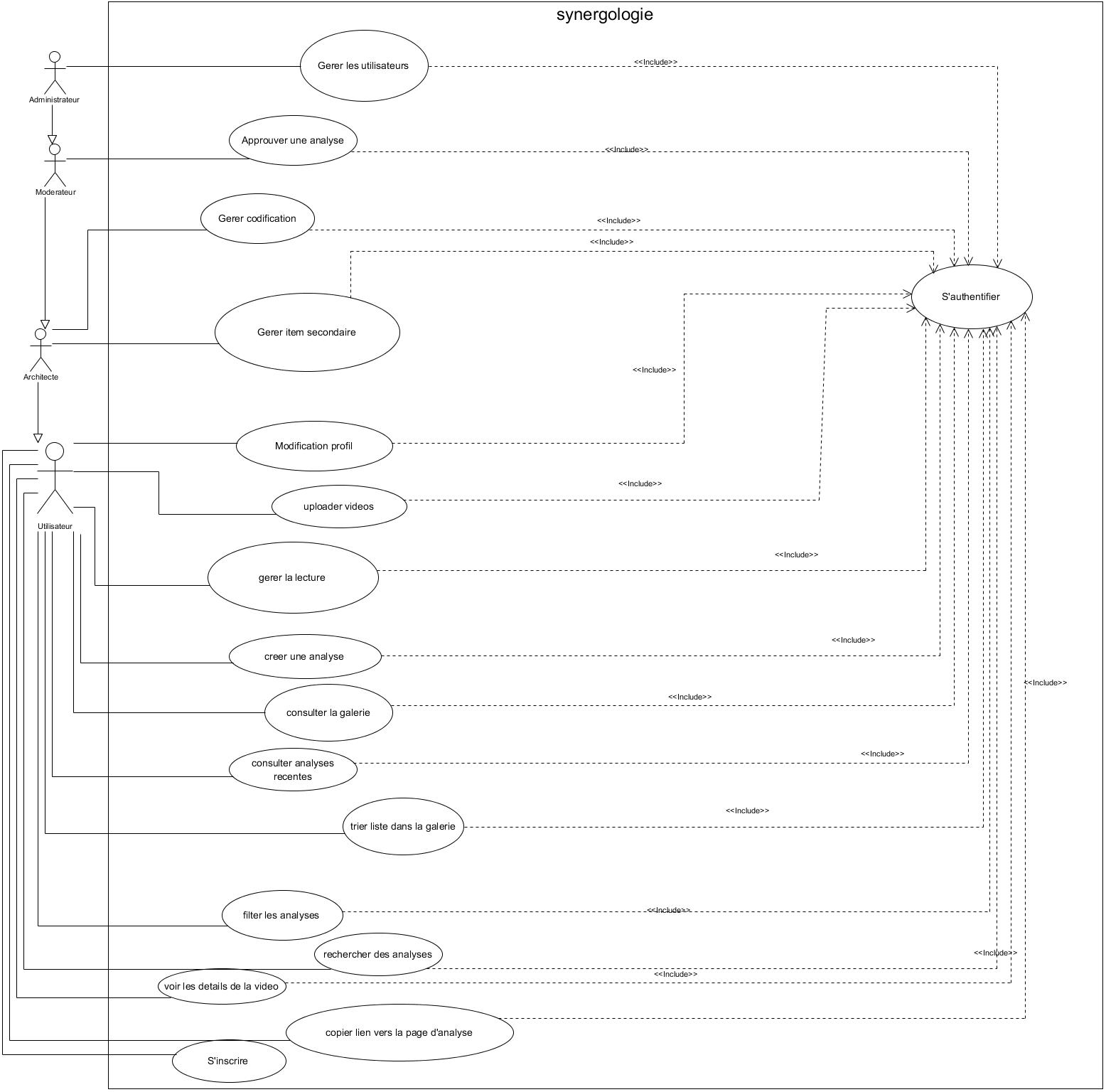


Figure 8 Diagrammes de cas d'utilisation

#### Priorisation et description des cas d’utilisation

##### Priorisation des cas d’utilisation

La figure 18 montre l’ordre de priorisation des cas d’utilisation

Tableau 18 Priorisation des cas d'utilisation

|  |  |
| --- | --- |
| Priorité | Cas d’utilisation |
| 1 | S’authentifier |
| 1 | Gérer les utilisateurs |
| 1 | S’inscrire |
| 3 | La gestion des items secondaires |
| 3 | La gestion d’une codification |
| 3 | L’approbation d’une analyse |
| 2 | L’upload vidéo |
| 3 | La gestion de lecture d’une vidéo |
| 3 | La galerie des vidéos |
| 2 | Gestion des analyses |

##### Détails des cas d’utilisation

**Description du cas d’utilisation « s’authentifier »**

* Nom du cas d’utilisation : s’authentifier
* Acteur : utilisateur
* Précondition : le responsable doit avoir un login/mot de passe
* Post condition : l’utilisateur arrive au page d’accueil
* Scénario nominal :
  + Le système affiche le formulaire d’authentification
  + L’utilisateur le remplit
  + Le système vérifie les informations
  + Le système redirige l’utilisateur vers la page d’accueil
* Scénario d’exception :
  + Un message d’erreur s’affiche en quoi l’email ou mot de passe incorrect

**Description du cas d’utilisation « Gérer les utilisateurs »**

* Nom du cas d’utilisation : Gérer les utilisateurs
* Acteur : administrateur
* Précondition : l’utilisateur s’est authentifié en tant qu’administrateur
* Post condition : l’utilisateur accède à la liste des utilisateurs
* Scénario nominal :
  + Le système affiche la liste des utilisateurs
  + L’utilisateur demande une mise à jour à un ou plusieurs utilisateurs
  + Le système envoie la modification et met à jour la liste
* Scénario d’exception :
  + : l’utilisateur saisit des informations incorrectes, afficher le message d’erreur et retour aux premières étapes du scénario nominal

**Description du cas d’utilisation « S’inscrire »**

* Nom du cas d’utilisation : s’inscrire
* Acteur : utilisateur
* Précondition : l’email n’a pas encore de compte sur le système
* Post condition : un mail de confirmation de création de comptes est envoyé
* Scénario nominal :
  + Le système affiche les formulaires à remplir
  + L’utilisateur joint son dossier pour l’inscription au système
  + L’utilisateur envoie les informations au système
  + Le système l’enregistre et envoie une page dans laquelle un mail de confirmation est envoyé à l’utilisateur
* Scénario d’exception :
  + : l’utilisateur saisit des informations incorrectes, afficher le message d’erreur et retour aux premières étapes du scénario nominal

**Description du cas d’utilisation « Gérer les items secondaires »**

* Nom du cas d’utilisation : Gérer les items secondaires
* Acteur : architecte
* Précondition : un utilisateur a fait une analyse dans le système
* Post condition : un item secondaire est ajouté dans le timeline
* Scénario nominal :
  + Le système affiche la page d’analyse
  + L’utilisateur choisit une analyse
  + L’utilisateur ajoute un item secondaire lié à l’analyse
  + Le système l’enregistre et renvoie la page d’analyse
* Scénario d’exception :
  + : l’utilisateur saisit des informations incorrectes, afficher le message d’erreur et retour aux premières étapes du scénario nominal

**Description du cas d’utilisation « Gérer les codifications »**

* Nom du cas d’utilisation : Gérer les codifications
* Acteur : architecte
* Précondition : l’utilisation arrive dans la page de gestion de codification
* Post condition : une partie de la classification mentale est insérée dans le système
* Scénario nominal :
  + Le système affiche la page de gestion de codification
  + L’utilisateur ajoute un nœud à l’arbre
  + L’utilisateur soumet l’enregistrement de l’arbre
  + L’arbre est inséré dans le système
  + Scénario d’exception : l’utilisateur saisit des informations incorrectes, afficher le message d’erreur et retour aux premières étapes du scénario nominal

**Description du cas d’utilisation « Approuver une analyse »**

* Nom du cas d’utilisation : Approuver une analyse
* Acteur : modérateur
* Précondition : une analyse a été créer dans le système
* Post condition : l’analyse est approuvée
* Scénario nominal :
  + Le système affiche la liste des analyses
  + L’utilisateur demande de mettre à jour l’état d’une analyse
  + L’utilisateur soumet l’enregistrement de la mise à jour
  + La liste des analyses a été mise à jour
* Scénario d’exception :
  + : l’utilisateur saisit des informations incorrectes, afficher le message d’erreur et retour aux premières étapes du scénario nominal

**Description du cas d’utilisation « Uploader une vidéo »**

* Nom du cas d’utilisation : Uploader une vidéo
* Acteur : utilisateur
* Précondition : l’utilisateur accède au système
* Post condition : la vidéo a été uploadée dans le système
* Scénario nominal :
  + Le système la page de soumission d’URL YouTube
  + Le système uploade la vidéo depuis YouTube vers le système
  + La vidéo est enregistrée
  + Le système redirige l’utilisateur vers la page d’analyse
* Scénario d’exception :
  + : l’utilisateur saisit des informations incorrectes, afficher le message d’erreur et retour aux premières étapes du scénario nominal

**Description du cas d’utilisation « Gérer la lecture d’une vidéo** **»**

* Nom du cas d’utilisation : Gérer la lecture d’une vidéo
* Acteur : utilisateur
* Précondition : une vidéo a été uploader dans le système
* Post condition : La vidéo est lue
* Scénario nominal :
  + Le système affiche la page d’analyse
  + L’utilisateur commence à lire la vidéo
  + L’utilisateur manipule la vidéo en bénéficiant des options de lecture de la vidéo
  + Scénario d’exception : le lecteur de vidéo ne fonctionne pas correctement

**Description du cas d’utilisation « Consulter la galerie des vidéos »**

* Nom du cas d’utilisation : Consulter la galerie des vidéos
* Acteur : utilisateur
* Précondition : des analyses ont été faites dans le système
* Post condition : La page de galerie d’analyse s’affiche
* Scénario nominal :
  + Le système affiche la page de galerie d’analyse vidéo
  + L’utilisateur veut trier la galerie
  + Scénario d’exception : l’utilisateur saisit des informations incorrectes, afficher le message d’erreur et retour aux premières étapes du scénario nominal

**Description du cas d’utilisation « Gérer les analyses »**

* Nom du cas d’utilisation : Gérer les analyses
* Acteur : utilisateur
* Précondition : une vidéo a été ajoutée dans le système
* Post condition : l’analyse est ajoutée dans le timeline
* Scénario nominal :
  + Le système affiche la page d’analyse
  + L’utilisateur fait une pause et ajoute une analyse
  + L’analyse est envoyée dans le système
  + Le système enregistre l’analyse
  + Un marqueur est affiché dans le timeline du lecteur
* Scénario d’exception :
  + : l’utilisateur saisit des informations incorrectes, afficher le message d’erreur et retour aux premières étapes du scénario nominal

### Capture des besoins techniques

Ce sont des exigences qui ne concernent pas spécifiquement le comportement du système, mais plutôt identifient des contraintes internes et externes du système. Et qui concerne notre application, nous avons dégagé les besoins suivants : le code doit être clair pour permettre des futures évolutions ou améliorations ; la disponibilité : l’application doit être disponible pour être utilisée à n’importe quel moment ; la sécurité : l’application doit respecter la confidentialité des données ; garantir l’intégrité et la cohérence des données à chaque mise à jour et à chaque insertion ; la convivialité de l’interface graphique : l’application doit fournir une interface conviviale et simple pour tout type d’utilisateur, car elle présente le premier contact de l’utilisateur avec l’application et par le biais de celle-ci, on découvrira se fonctionnalités.

### Conception préliminaire

#### Présentation de la méthode utilisée

##### Notion de processus unifié

Pour définir le processus unifié, on va simplement définir les 2 termes qui le composent :

* **Processus :** Suite continue d’opérations constituant la manière de fabriquer. En d’autres termes, c’est une succession de tâches dans le but d’accomplir un travail, un projet.
* **Unifié :** participe passé du verbe « unifier » être amené à l’unité, se fondre en un tout. En fait, les méthodes d’analyse et de conception orientée objet étaient variées jusqu’à ce que Rumbaugh, Jacobson et Booch aient eu l’idée de les unifier.

Le processus unifié s’appuie sur les principes suivants :

* **Piloté par les cas d’utilisation** : un cas d’utilisation représente une fonctionnalité qui satisfait un besoin d’un utilisateur. Le processus suit une voie spécifique, en procédant par une série d’enchainement d’activités, dérivées d’un cas d’utilisation.
* **Centré sur l’architecture** : l’architecture logicielle représente les aspects statiques et dynamiques du système. L’architecture émerge des besoins de l’entreprise, tel qu’ils sont exprimés par les utilisateurs et reflétés par les cas d’utilisation. L’architecture propose une vue d’ensemble de ka conception faisant ressortir les caractéristiques essentielles en laissant de côté les détails secondaires. Il faut noter que tout article est à la fois forme et fonction. L’une ou l’autre isolément ne saurait suffire. Les cas d’utilisation et l’architecture doivent s’équilibrer pour créer un article réussi. La figure suivante va représenter les différentes perspectives qui permettent de définir un modèle d’architecture appelée vue (4+1). Cette dernière a beaucoup inspiré UML.

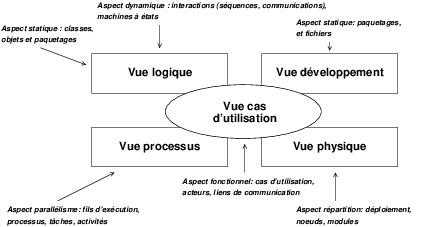


Figure 9 PU centré sur l'architecture

* **Itératif et incrémental** : vu que les projets à réaliser sont de plus en plus complexes et grands, l’idée est de découper le travail en mini projets. Chacun d’entre eux représente une itération qui donne lieu à un incrément. Les itérations désignent des étapes de l’enchainement d’activités tandis que les incréments correspondent à des stades de développement du produit.

La figure 10 représente le concept d’itération et incrémentation.

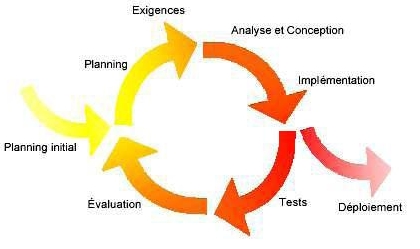


Figure 10 PU itératif et incrémental

**Les phases du processus unifié**

Le processus unifié se déroule en 4 phases : création, élaboration, construction et transition. Chaque phase répète un nombre de fois une série d’itérations. Et chaque itération est composée de 5 activités : capture des besoins, analyse, conception, implémentation et test.

* **Création :** c’est la première phase du processus unifié. Il s’agit de tracer ce qui doit figurer à l’intérieur du système et ce qui doit rester à l’extérieur, identifier les acteurs, lever les ambigüités sur les besoins et les exigences nécessaires dans cette phase.
* **Elaboration :** c’est la deuxième phase du processus. Il s’agit de purifier le modèle initial de cas d’utilisation, voire capturer de nouveaux besoins, analyser et concevoir la majorité des cas d’utilisation formulés et si possible implémenter et tester les cas d’utilisation initiaux.
* **Construction :** dans cette phase, il faut essayer de capturer tous les besoins. Ensuite, continuer l’analyse, la conception et surtout l’implémentation de tous les cas d’utilisation. À la fin de cette phase, les développeurs doivent fournir une version exécutable du système.
* **Transition :** il s’agit au cours de cette phase de vérifier si le système offre véritablement les services exigés par les utilisateurs, détecter les défaillances, combler les manques dans la documentation du logiciel et adapter le produit à l’environnement.

##### La méthode 2TUP

2TUP signifie « 2 Track Unified Process ». C’est un processus UP qui répond aux caractéristiques que nous venons de citer. Le processus 2TUP apporte une réponse aux contraintes de changement continuel imposées aux systèmes d’information de l’entreprise. En ce sens, il renforce le contrôle sur les capacités d’évolution et de correction de tels systèmes. «2 Track » signifie littéralement que le processus suit deux chemins. Il s’agit des chemins « fonctionnels » et « d’architecture technique », qui correspondent aux deux axes des changements imposés au système informatique.

La figure 11 représente le système d'information soumis à deux natures de contraintes

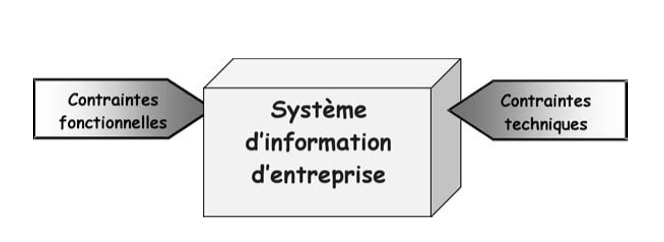


Figure 11 Systèmes d'information soumis à deux natures de contraintes

L’axiome fondateur du 2TUP consiste à constater que toute évolution imposée au système d’information peut se décomposer et se traiter parallèlement, suivant un axe fonctionnel et un axe technique.

À l’issue des évolutions du modèle fonctionnel et de l’architecture technique, la réalisation du système consiste à fusionner les résultats des deux branches. Cette fusion conduit à l’obtention d’un processus de développement en forme Y, comme illustré par la figure12.

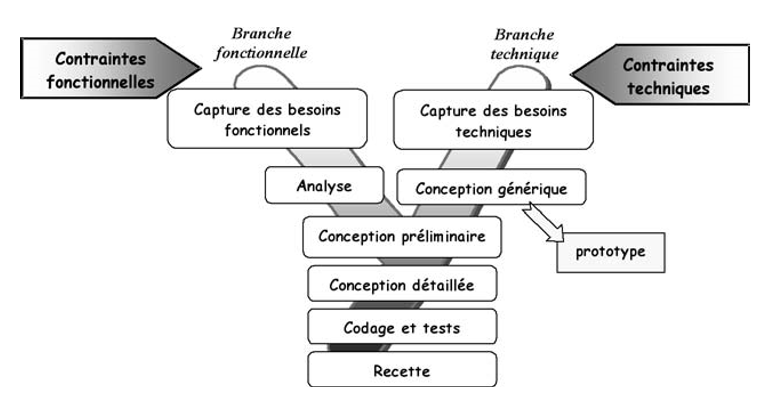


Figure 12 Processus de développement en Y

La branche gauche (fonctionnelle) comporte :

* La capture des besoins fonctionnels, qui produit un modèle des besoins focalisé sur le métier des utilisateurs. Elle qualifie au plus tôt le risque de produire un système inadapté aux utilisateurs. De son côté, la maitrise d’œuvre consolide les spécifications et en vérifie la cohérence et l’exhaustivité l’analyse, qui consiste à étudier précisément la spécification fonctionnelle de manière à obtenir une idée de ce que va réaliser le système en termes de métier. Les résultats de l’analyse ne dépendent d’aucune technologie particulière.

La branche droite (architecture technique) comporte :

* La capture des besoins techniques, qui recense toutes les contraintes et les choix dimensionnant la conception du système. Les outils et les matériels sélectionnés ainsi que la prise en compte de contraintes d’intégration avec l’existant conditionnent généralement des prérequis d’architecture technique.
* La conception générique, qui définit ensuite les composants nécessaires à la construction de l’architecture technique. Cette conception est la moins dépendante possible des aspects fonctionnels. Elle a pour objectif d’uniformiser et de réutiliser les mêmes mécanismes pour tout un système. L’architecture technique construit le squelette du système informatique et écarte la plupart des risques de niveau technique. L’importance de sa réussite est telle qu’il est conseillé de réaliser un prototype pour assurer sa validité.

La branche du milieu comporte :

* La conception préliminaire, qui représente une étape délicate, car elle intègre le modèle d’analyse dans l’architecture technique de manière à tracer la cartographie des composants du système à développer ;
* La conception détaillée, qui étudie ensuite comment réaliser chaque composant ;
* L’étape de codage, qui produit ces composants et teste au fur et à mesure les unités de code réalisées ;
* L’étape de recette, qui consiste enfin à valider les fonctions du système développé.

La méthode est par ailleurs incrémentale c’est-à-dire qu’à partir de la capture des besoins fonctionnels, on définit plusieurs cas d’utilisation représentant chacun un incrément du cycle de développement.

#### Diagrammes des séquences système

Le diagramme de séquences permet de représenter la succession chronologique des opérations et des collaborations entre objets selon un point de vue temporel. Il peut servir à illustrer un cas d’utilisation. On y met l’accent sur la chronologie des envois de messages qui est déterminée par sa position sur l’axe vertical du diagramme, le temps s’écoule de haut en bas de cet axe. La disposition des objets sur l’axe horizontal n’a pas de séquence pour la sémantique du diagramme.

* Cas d’utilisation s’authentifier

La figure 13 montre le diagramme de séquence système du cas d’utilisation s’authentifier

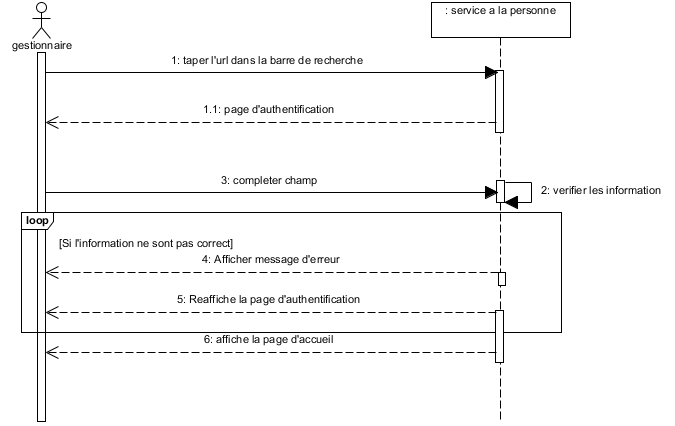


Figure 13 Séquence de système pour le cas d'utilisation s'authentifier

* Cas d’utilisation « Gérer utilisateur »

La figure 14 montre le diagramme de séquence système du cas d’utilisation gérer utilisateur

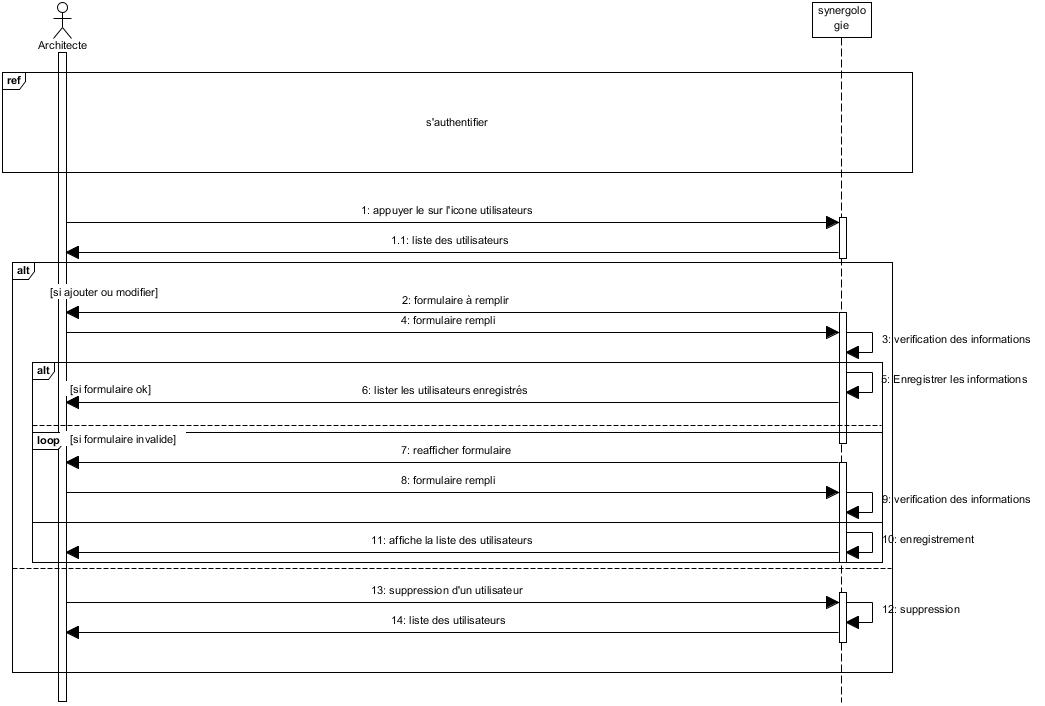


Figure 14 Diagramme de séquence système du cas d'utilisation gérer utilisateur

* Cas d’utilisation gérer codification

La figure 15 illustre le diagramme de séquence du cas d’utilisation gérer codification

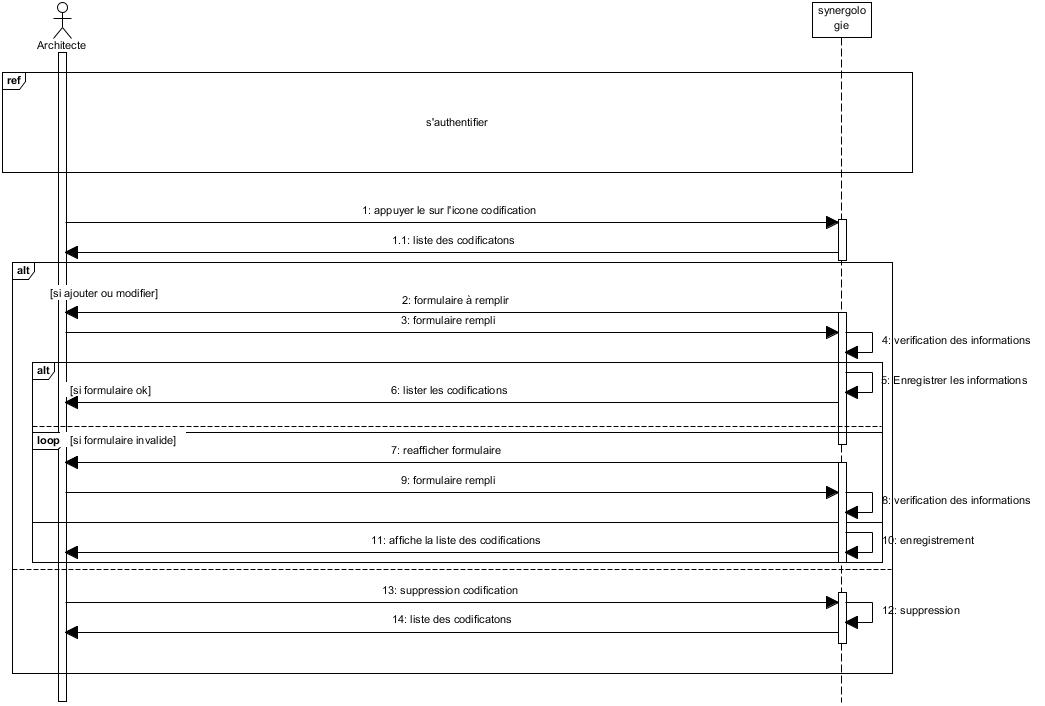


Figure 15 Diagramme de séquence du cas d'utilisation gérer codification

* Cas d’utilisation s’inscrire

La figure 16 illustre le diagramme de séquence du cas d’utilisation s’inscrire

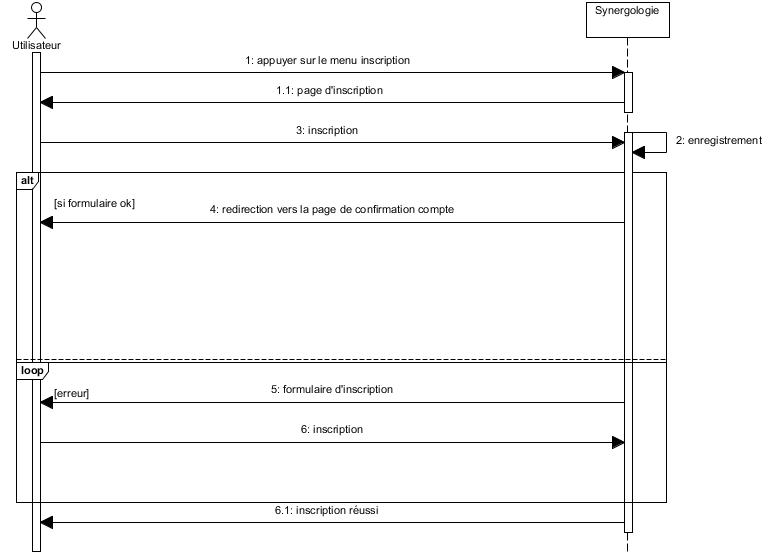


Figure 16 Diagramme de séquence du cas d'utilisation s'inscrire

* Cas d’utilisation gérer les items secondaires

La figure 17 illustre le diagramme de séquence du cas d’utilisation d’items secondaires

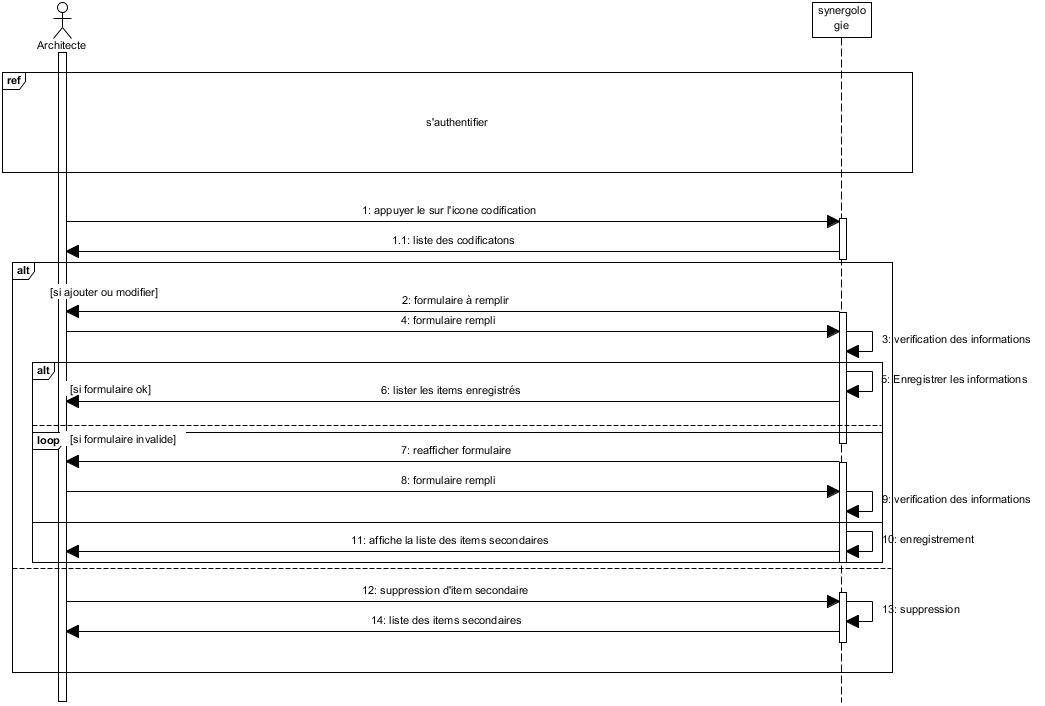


Figure 17 Diagramme de séquence de cas d'utilisation d'items secondaires

* Cas d’utilisation approuver une analyse

La figure 18 illustre le diagramme de séquence de cas d’utilisation approuver une analyse

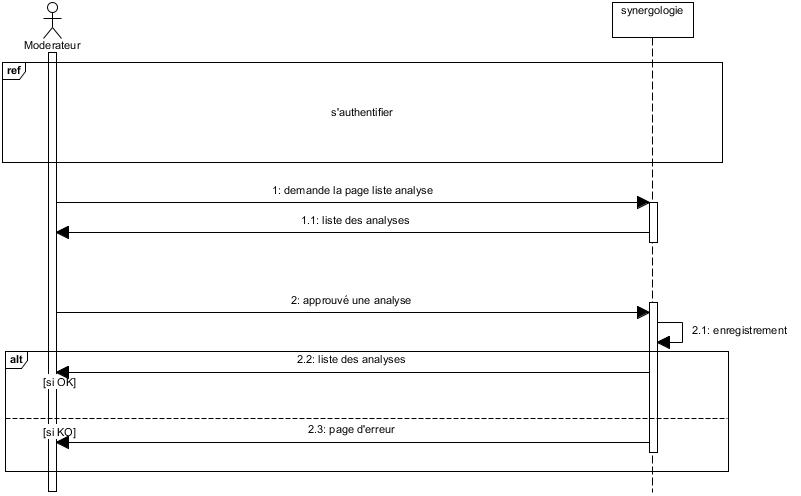


Figure 18 Diagramme de séquence de cas d'utilisation approuver une analyse

* Cas d’utilisation uploader vidéo

La figure 19 illustre le diagramme de séquence du cas d’utilisation uploader une vidéo

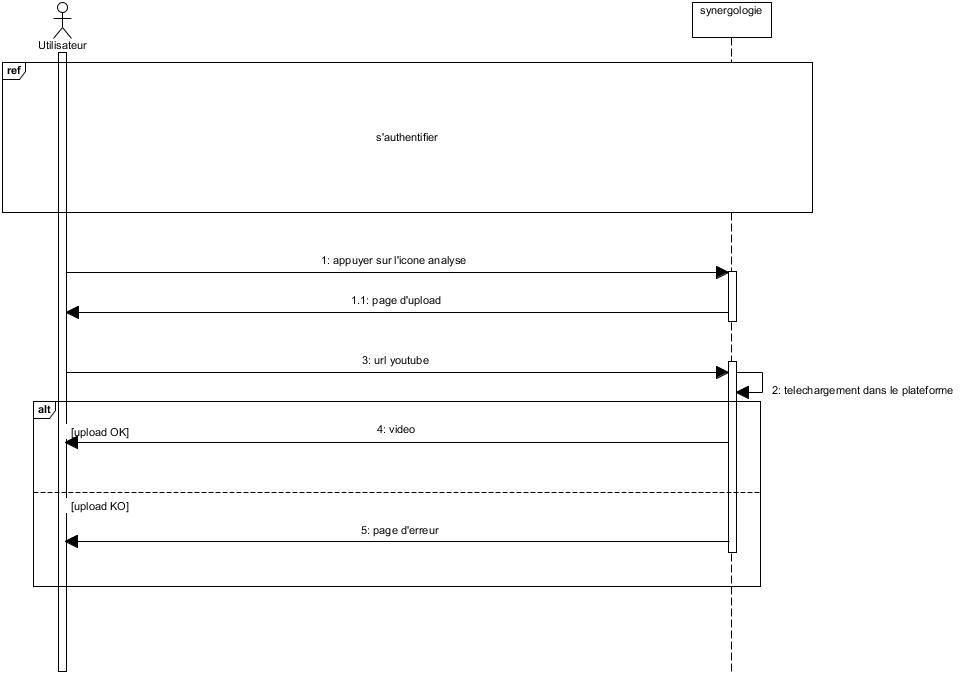


Figure 19 Diagramme de séquence du cas d'utilisation uploader vidéo

* Cas d’utilisation gérer lecture d’une vidéo

La figure 20 montre le diagramme de séquence du cas d’utilisation gérer lecture d’une vidéo



Figure 20 Diagramme de séquence du cas d'utilisation gérer lecture d'une vidéo

* Cas d’utilisation consulter galerie de vidéo

La figure 21 montre le diagramme de séquence du cas d’utilisation consulter galerie de vidéo

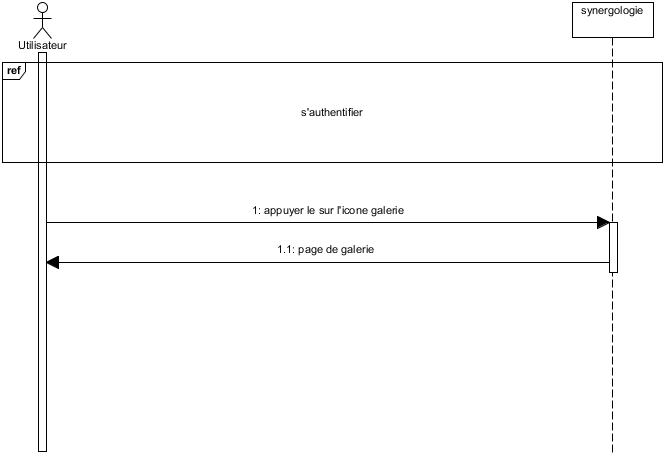


Figure 21 Diagramme de séquence du cas d'utilisation consulter galerie de vidéo

* Cas d’utilisation gérer les analyses

La figure 22 montre le diagramme de séquence du cas d’utilisation gérer les analyses

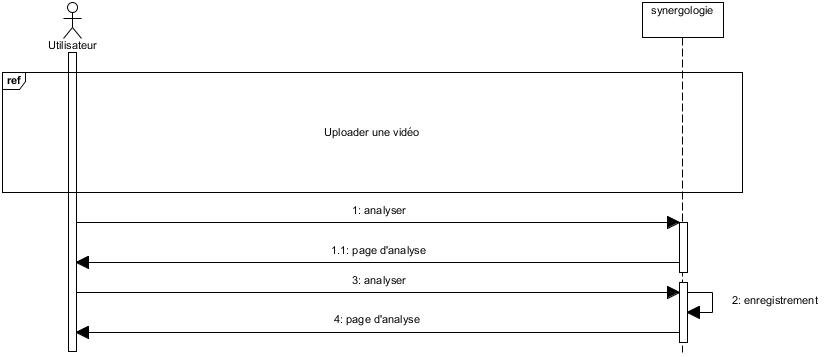


Figure 22 Diagrammes de séquence du cas d'utilisation gérer analyses

#### Modélisation du domaine

Le figure 23 montre le modèle de domaine

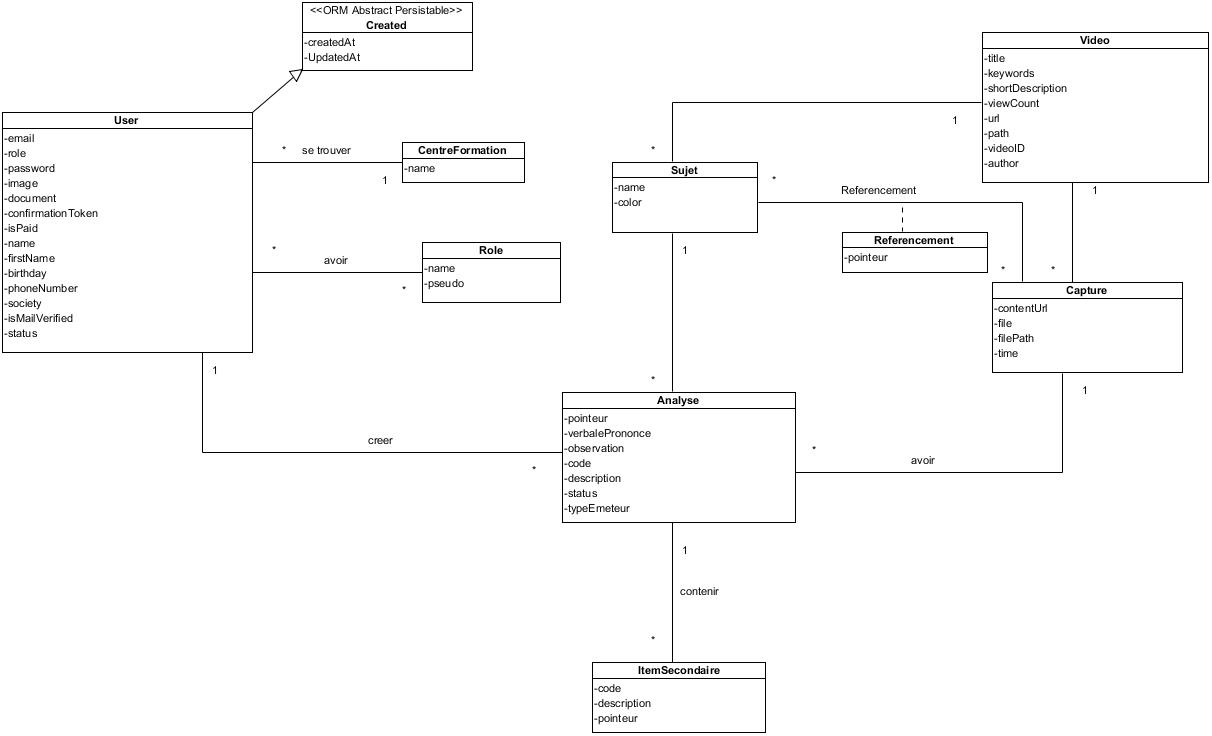


Figure 23 Modèle de domaine

## Conception détaillée

### Architecture du système

Pour l’architecture du projet, on a choisi d’adopter l’architecture MVC (Modèle Vue Contrôleur) couplée à une architecture 3-tiers pour structurer et fractionner le code du projet.

L’architecture MVC est un concept très puissant qui intervient dans la réalisation d’une application. Son principal intérêt est la séparation des données (modèle) de l’affichage (vue) et des actions (contrôleur).

Ce concept peut très bien être appliqué à un site internet. De nombreux sites internet disposent d’une plateforme d’administration pour modifier facilement les contenus. Il est bien évident que l’utilisateur qui va utiliser ce système pour créer des pages ne doit pas modifier la structure complète du site, ni les actions, ni l’affichage. Pour qu’il ne modifie que les données, celles-ci doivent être alors isolées.

C’est sur ce constat que se base le système MVC. Chaque partie est bien distincte. Les données peuvent provenir d’une source quelconque et la vue peut être conçue par des webdesigners n’ayant pas de connaissance en programmation.

L’architecture MVC est donc décomposée en trois couches :

* **Le modèle :** Le modèle correspond aux données, la plupart du temps stockées dans une base de données. Mais celle-ci peut également être contenue dans un fichier XML, ou dans des fichiers texte. Les données peuvent être exploitées sous forme de classes dans un langage de programmation orientée objet.
* **La vue :** La vue corresponde à un fichier HTML, peut-être un fichier XML, avec une feuille de style XLS la plupart du temps. Elle se contente d’afficher le contenu qu’elle reçoit sans avoir connaissance des données. Elle ne contient que les informations liées à l’affichage.
* **Le contrôleur :** le contrôler est l’élément qui va utiliser les données pour les envoyer à la vue. Son rôle est donc de récupérer les informations, de les traiter en fonction des paramètres demandés par la vue (par l’utilisateur, exemple : affiche les derniers articles), puis de renvoyer à la vue les données afin d’être affichée. Le contrôleur peut donc instancier différents objets (class user, class articles …) qui enverront des requêtes vers la base de données ou récupèreront des données XML la transmission à la vue des données récupérées se fait généralement à l’aide d’un Template.

La figure 24 montre l’architecture MVC.

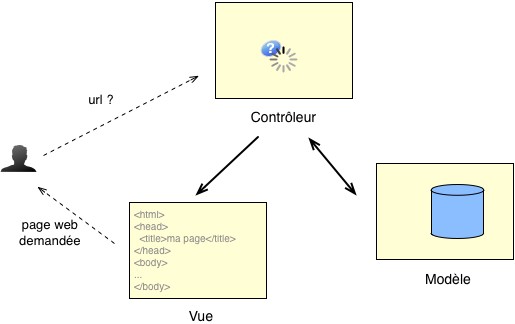
****

Figure 24 Architecture MVC

### Diagramme de séquence de conception

* Cas d’utilisation s’authentifier

La figure 25 illustre le diagramme de séquence de conception du cas d’utilisation s’authentifier.

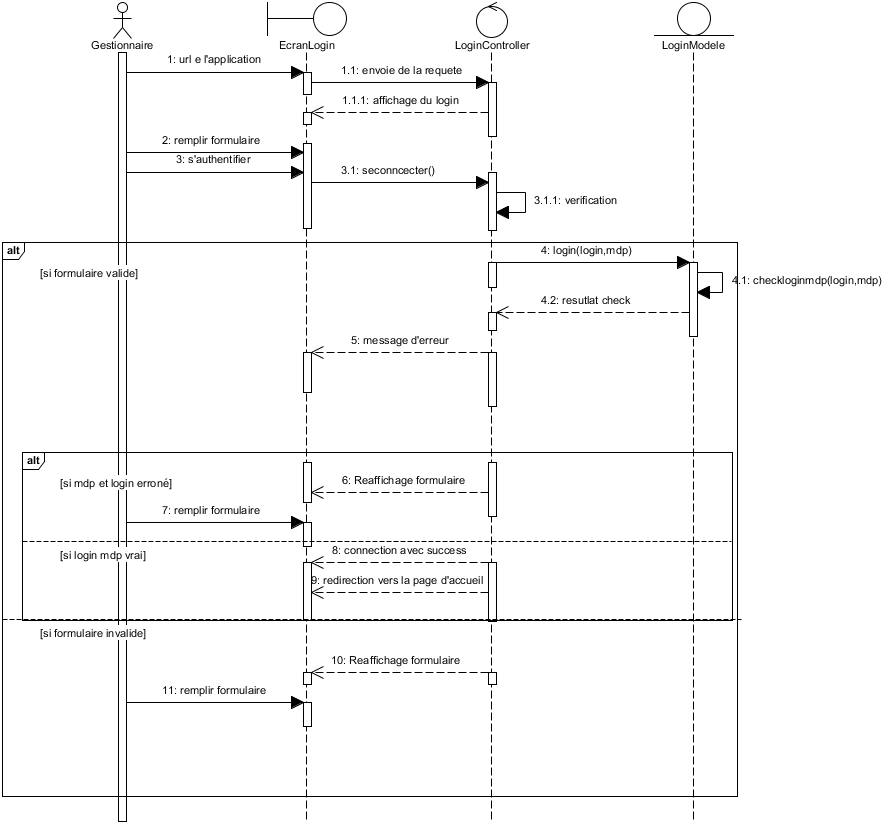


Figure 25 Diagramme de séquence de conception du cas d’utilisation s’authentifier

* Cas d’utilisation uploader vidéo

La figure 26 montre le diagramme de séquence de conception du cas d’utilisation uploader vidéo.

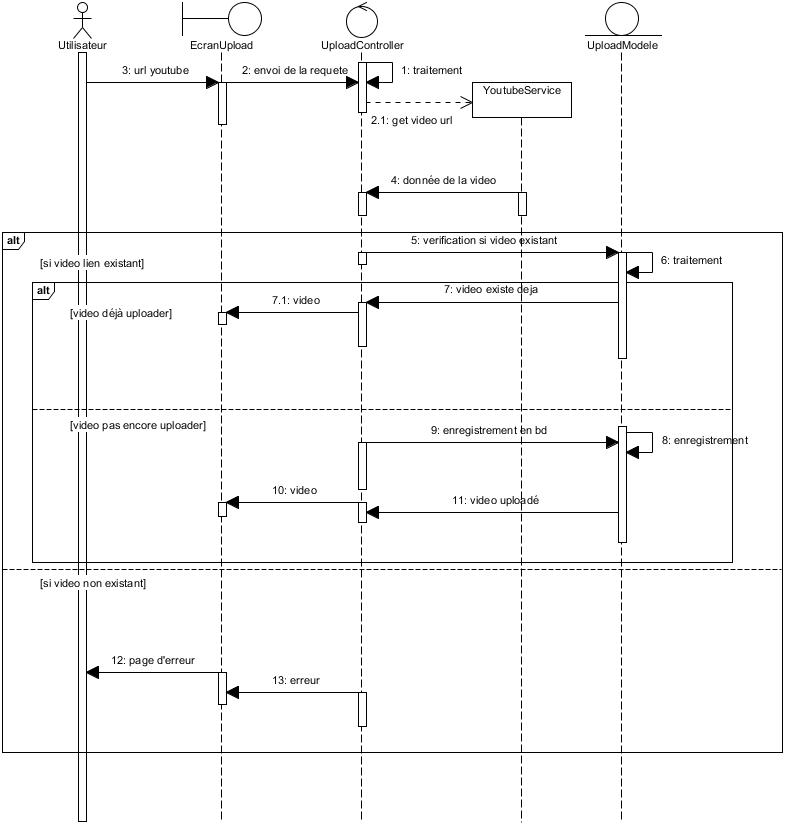


Figure Diagramme de séquence de conception du cas d'utilisation uploader vidéo

### Diagramme de classe de conception

L’application implémentera le pattern Modèle-Vue-Contrôleur. Les classes des modèles de conception seront donc stéréotypées « Modele », « View » ou « Controller ».

* Cas d’utilisation s’authentifier

La figure 27 illustre le diagramme de classe de conception du cas d’utilisation s’authentifier

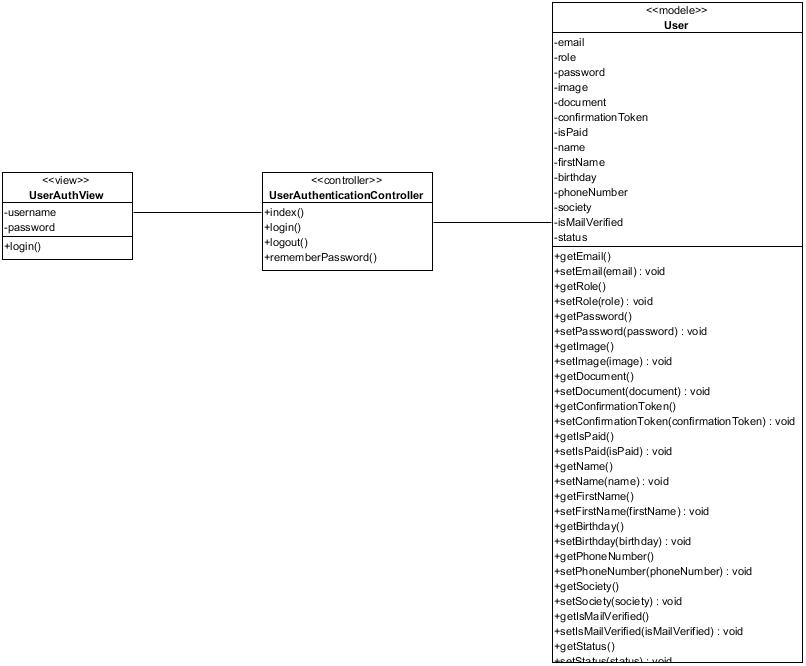


Figure 27 Diagramme de classe de conception du cas d'utilisation s'authentifier

* Cas d’utilisation gérer utilisateur

La figure 28 montre le diagramme de classe de conception du cas d’utilisation gérer utilisateur

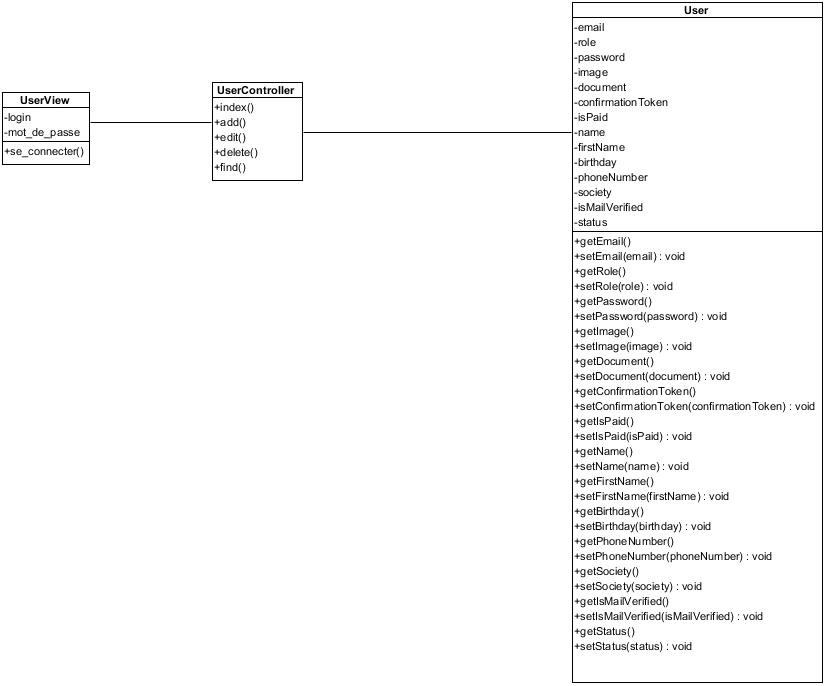


Figure Diagramme de classe de conception de gérer utilisateurs

* Cas d’utilisation gérer codification

La figure 29 montre le diagramme de classe de conception du cas d’utilisation gérer codification

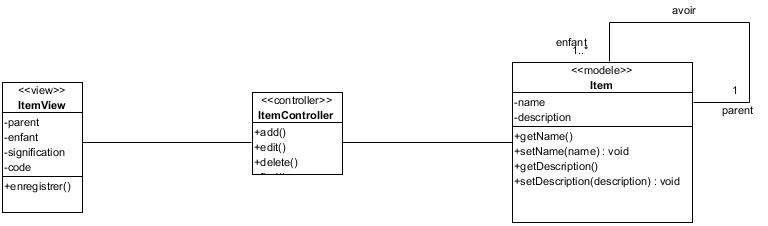


Figure 29 Diagramme de classe de conception de gérer codification

* Cas d’utilisation gérer items secondaires

La figure 30 illustre le diagramme de classe de conception du cas d’utilisation gérer items secondaires

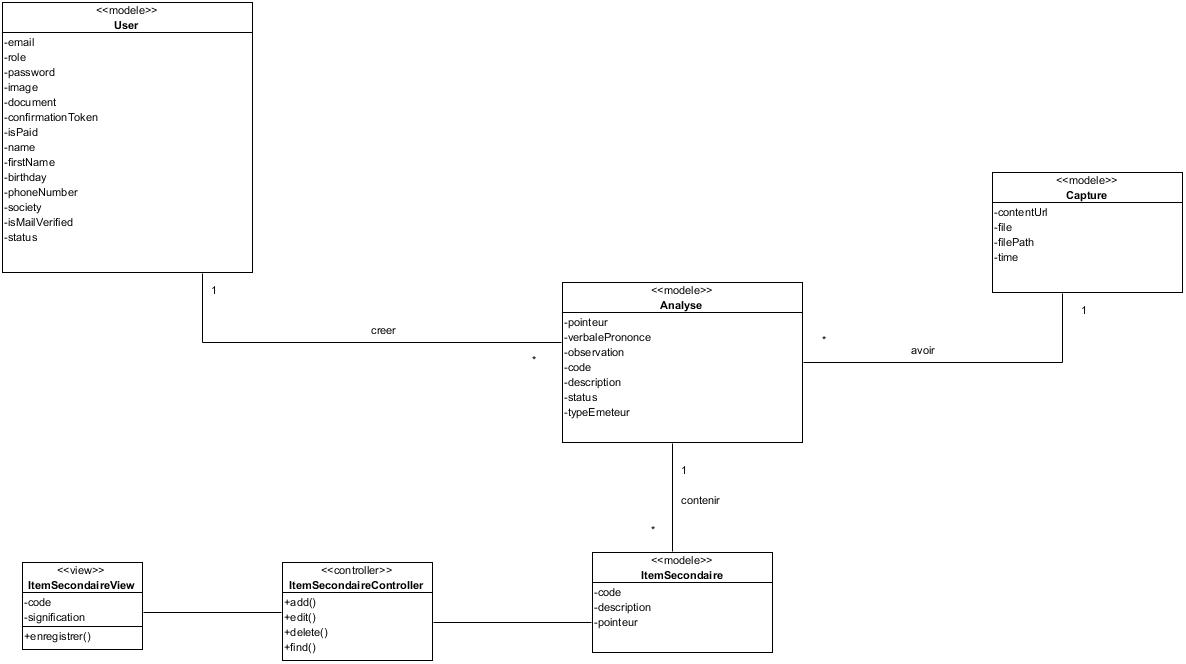


Figure 30 Diagramme de classe de conception de gérer items secondaires

* Cas d’utilisation upload vidéo

La figure 31 montre le diagramme de classe de conception du cas d’utilisation upload vidéo

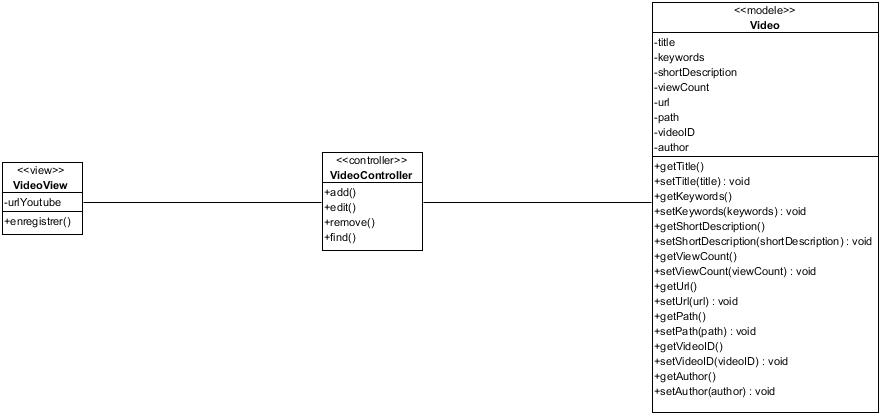


Figure 31 Diagramme de classe de conception de upload vidéo

* Cas d’utilisation approuver analyse

La figure 32 montre le diagramme de classe de conception du cas d’utilisation approuver analyse

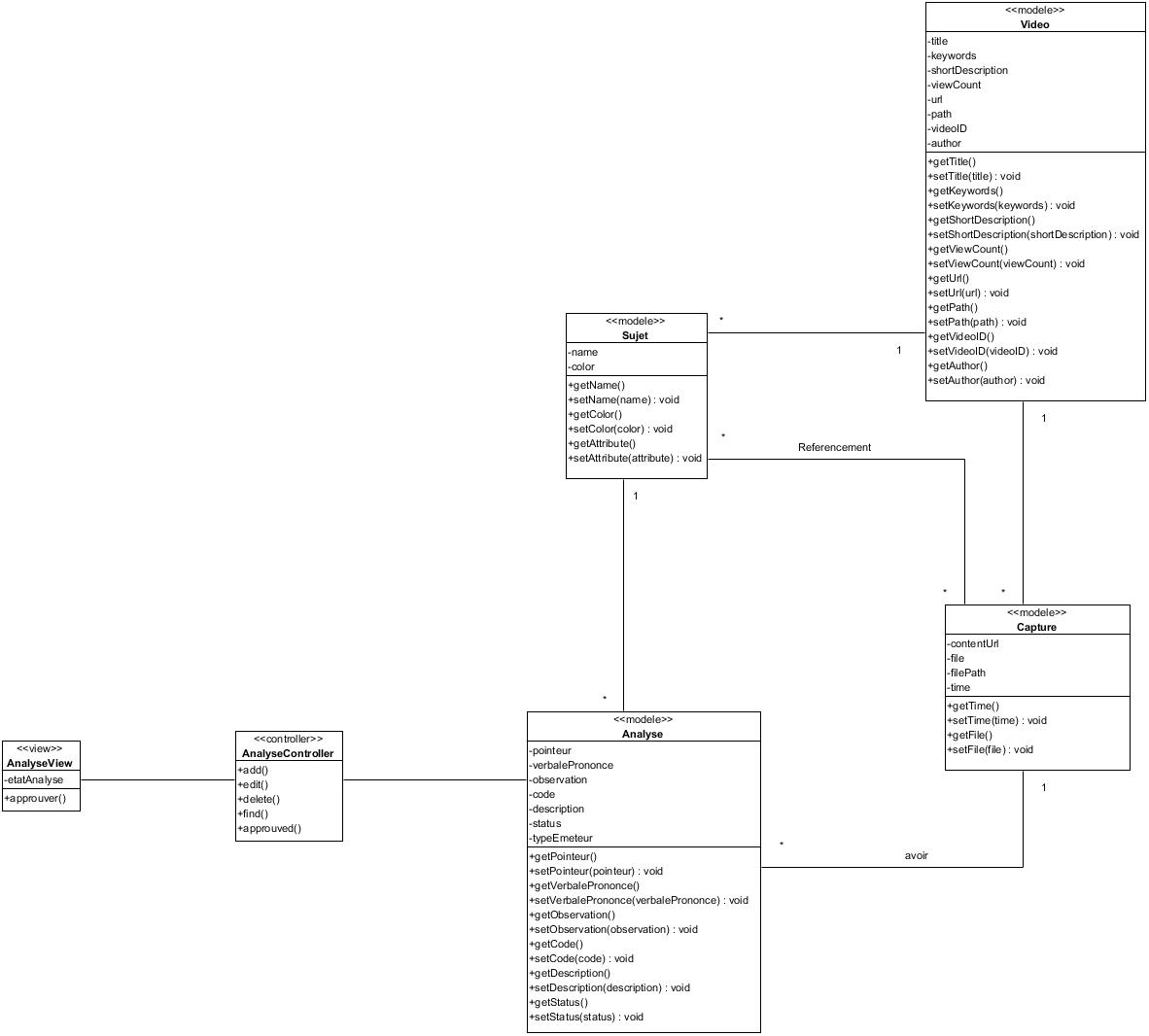


Figure 32 Diagramme de classe de conception du cas d’utilisation approuver analyse

* Cas d’utilisation gérer lecture d’une vidéo

La figure 33 illustre le diagramme de classe de conception du cas d’utilisation gérer lecture d’une vidéo

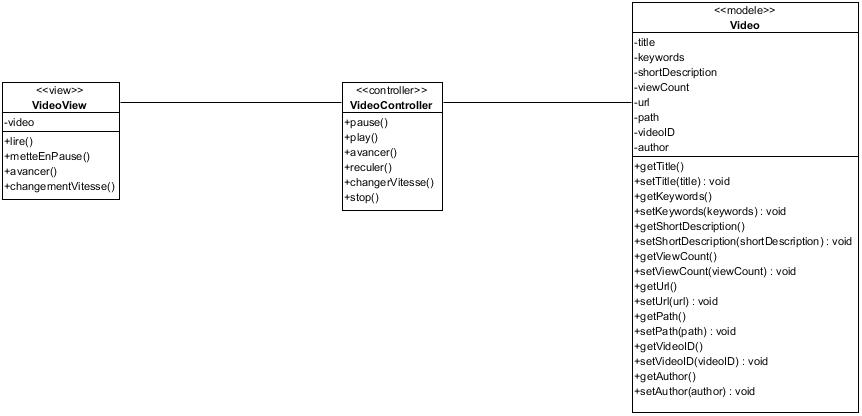


Figure 33 Diagramme de classe de conception de gérer lecture d'une vidéo

* Cas d’utilisation consulter la galerie

La figure 34 illustre le diagramme de classe de conception du cas d’utilisation consulter la galerie.



Figure 34 Diagramme de classe de conception de consulter galerie

* Cas d’utilisation gérer analyse

La figure 35 montre le diagramme de classe de conception du cas d’utilisation gérer analyse

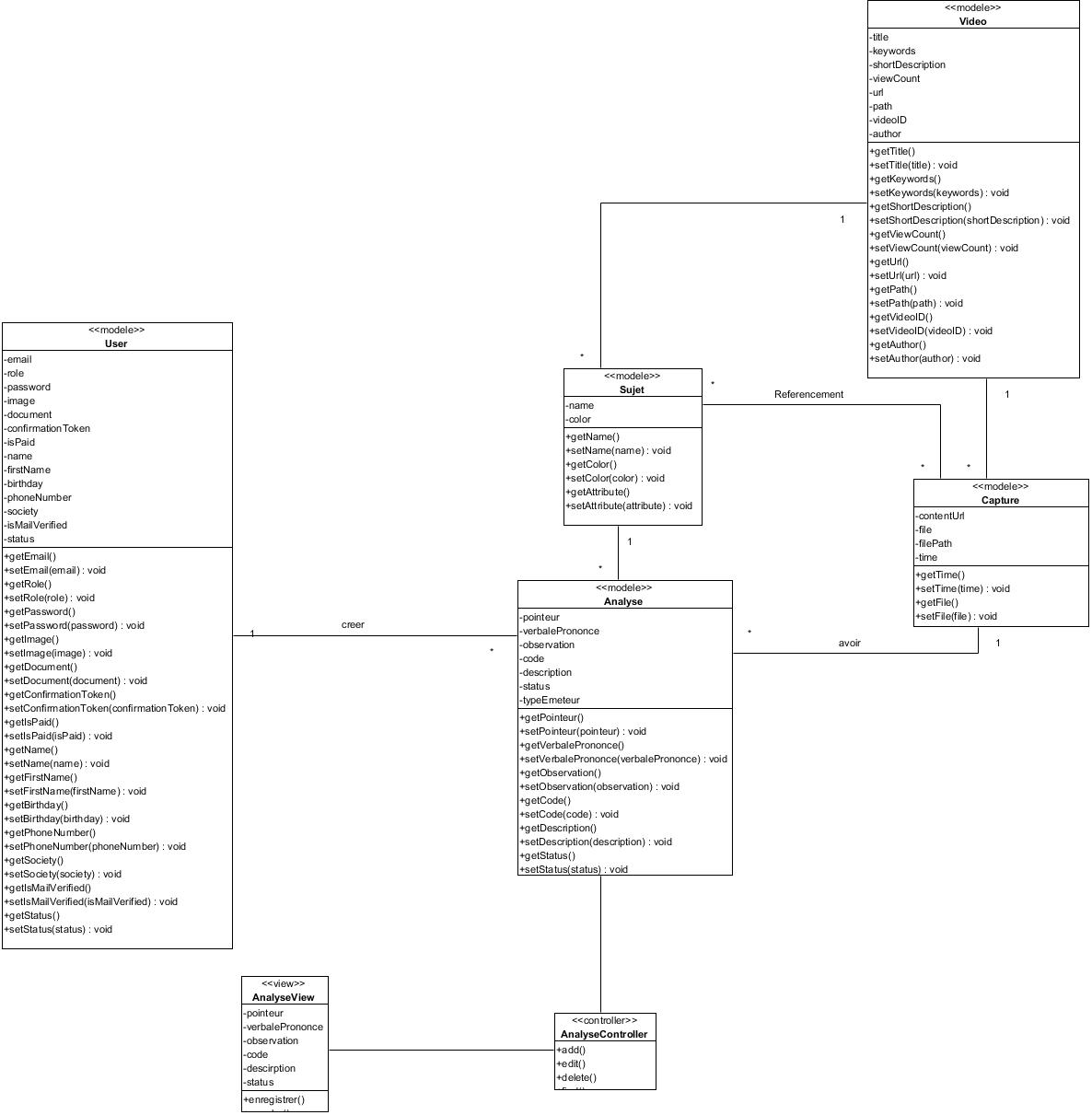


Figure 35 Diagramme de classe de conception de gérer analyse

### Diagramme de classe de conception global

La figure 36 illustre le diagramme de classe de conception global

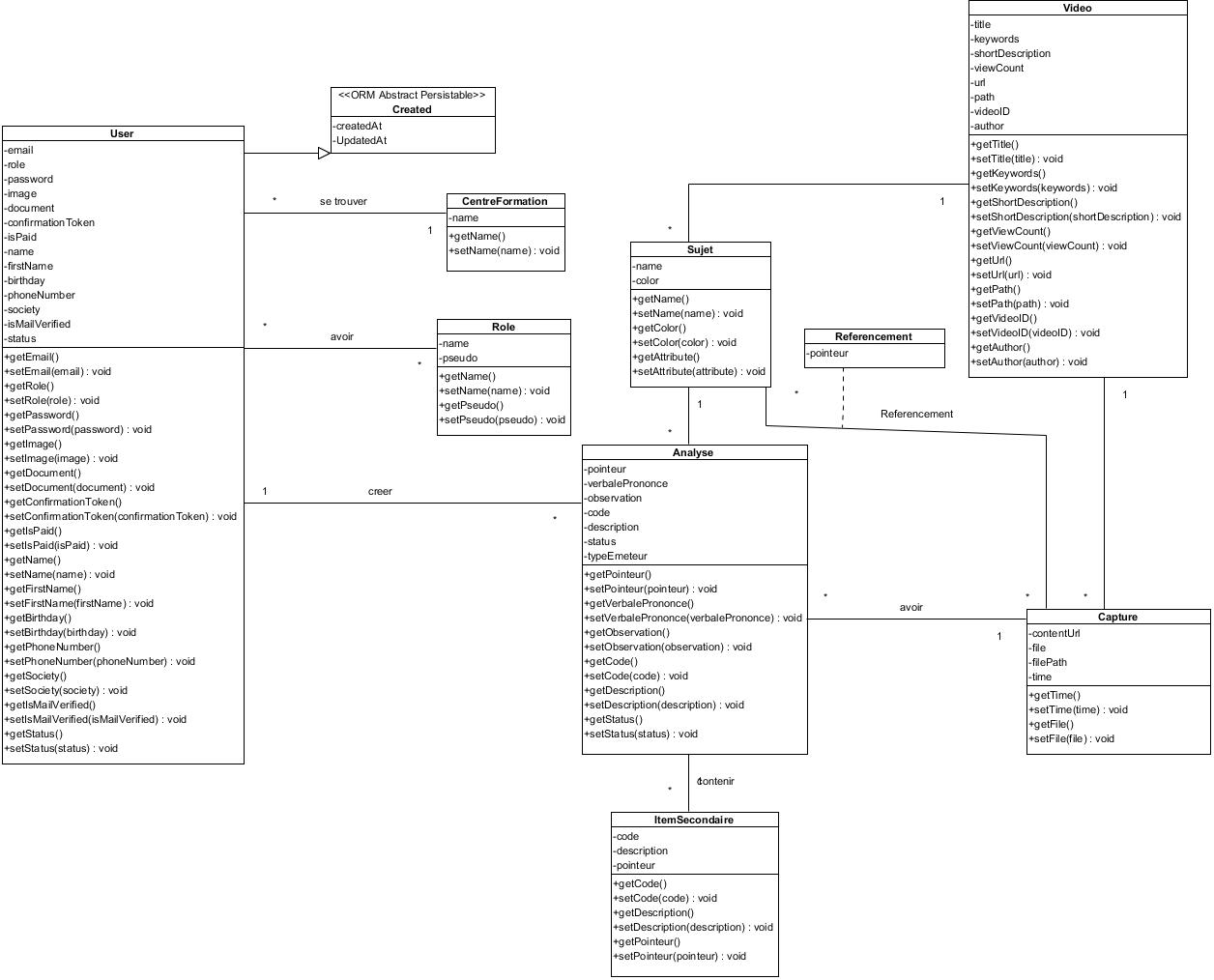


Figure 36 Diagramme de classe de conception globale

### Diagramme de paquetage

La figure 37 illustre le diagramme de paquetage.

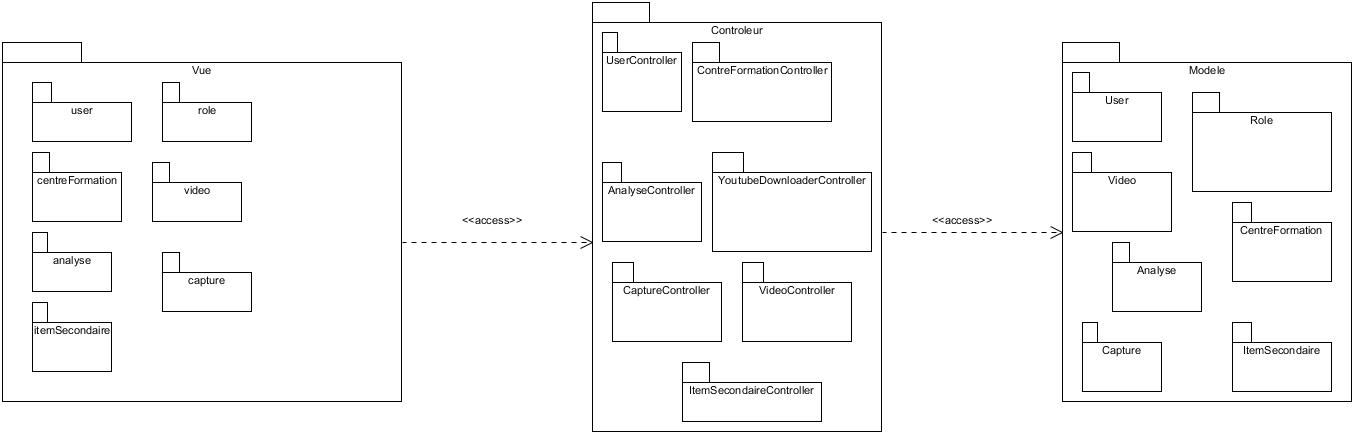


Figure 37 Diagrammes de paquetage

### Diagramme de déploiements

Les diagrammes de déploiement montrent la disposition physique des matériels qui composent le système et la répartition des composants sur ces matériels et les ressources matérielles sont représentées sous forme de nœuds. Les nœuds sont connectés entre eux, à l’aide d’un support de communication. La nature des lignes de communication et leurs caractéristiques peuvent être précisées. Les diagrammes de déploiement peuvent montrer des instances de nœuds (un matériel précis), ou des classes de nœuds.

La figure 38 montre le diagramme de déploiement de l’application

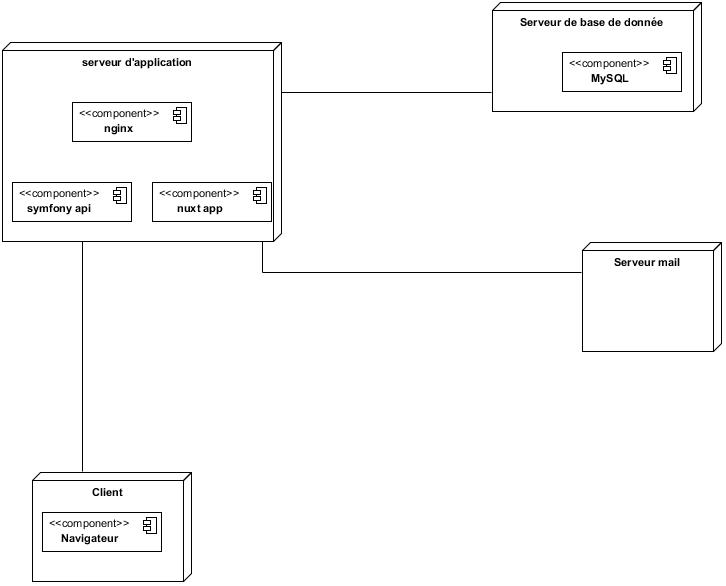


Figure 38 Diagramme de déploiement

# REALISATION

## Mise en place de l’environnement de développement

### Installation et configuration des outils

#### Visual Paradigm sur windows

Visual paradigm est utilisé pour la partie conception qui aboutit surtout à l’obtention de la base de données à partir du diagramme de classe.

La figure 39 montre la toute première étape de son installation.

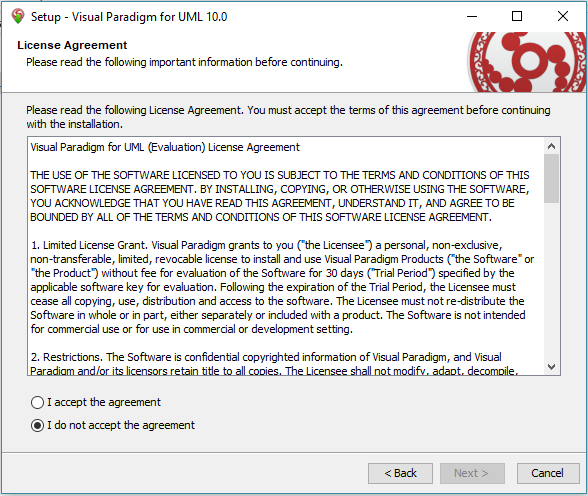


Figure 39 Installation de visual paradigm

#### Installation de php7.4, phpMyAdmin, MySQL(MariaDB) sous Ubuntu 20

##### Php7.4

sudo apt update

sudo apt install php php-cli php-fpm php-json php-common php-mysql php-zip php-gd php-mbstring php-curl php-xml php-pear php-bcmath

Cette commande suffise pour installer PHP dans notre environnement de développement.

##### phpMyAdmin

Pour installer phpMyAdmin, il faut installer apache2, et mariadb. Les commandes pour leurs mise en place sont :

sudo apt update

sudo apt install apache2

Après avoir installé Apache2, the les commande suivantes nous aidera à stopper , démarrer  et activer  le service d’Apache2 pour activer le service au démarrage de notre serveur.

Ensuite on installe MariaDB

sudo apt install mariadb-server mariadb-client

Après avoir installé MariaDB, les commandes suivantes nous aideront à stopper , démarrer  et activer  le service de MariaDB pour activer le service au démarrage de notre serveur.

Après, taper la commande suivante pour sécuriser MariaDB en créant un compte administrateur.

**sudo mysql\_secure\_installation**

#### PHPStorm

PhpStorm est idéal pour travailler avec Symfony, Laravel, Drupal, WordPress, Zend Framework, Magento, Joomla!, CakePHP, Yii, entre autres.

La figure 40 montre l’interface de phpStorm après l’avoir installé dans notre environnement de développement

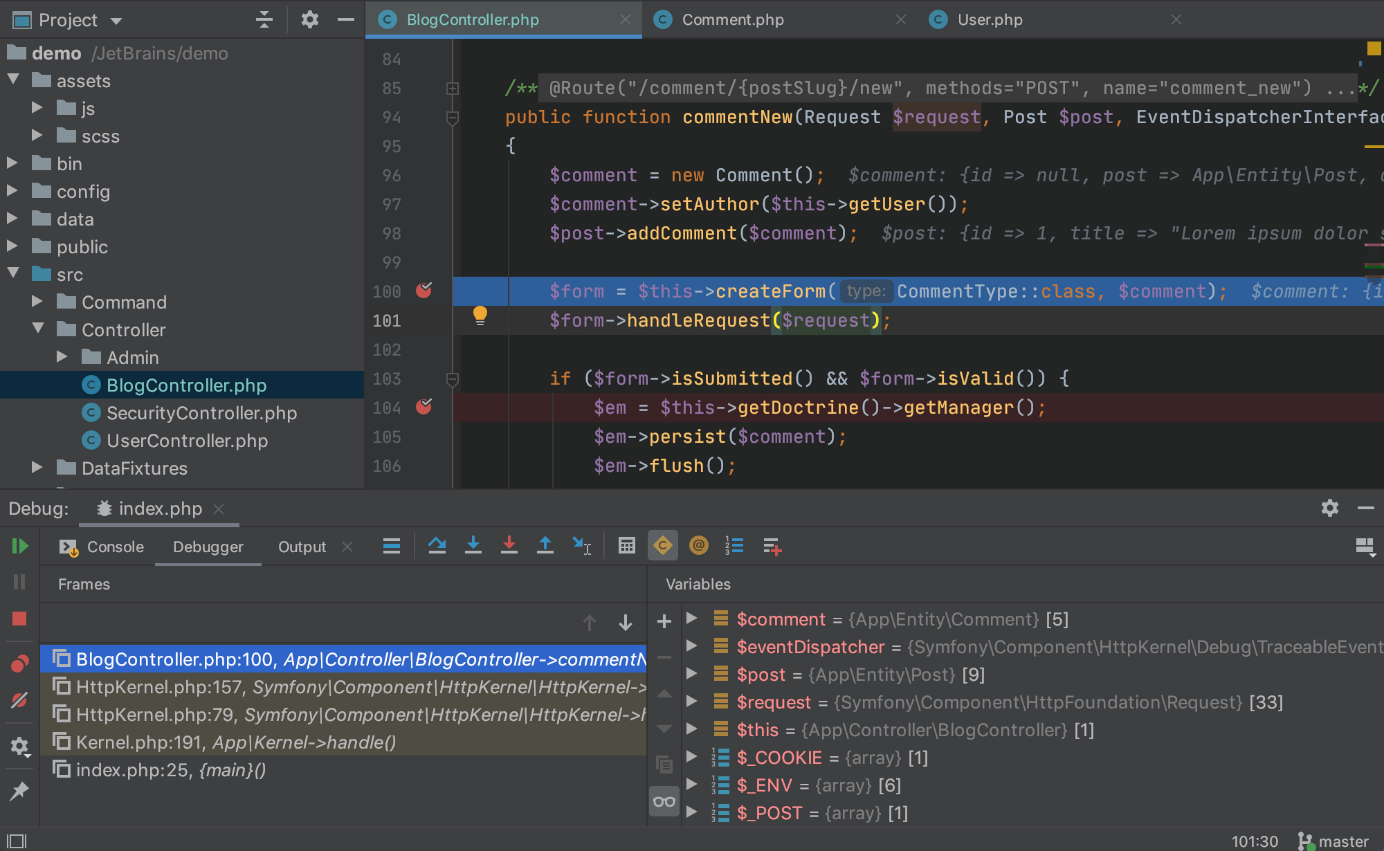


Figure 40 Interface de phpStorm

### Architecture de l’application

Vu que les besoins de la société adoptent le style, architecture 3-tiers, il sera préférable d’adopter ce type d’architecture, cela économisera le cout et le temps.

L’architecture trois tierce architecture 3-tiers, architecture à trois niveaux ou architecture à trois couches est l’application du modèle plus général qui est le multi-tiers. L’architecture logique du système est divisée en trois niveaux ou couches :

* Couche de présentation ;
* Couche métier ;
* Couche accès aux données

La figure 41 illustre l’architecture 3-tiers

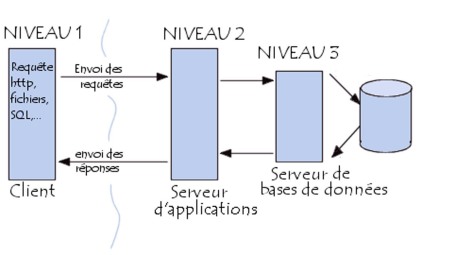


Figure 41 Architecture 3-tiers

## Développement de l’application

### Création de la base de données

Symfony nous permet de générer la base de données à l’aide de DOCTRINE qui est un des principaux points forts de ce framework. Avec un composant dotEnv on écrira la configuration de la base de données.

DATABASE\_URL="mysql://root:@127.0.0.1:3306/synergology-api-local?serverVersion=5.7"

La configuration de doctrine se fait dans les fichiers : config/packages/doctrine.yamldont voici les extraits :

doctrine:

    dbal:

        override\_url: **true**

        url: '%env(resolve:DATABASE\_URL)%'

*# IMPORTANT: You MUST configure your server version,*

*# either here or in the DATABASE\_URL env var (see .env file)*

*#server\_version: '13'*

    orm:

        auto\_generate\_proxy\_classes: **true**

        naming\_strategy: doctrine.orm.naming\_strategy.underscore\_number\_aware

        auto\_mapping: **true**

        mappings:

            App:

                is\_bundle: **false**

                type: annotation

                dir: '%kernel.project\_dir%/src/Entity'

                prefix: 'App\Entity'

                alias: App

La création de bases de données se fait avec la commande :

PHP bin/console doctrine:database:create

Après, on crée les entités :

PHP bin/console make:entity

Une fois le modèle défini, on peut générer la table associée dans la base de données.

Pour cela on utilise la commande :

PHP bin/console make:migration

Et finalement, pour l’exécution de la requête, on utilise la commande :

PHP bin/console doctrine:migrations:migrate

Si l’on se réfère au diagramme de classe global, voici le migration qu’on peut en tirer

**<?php**

**declare**(strict\_types=1);

**namespace** DoctrineMigrations;

**use** Doctrine\DBAL\Schema\Schema;

**use** Doctrine\Migrations\AbstractMigration;

*/\*\**

*\* Auto-generated Migration: Please modify to your needs!*

*\*/*

**final** **class** Version20210825122322 **extends** AbstractMigration

{

**public** **function** getDescription(): string

    {

**return** '';

    }

**public** **function** up(Schema $schema): void

    {

*// this up() migration is auto-generated, please modify it to your needs*

        $this->addSql('CREATE TABLE analyse (id INT AUTO\_INCREMENT NOT NULL, capture\_id INT NOT NULL, author\_id INT NOT NULL, sujet\_id INT DEFAULT NULL, pointeur VARCHAR(255) DEFAULT NULL, sujet\_analyse VARCHAR(255) DEFAULT NULL, verbale\_prononce VARCHAR(255) DEFAULT NULL, observation LONGTEXT DEFAULT NULL, code VARCHAR(255) DEFAULT NULL, description LONGTEXT DEFAULT NULL, status TINYINT(1) NOT NULL, type\_emeteur VARCHAR(255) DEFAULT NULL, INDEX IDX\_351B0C7E6B301384 (capture\_id), INDEX IDX\_351B0C7EF675F31B (author\_id), INDEX IDX\_351B0C7E7C4D497E (sujet\_id), PRIMARY KEY(id)) DEFAULT CHARACTER SET utf8mb4 COLLATE `utf8mb4\_unicode\_ci` ENGINE = InnoDB');

        $this->addSql('CREATE TABLE capture (id INT AUTO\_INCREMENT NOT NULL, video\_id INT DEFAULT NULL, file\_path VARCHAR(255) DEFAULT NULL, time VARCHAR(20) DEFAULT NULL, INDEX IDX\_8BFEA6E529C1004E (video\_id), PRIMARY KEY(id)) DEFAULT CHARACTER SET utf8mb4 COLLATE `utf8mb4\_unicode\_ci` ENGINE = InnoDB');

        $this->addSql('CREATE TABLE country (id INT AUTO\_INCREMENT NOT NULL, name VARCHAR(255) NOT NULL, PRIMARY KEY(id)) DEFAULT CHARACTER SET utf8mb4 COLLATE `utf8mb4\_unicode\_ci` ENGINE = InnoDB');

        $this->addSql('CREATE TABLE item (id INT AUTO\_INCREMENT NOT NULL, parent\_id INT DEFAULT NULL, name VARCHAR(10) NOT NULL, description VARCHAR(255) NOT NULL, INDEX IDX\_1F1B251E727ACA70 (parent\_id), PRIMARY KEY(id)) DEFAULT CHARACTER SET utf8mb4 COLLATE `utf8mb4\_unicode\_ci` ENGINE = InnoDB');

        $this->addSql('CREATE TABLE item\_secondaire (id INT AUTO\_INCREMENT NOT NULL, analyse\_id INT NOT NULL, code VARCHAR(255) DEFAULT NULL, description LONGTEXT DEFAULT NULL, pointeur VARCHAR(255) DEFAULT NULL, INDEX IDX\_CC2A70951EFE06BF (analyse\_id), PRIMARY KEY(id)) DEFAULT CHARACTER SET utf8mb4 COLLATE `utf8mb4\_unicode\_ci` ENGINE = InnoDB');

        $this->addSql('CREATE TABLE media\_object\_document (id INT AUTO\_INCREMENT NOT NULL, file\_path VARCHAR(255) DEFAULT NULL, status VARCHAR(10) NOT NULL, created\_at DATETIME NOT NULL, updated\_at DATETIME DEFAULT NULL, name VARCHAR(255) NOT NULL, PRIMARY KEY(id)) DEFAULT CHARACTER SET utf8mb4 COLLATE `utf8mb4\_unicode\_ci` ENGINE = InnoDB');

        $this->addSql('CREATE TABLE media\_object\_image (id INT AUTO\_INCREMENT NOT NULL, file\_path VARCHAR(255) DEFAULT NULL, name VARCHAR(255) NOT NULL, PRIMARY KEY(id)) DEFAULT CHARACTER SET utf8mb4 COLLATE `utf8mb4\_unicode\_ci` ENGINE = InnoDB');

        $this->addSql('CREATE TABLE password\_token (id INT AUTO\_INCREMENT NOT NULL, user\_id INT NOT NULL, token VARCHAR(50) NOT NULL, expires\_at DATETIME NOT NULL, UNIQUE INDEX UNIQ\_BEAB6C245F37A13B (token), INDEX IDX\_BEAB6C24A76ED395 (user\_id), PRIMARY KEY(id)) DEFAULT CHARACTER SET utf8mb4 COLLATE `utf8mb4\_unicode\_ci` ENGINE = InnoDB');

        $this->addSql('CREATE TABLE referencement (id INT AUTO\_INCREMENT NOT NULL, capture\_id INT NOT NULL, sujet\_id INT NOT NULL, pointeur VARCHAR(10) NOT NULL, INDEX IDX\_83A125FE6B301384 (capture\_id), INDEX IDX\_83A125FE7C4D497E (sujet\_id), PRIMARY KEY(id)) DEFAULT CHARACTER SET utf8mb4 COLLATE `utf8mb4\_unicode\_ci` ENGINE = InnoDB');

        $this->addSql('CREATE TABLE refresh\_tokens (id INT AUTO\_INCREMENT NOT NULL, refresh\_token VARCHAR(128) NOT NULL, username VARCHAR(255) NOT NULL, valid DATETIME NOT NULL, UNIQUE INDEX UNIQ\_9BACE7E1C74F2195 (refresh\_token), PRIMARY KEY(id)) DEFAULT CHARACTER SET utf8mb4 COLLATE `utf8mb4\_unicode\_ci` ENGINE = InnoDB');

        $this->addSql('CREATE TABLE role (id INT AUTO\_INCREMENT NOT NULL, name VARCHAR(20) NOT NULL, pseudo VARCHAR(20) NOT NULL, PRIMARY KEY(id)) DEFAULT CHARACTER SET utf8mb4 COLLATE `utf8mb4\_unicode\_ci` ENGINE = InnoDB');

        $this->addSql('CREATE TABLE role\_user (role\_id INT NOT NULL, user\_id INT NOT NULL, INDEX IDX\_332CA4DDD60322AC (role\_id), INDEX IDX\_332CA4DDA76ED395 (user\_id), PRIMARY KEY(role\_id, user\_id)) DEFAULT CHARACTER SET utf8mb4 COLLATE `utf8mb4\_unicode\_ci` ENGINE = InnoDB');

        $this->addSql('CREATE TABLE sujet (id INT AUTO\_INCREMENT NOT NULL, video\_id INT NOT NULL, name VARCHAR(50) NOT NULL, color VARCHAR(10) NOT NULL, INDEX IDX\_2E13599D29C1004E (video\_id), PRIMARY KEY(id)) DEFAULT CHARACTER SET utf8mb4 COLLATE `utf8mb4\_unicode\_ci` ENGINE = InnoDB');

        $this->addSql('CREATE TABLE `user` (id INT AUTO\_INCREMENT NOT NULL, image\_id INT DEFAULT NULL, document\_id INT DEFAULT NULL, country\_id INT DEFAULT NULL, email VARCHAR(180) NOT NULL, roles JSON NOT NULL, password VARCHAR(255) NOT NULL, confirmation\_token VARCHAR(255) NOT NULL, is\_paid TINYINT(1) NOT NULL, name VARCHAR(50) NOT NULL, first\_name VARCHAR(50) NOT NULL, birthday DATE NOT NULL, phone\_number VARCHAR(20) NOT NULL, society VARCHAR(255) NOT NULL, is\_mail\_verified TINYINT(1) NOT NULL, status VARCHAR(10) NOT NULL, created\_at DATETIME DEFAULT NULL, updated\_at DATETIME DEFAULT NULL, UNIQUE INDEX UNIQ\_8D93D649E7927C74 (email), INDEX IDX\_8D93D6493DA5256D (image\_id), INDEX IDX\_8D93D649C33F7837 (document\_id), INDEX IDX\_8D93D649F92F3E70 (country\_id), PRIMARY KEY(id)) DEFAULT CHARACTER SET utf8mb4 COLLATE `utf8mb4\_unicode\_ci` ENGINE = InnoDB');

        $this->addSql('CREATE TABLE video (id INT AUTO\_INCREMENT NOT NULL, title VARCHAR(255) NOT NULL, keywords JSON NOT NULL, short\_description LONGTEXT NOT NULL, view\_count INT DEFAULT NULL, url VARCHAR(255) NOT NULL, path VARCHAR(255) NOT NULL, video\_youtube\_id VARCHAR(255) NOT NULL, author VARCHAR(255) DEFAULT NULL, UNIQUE INDEX UNIQ\_7CC7DA2C3BD95867 (video\_youtube\_id), PRIMARY KEY(id)) DEFAULT CHARACTER SET utf8mb4 COLLATE `utf8mb4\_unicode\_ci` ENGINE = InnoDB');

        $this->addSql('ALTER TABLE analyse ADD CONSTRAINT FK\_351B0C7E6B301384 FOREIGN KEY (capture\_id) REFERENCES capture (id)');

        $this->addSql('ALTER TABLE analyse ADD CONSTRAINT FK\_351B0C7EF675F31B FOREIGN KEY (author\_id) REFERENCES `user` (id)');

        $this->addSql('ALTER TABLE analyse ADD CONSTRAINT FK\_351B0C7E7C4D497E FOREIGN KEY (sujet\_id) REFERENCES sujet (id)');

        $this->addSql('ALTER TABLE capture ADD CONSTRAINT FK\_8BFEA6E529C1004E FOREIGN KEY (video\_id) REFERENCES video (id)');

        $this->addSql('ALTER TABLE item ADD CONSTRAINT FK\_1F1B251E727ACA70 FOREIGN KEY (parent\_id) REFERENCES item (id)');

        $this->addSql('ALTER TABLE item\_secondaire ADD CONSTRAINT FK\_CC2A70951EFE06BF FOREIGN KEY (analyse\_id) REFERENCES analyse (id)');

        $this->addSql('ALTER TABLE password\_token ADD CONSTRAINT FK\_BEAB6C24A76ED395 FOREIGN KEY (user\_id) REFERENCES `user` (id)');

        $this->addSql('ALTER TABLE referencement ADD CONSTRAINT FK\_83A125FE6B301384 FOREIGN KEY (capture\_id) REFERENCES capture (id)');

        $this->addSql('ALTER TABLE referencement ADD CONSTRAINT FK\_83A125FE7C4D497E FOREIGN KEY (sujet\_id) REFERENCES sujet (id)');

        $this->addSql('ALTER TABLE role\_user ADD CONSTRAINT FK\_332CA4DDD60322AC FOREIGN KEY (role\_id) REFERENCES role (id) ON DELETE CASCADE');

        $this->addSql('ALTER TABLE role\_user ADD CONSTRAINT FK\_332CA4DDA76ED395 FOREIGN KEY (user\_id) REFERENCES `user` (id) ON DELETE CASCADE');

        $this->addSql('ALTER TABLE sujet ADD CONSTRAINT FK\_2E13599D29C1004E FOREIGN KEY (video\_id) REFERENCES video (id)');

        $this->addSql('ALTER TABLE `user` ADD CONSTRAINT FK\_8D93D6493DA5256D FOREIGN KEY (image\_id) REFERENCES media\_object\_image (id)');

        $this->addSql('ALTER TABLE `user` ADD CONSTRAINT FK\_8D93D649C33F7837 FOREIGN KEY (document\_id) REFERENCES media\_object\_document (id)');

        $this->addSql('ALTER TABLE `user` ADD CONSTRAINT FK\_8D93D649F92F3E70 FOREIGN KEY (country\_id) REFERENCES country (id)');

    }

**public** **function** down(Schema $schema): void

    {

*// this down() migration is auto-generated, please modify it to your needs*

        $this->addSql('ALTER TABLE item\_secondaire DROP FOREIGN KEY FK\_CC2A70951EFE06BF');

        $this->addSql('ALTER TABLE analyse DROP FOREIGN KEY FK\_351B0C7E6B301384');

        $this->addSql('ALTER TABLE referencement DROP FOREIGN KEY FK\_83A125FE6B301384');

        $this->addSql('ALTER TABLE `user` DROP FOREIGN KEY FK\_8D93D649F92F3E70');

        $this->addSql('ALTER TABLE item DROP FOREIGN KEY FK\_1F1B251E727ACA70');

        $this->addSql('ALTER TABLE `user` DROP FOREIGN KEY FK\_8D93D649C33F7837');

        $this->addSql('ALTER TABLE `user` DROP FOREIGN KEY FK\_8D93D6493DA5256D');

        $this->addSql('ALTER TABLE role\_user DROP FOREIGN KEY FK\_332CA4DDD60322AC');

        $this->addSql('ALTER TABLE analyse DROP FOREIGN KEY FK\_351B0C7E7C4D497E');

        $this->addSql('ALTER TABLE referencement DROP FOREIGN KEY FK\_83A125FE7C4D497E');

        $this->addSql('ALTER TABLE analyse DROP FOREIGN KEY FK\_351B0C7EF675F31B');

        $this->addSql('ALTER TABLE password\_token DROP FOREIGN KEY FK\_BEAB6C24A76ED395');

        $this->addSql('ALTER TABLE role\_user DROP FOREIGN KEY FK\_332CA4DDA76ED395');

        $this->addSql('ALTER TABLE capture DROP FOREIGN KEY FK\_8BFEA6E529C1004E');

        $this->addSql('ALTER TABLE sujet DROP FOREIGN KEY FK\_2E13599D29C1004E');

        $this->addSql('DROP TABLE analyse');

        $this->addSql('DROP TABLE capture');

        $this->addSql('DROP TABLE country');

        $this->addSql('DROP TABLE item');

        $this->addSql('DROP TABLE item\_secondaire');

        $this->addSql('DROP TABLE media\_object\_document');

        $this->addSql('DROP TABLE media\_object\_image');

        $this->addSql('DROP TABLE password\_token');

        $this->addSql('DROP TABLE referencement');

        $this->addSql('DROP TABLE refresh\_tokens');

        $this->addSql('DROP TABLE role');

        $this->addSql('DROP TABLE role\_user');

        $this->addSql('DROP TABLE sujet');

        $this->addSql('DROP TABLE `user`');

        $this->addSql('DROP TABLE video');

    }

}

### Codage de d’application

Pour tirer le meilleur de symfony, on a utilisé un célèbre bundle de symfony pour le webservice. Ceci nous force à utiliser le système d’annotation et ainsi personnaliser l’api à notre guise. Voici quelque ligne de code avec quoi toute est centré dans le modèle. Tout s’écrit dans le modèle avec le système d’annotation.

#### Code modele

**<?php**

**namespace** App\Entity;

**use** ApiPlatform\Core\Annotation\ApiFilter;

**use** ApiPlatform\Core\Bridge\Doctrine\Orm\Filter\SearchFilter;

**use** ApiPlatform\Core\Annotation\ApiResource;

**use** App\Repository\AnalyseRepository;

**use** Doctrine\Common\Collections\ArrayCollection;

**use** Doctrine\Common\Collections\Collection;

**use** Doctrine\ORM\Mapping **as** ORM;

**use** App\Controller\LastestTenAnalysesRecentesController;

**use** Symfony\Component\Serializer\Annotation\Groups;

**use** App\Filter\SearchAnnotation **as** Searchable;

*/\*\**

*\* @ApiResource(*

*\*     mercure="true",*

*\*     collectionOperations={*

*\*         "get"={*

*\*                  "security"="is\_granted('ROLE\_CLIENT')",*

*\*                  "normalization\_context"={"groups"={"analyse:read"}}*

*\*          },*

*\*         "get\_lastest\_ten" = {*

*\*              "method" = "GET",*

*\*              "path" = "/analyses/recents",*

*\*              "controller" = LastestTenAnalysesRecentesController::class*

*\*         },*

*\*         "post"={"security"="is\_granted('ROLE\_CLIENT')"}*

*\*     },*

*\*     itemOperations={*

*\*         "get" = {"security\_post\_denormalize"="is\_granted('ROLE\_ADMIN') or object.author == user"},*

*\*         "put"={"security"="is\_granted('ROLE\_ADMIN') or object.author == user"},*

*\*         "delete"={"security"="is\_granted('ROLE\_CLIENT')"},*

*\*     }*

*\* )*

*\**

*\* @ORM\Entity(repositoryClass="App\Repository\AnalyseRepository", repositoryClass=AnalyseRepository::class)*

*\* @ORM\HasLifecycleCallbacks()*

*\* @ApiFilter(SearchFilter::class, properties={*

*\*     "sujetAnalyse": "partial",*

*\*     "code":"partial",*

*\*     "verbalePrononce": "partial",*

*\*     "observation": "partial",*

*\*     "author.name": "partial",*

*\*     "author.firstName":"partial"})*

*\* @Searchable({"sujetAnalyse", "code", "email", "verbalePrononce", "observation", "author.name", "author.firstName"})*

*\*/*

**class** Analyse

{

*/\*\**

*\* @ORM\Id*

*\* @ORM\GeneratedValue*

*\* @ORM\Column(type="integer")*

*\*/*

**private** $id;

*/\*\**

*\* @ORM\Column(type="string", length=255, nullable=true)*

*\* @Groups("analyse:read")*

*\*/*

**private** $pointeur;

*/\*\**

*\* @ORM\Column(type="string", length=255, nullable=true)*

*\* @Groups("analyse:read")*

*\*/*

**private** $sujetAnalyse;

*/\*\**

*\* @ORM\Column(type="string", length=255, nullable=true)*

*\* @Groups("analyse:read")*

*\*/*

**private** $verbalePrononce;

*/\*\**

*\* @ORM\Column(type="text", nullable=true)*

*\* @Groups("analyse:read")*

*\*/*

**private** $observation;

*/\*\**

*\* @ORM\Column(type="string", length=255, nullable=true)*

*\* @Groups("analyse:read")*

*\*/*

**private** $code;

*/\*\**

*\* @ORM\Column(type="text", nullable=true)*

*\* @Groups("analyse:read")*

*\*/*

**private** $description;

*/\*\**

*\* @ORM\ManyToOne(targetEntity=Capture::class, inversedBy="analyses")*

*\* @ORM\JoinColumn(nullable=false)*

*\* @Groups({"items:read","analyse:read"})*

*\*/*

**private** $capture;

*/\*\**

*\* @ORM\ManyToOne(targetEntity=User::class, inversedBy="analyses")*

*\* @ORM\JoinColumn(nullable=false)*

*\*/*

**public** $author;

*/\*\**

*\* @ORM\Column(type="boolean")*

*\*/*

**private** $status;

*/\*\**

*\* @ORM\OneToMany(targetEntity=ItemSecondaire::class, mappedBy="analyse", orphanRemoval=true)*

*\*/*

**private** $itemSecondaires;

**public** **function** **\_\_construct**()

    {

        $this->itemSecondaires = **new** ArrayCollection();

    }

*/\*\**

*\* @ORM\ManyToOne(targetEntity=Sujet::class, inversedBy="analyses")*

*\*/*

**private** $sujet;

*/\*\**

*\* @ORM\Column(type="string", length=255, nullable=true)*

*\* @Groups("analyse:read")*

*\*/*

**private** $typeEmeteur;

**public** **function** getId(): ?int

    {

**return** $this->id;

    }

**public** **function** getPointeur(): ?string

    {

**return** $this->pointeur;

    }

**public** **function** setPointeur(?string $pointeur): self

    {

        $this->pointeur = $pointeur;

**return** $this;

    }

**public** **function** getSujetAnalyse(): ?string

    {

**return** $this->sujetAnalyse;

    }

**public** **function** setSujetAnalyse(?string $sujetAnalyse): self

    {

        $this->sujetAnalyse = $sujetAnalyse;

**return** $this;

    }

**public** **function** getVerbalePrononce(): ?string

    {

**return** $this->verbalePrononce;

    }

**public** **function** setVerbalePrononce(?string $verbalePrononce): self

    {

        $this->verbalePrononce = $verbalePrononce;

**return** $this;

    }

**public** **function** getObservation(): ?string

    {

**return** $this->observation;

    }

**public** **function** setObservation(?string $observation): self

    {

        $this->observation = $observation;

**return** $this;

    }

**public** **function** getCode(): ?string

    {

**return** $this->code;

    }

**public** **function** setCode(?string $code): self

    {

        $this->code = $code;

**return** $this;

    }

**public** **function** getDescription(): ?string

    {

**return** $this->description;

    }

**public** **function** setDescription(?string $description): self

    {

        $this->description = $description;

**return** $this;

    }

**public** **function** getCapture(): ?capture

    {

**return** $this->capture;

    }

**public** **function** setCapture(?capture $capture): self

    {

        $this->capture = $capture;

**return** $this;

    }

**public** **function** getAuthor(): ?User

    {

**return** $this->author;

    }

**public** **function** setAuthor(?User $author): self

    {

        $this->author = $author;

**return** $this;

    }

**public** **function** getStatus(): ?bool

    {

**return** $this->status;

    }

**public** **function** setStatus(bool $status): self

    {

        $this->status = $status;

**return** $this;

    }

*/\*\**

*\* @return Collection|ItemSecondaire[]*

*\*/*

**public** **function** getItemSecondaires(): Collection

    {

**return** $this->itemSecondaires;

    }

**public** **function** addItemSecondaire(ItemSecondaire $itemSecondaire): self

    {

**if** (!$this->itemSecondaires->contains($itemSecondaire)) {

            $this->itemSecondaires[] = $itemSecondaire;

            $itemSecondaire->setAnalyse($this);

        }

**return** $this;

    }

*/\*\**

*\* @ORM\PrePersist()*

*\*/*

**public** **function** defaultStatus()

    {

        $this->status = **false**;

    }

**public** **function** removeItemSecondaire(ItemSecondaire $itemSecondaire): self

    {

**if** ($this->itemSecondaires->removeElement($itemSecondaire)) {

*// set the owning side to null (unless already changed)*

**if** ($itemSecondaire->getAnalyse() === $this) {

                $itemSecondaire->setAnalyse(**null**);

            }

        }

**return**  $this;

    }

**public** **function** getSujet(): ?Sujet

    {

**return** $this->sujet;

    }

**public** **function** setSujet(?Sujet $sujet): self

    {

        $this->sujet = $sujet;

**return** $this;

    }

**public** **function** getTypeEmeteur(): ?string

    {

**return** $this->typeEmeteur;

    }

**public** **function** setTypeEmeteur(?string $typeEmeteur): self

    {

        $this->typeEmeteur = $typeEmeteur;

**return** $this;

    }

}

#### Code du controller

Comme on utilise des annotations, nos controleur ne sera composer que d’une seule méthode qui sera la méthode \_\_invoke()

Voici le code de notre controller youtubeDownloader

**<?php**

**namespace** App\Controller;

**use** App\Entity\Video;

**use** App\Repository\VideoRepository;

**use** Symfony\Component\Filesystem\Filesystem;

**use** Symfony\Component\HttpClient\CurlHttpClient;

**use** Symfony\Component\HttpFoundation\JsonResponse;

**use** Symfony\Component\HttpFoundation\Request;

**use** Symfony\Component\HttpFoundation\Response;

**use** YouTube\YouTubeDownloader **as** Yd;

**use** Symfony\Bundle\FrameworkBundle\Controller\AbstractController;

**use** Cocur\Slugify\Slugify;

**class** YoutubeDownloader **extends** AbstractController

{

*/\*\**

*\* @var Yd*

*\*/*

**private** $youtubeDownloader;

*/\*\**

*\* @var VideoRepository*

*\*/*

**private** VideoRepository $videoRepository;

**public** **function** **\_\_construct**(Yd $youtubeDownloader, VideoRepository $videoRepository)

    {

        $this->youtubeDownloader = $youtubeDownloader;

        $this->videoRepository = $videoRepository;

    }

**public** **function** \_\_invoke(Request $request)

    {

        $video = **new** Video();

        $video = $request->attributes->get("data");

        $downloadOptions = $this->youtubeDownloader->getDownloadLinks($video->getUrl());

**if** ($this->videoRepository->findByVideoYoutubeId($downloadOptions->getInfo()->getId())) {

**return** $this->videoRepository->findOneByVideoYoutubeId($downloadOptions->getInfo()->getId());

        } **else** {

**try** {

                $client = **new** CurlHttpClient();

                $videoUrl='';

                $audioUrl='';

                $slugify = **new** Slugify();

                $keyword = [];

**if** (!**empty**($downloadOptions->getInfo()->getKeywords())) {

                    $video->setKeywords($downloadOptions->getInfo()->getKeywords());

                } **else** {

                    $video->setKeywords($keyword);

                }

                $video->setTitle($downloadOptions->getInfo()->getTitle())

                    ->setvideoYoutubeId($downloadOptions->getInfo()->getId())

                    ->setShortDescription($downloadOptions->getInfo()->getShortDescription());

                $hightEstVideo = $this->getHightToHighVideoFormats($downloadOptions->getVideoFormats());

**foreach** ($hightEstVideo **as** $key => $value) {

                    $videoUrl = $value->qualityLabel == "1080p" ? $value->url : $videoUrl;

**if** (!$videoUrl) {

                        $videoUrl = $value->qualityLabel == "720p" ? $value->url : $videoUrl;

                    } **else** {

**break**;

                    }

                }

                $hightEstAudio = $this->getHightToHighAudioFormats($downloadOptions->getAudioFormats());

**foreach** ($hightEstAudio **as** $key => $value) {

                    $audioUrl = $value->audioQuality == "AUDIO\_QUALITY\_MEDIUM" ? $value->url : $audioUrl;

**if** ($audioUrl) {

**break**;

                    }

                }

                $audioFileName = uniqid(rand(), **true**). '.mp3';

                $videoFileName = uniqid(rand(), **true**). '.mp4';

                $videoOutputFileName = $slugify->slugify($downloadOptions->getInfo()->getTitle()).'.mp4';

                $video->setPath($this->getParameter('videos\_directory\_public') .'/'.$videoOutputFileName);

                $audio = $client->request('POST', $audioUrl)->getContent();

                file\_put\_contents($this->getParameter('videos\_directory') . '/'.$audioFileName, $audio);

                $videoFile = $client->request('POST', $videoUrl)->getContent();

                file\_put\_contents($this->getParameter('videos\_directory') . '/'.$videoFileName, $videoFile);

                exec('ffmpeg -i '.$this->getParameter('videos\_directory') .'/'.$videoFileName.' -i '.$this->getParameter('videos\_directory') .'/'.$audioFileName.' -c:v copy -c:a aac '.$this->getParameter('videos\_directory') .'/'.$videoOutputFileName);

                $entityManager = $this->getDoctrine()->getManager();

                $entityManager->persist($video);

                $entityManager->flush();

                $filesystem = **new** Filesystem();

                $filesystem->remove($this->getParameter('videos\_directory') . '/'.$videoFileName);

                $filesystem->remove($this->getParameter('videos\_directory') . '/'.$audioFileName);

            } **catch** (YouTubeException $e) {

**echo** 'Something went wrong: ' . $e->getMessage();

            }

**return** $video;

        }

    }

**private** **function** getHightToHighVideoFormats($videoFormat)

    {

        $copy = array\_values($videoFormat);

        usort($copy, **function** ($a, $b) {

*/\*\* @var StreamFormat $a \*/*

*/\*\* @var StreamFormat $b \*/*

**return** $b->height - $a->height;

        });

**return** $copy;

    }

**private** **function** getHightToHighAudioFormats($audio)

    {

        $copy = array\_values($audio);

*// just assume higher filesize => higher quality...*

        usort($copy, **function** ($a, $b) {

*/\*\* @var StreamFormat $a \*/*

*/\*\* @var StreamFormat $b \*/*

**return** $b->contentLength - $a->contentLength;

        });

**return** $copy;

    }

}

### Présentation de l’application

* La page d’authentification

La figure 42 montre la page d’authentification dans la plateforme

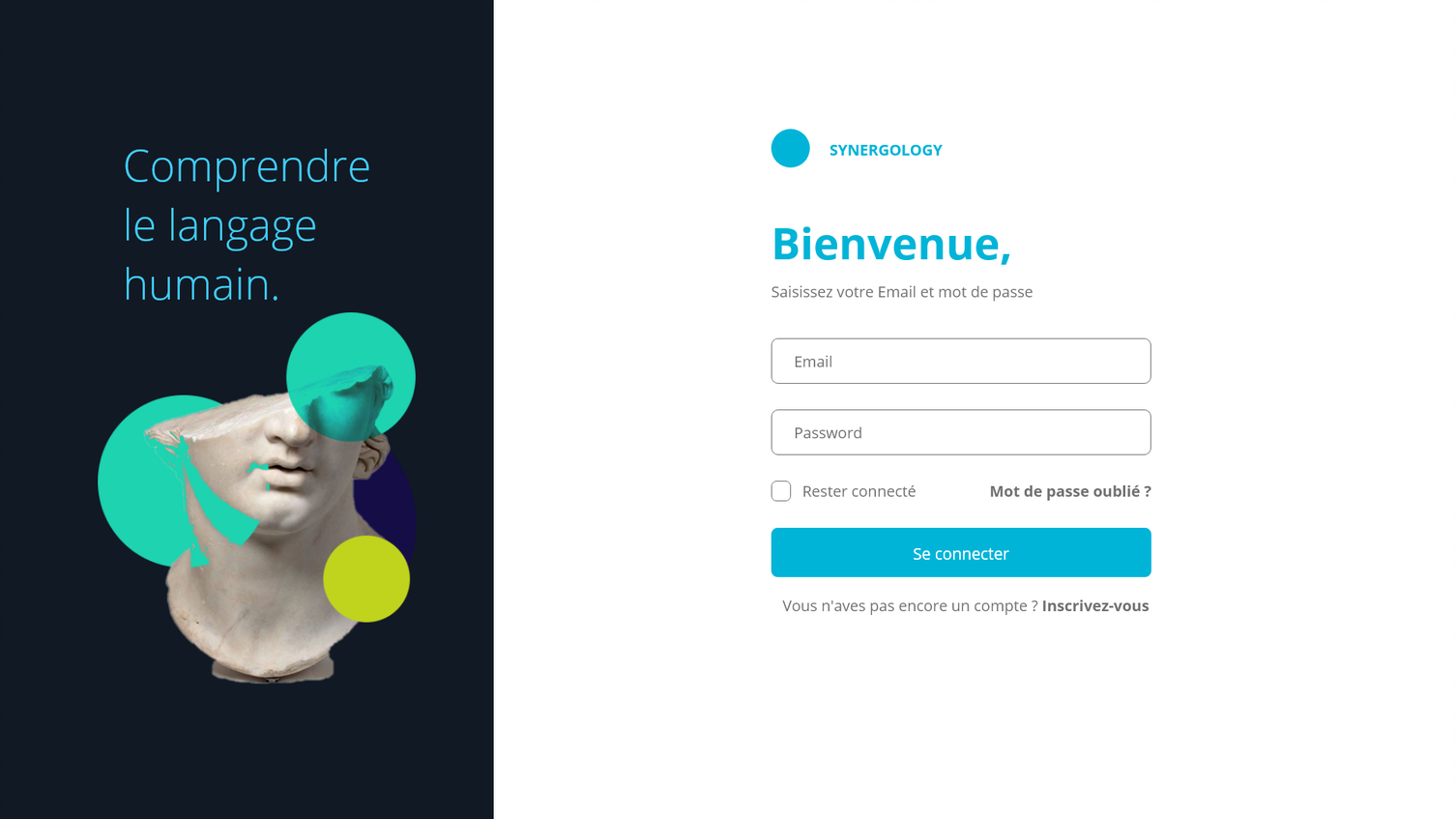


Figure 42 Page d'authentification

* La page de gestion des utilisateurs

La figure 43 montre la page de gestion des utilisateurs

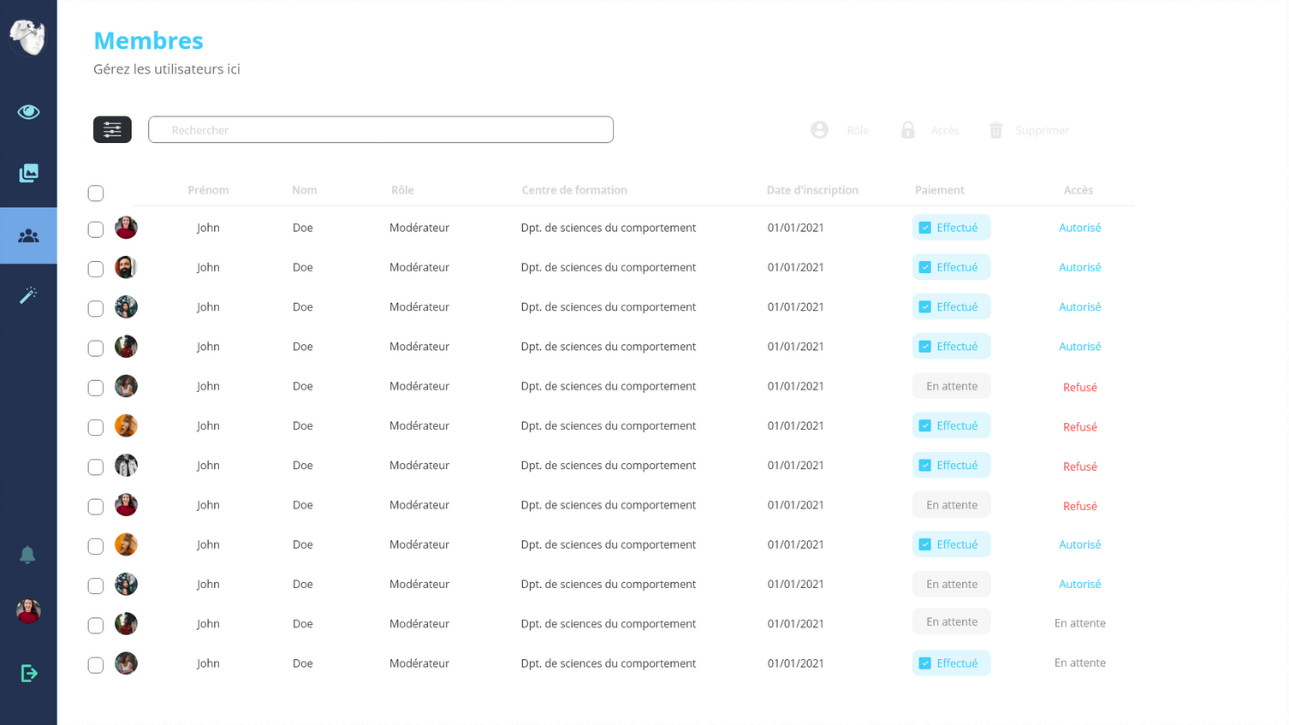


Figure 43 Page de gestion d'utilisateur

* Page de galerie

La figure 44 montre la page galerie photo

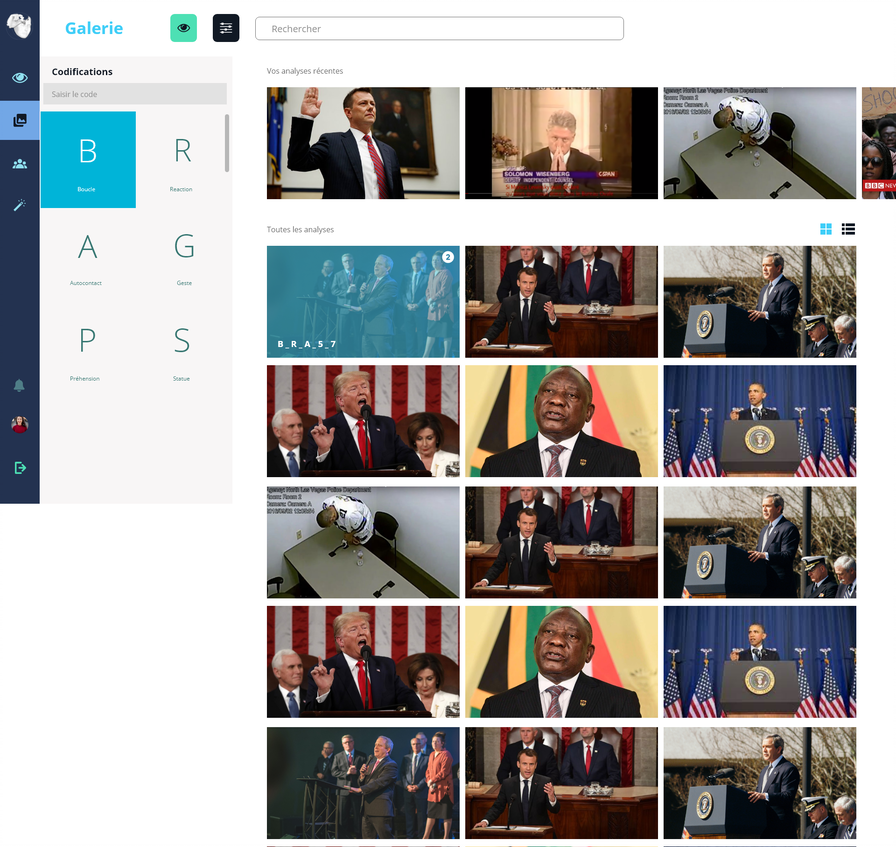


Figure 44 page de galerie photo

* La page de gestion de codification

La figure 45 montre la gestion de codification

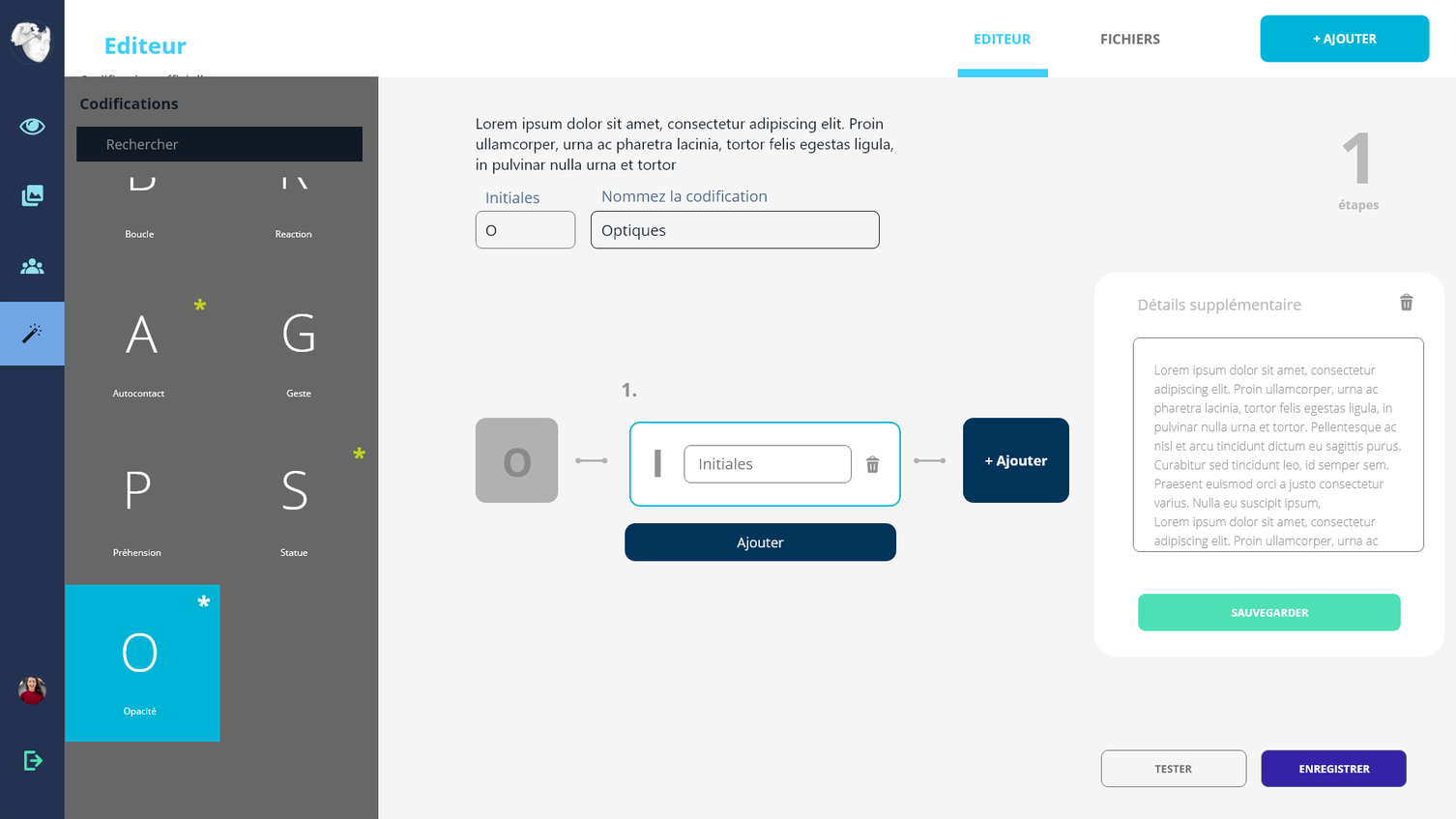


Figure 45 page de gestion de codification

* Page de gestion des analyses d’item secondaire

La figure 46 montre la gestion d’analyse avec item secondaire

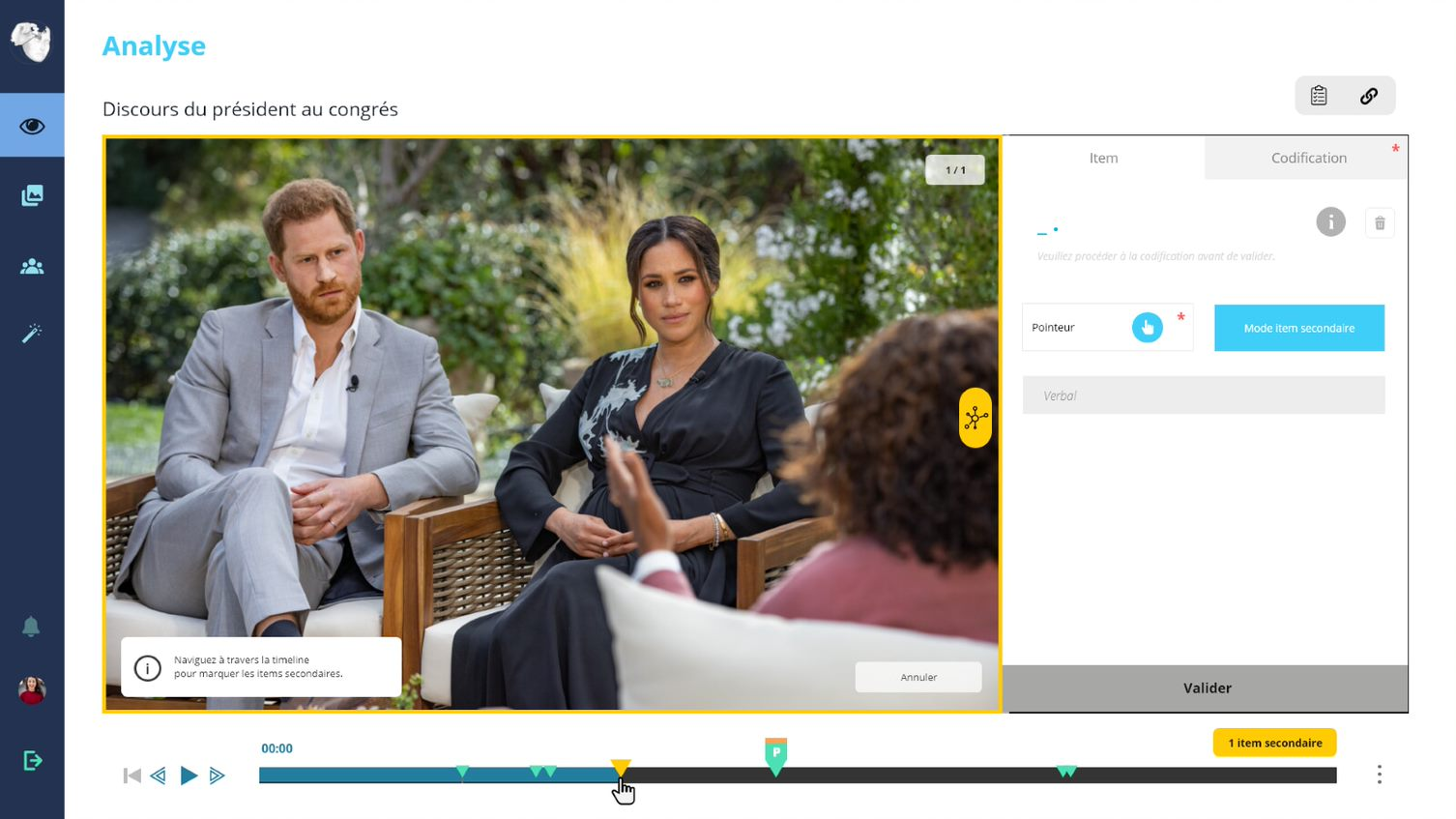


Figure 46 page d'item secondaire

CONCLUSION

Ce stage s’est déroulé au sein de la société Moralogiq. Le présent logiciel a été réalisé en se basant sur le pratique habituel de l’institut européen de synergologie, pour réaliser le projet d’associer leur pratique aux nouvelles technologies de l’information et de la communication, moralogiq nous a recrutés en tant que stagiaire. Rappelons que l'objectif du projet est de faire développer une application web pour la gestion de la relation humaine par communication non verbale. L’application permet en général de faire une analyse d’une vidéo en se focalisant sur le non verbal et en utilisation une base de données en arbre qui existe déjà. Elle est capable de naviguer dans l’arbre et concaténer les significations correspondantes et ainsi d’obtenir un rapport.

Pour sa réalisation, on a utilisé 2TUP comme méthode de conception, PHP comme langage de programmation, SCRUM pour la gestion de projet. Pour l’hébergement, on a utilisé ngnix et en ce qui concerne la SGBD, on a utilisé MySQL et pour la conception du système, on a utilisé UML avec visual paradigm pour les différents diagrammes.

Cette application est opérationnelle et répond au besoin du client. Néanmoins dans la perspective, on envisage de mettre à disposition des recruteurs ce puissant outil afin de les aider à recruter des vrais talents lors des recrutements. Identifier les menteurs parce que de nos jours, les CV sont boostés par les candidats et ainsi trouvé une personne pour bien investir.

Ce stage nous a permis de nous familiariser avec le mode professionnel et d’appliquer les connaissances acquises à l’ENI. Elle nous a fait vivre une expérience formidable et importante dans le mode du développement.

BIBLIOGRAPHIE

[1] P. Kruchten, 2000, Introduction au Rational Unified Process, Edition EYROLLES.

[2] Pascal ROQUES, 2008, Les Cahiers du programmeur UML2 – Modéliser une application, Edition EYROLLES, 4è édition, 264 pages.

[3] F. Juliard UML Unified Method Language, Journal Univerdité de Bretagne Sud UFR SSI-IUP Vannes, 2001-2002.

[4] Taha M’HAND, 2009, Modélisation Conceptuelle BD 2, 23 pages

[5] Bacco, B. (2018). Développez votre site Web avec le framework Symfony 4. Eyrolles.

[6] Saliha, Y. (2013). User-stories et Backlog de produit.

[7] Foundikou, V. (2005). Modèle Conceptuelle orientée Objet, Eyrolles.

[8] Piechoki, L. (2007). UML, le langage de modélisation Objet unifié. Corel.

[9] Voisin, F. (2014). Les diagrammes de séquence en phase de conception. Polytech ParisSud

[10] Roques, P. Vallée, F. (2007). UML2 en action. Eyrolles.

[11] Aubrey, C. (2011). SCRUM le guide pratique de la méthode agile le plus populaire. Dunod. [12] Laurent AUDIBERT, 2007, UML 2, Edition 2007-2008, 178 pages

[13] GABAY J. et GABAY D., DUNOD, 2008, « UML 2 Analyse et conception », page 2 et 46.

[14] Joseph Gabay et David Gabay, Mise en œuvre guidée avec études de cas, édition Dunod, Paris 2008

[15] ROQUES P., EYROLLES, 2006. « UML 2 par la pratique », page 76 et 118.

WEBOGRAPHIE

[16] <http://laurent-audibert.developpez.com/Cours-UML/>, consulté le 15 Avril 2021

[17] <http://fr.wikipedia.org/wiki/Diagramme_sequence/>, définition de diagramme de séquence, 15 avril 2021

[18] <http://www.additeam.com/SSII/systeme-d%E2%80%99information-si/>, définition de Système consulté le 3 Mai 2021

[19] <http://www.iro.umontreal.ca/~pift3901/Slides/A05-S05-ModDom.pdf>, Modèle du

Domaine, consulté en 4 Mai 2021

[20] <http://book.cakephp.org/2.0/fr/cakephp-overview/understanding-modelviewcontroller.html>, Comprendre le système MVC (Model-View-Controller) consulté

le 6 Mai 2021

[21] <http://www.letudiant.fr/etudes/tout-pour-reussir-son-memoire-son-rapport-destage.html>, Tout pour réussir son mémoire, son rapport de stage, Consulté le 1er Juin 2021.

[21] <http://www.letudiant.fr/etudes/tout-pour-reussir-son-memoire-son-rapport-destage.html>, Tout pour réussir son mémoire, son rapport de stage, Consulté le 1er Juin 2021.

[22] <https://www.journalducm.com/trello-guide-utilisation/>, Trello, consulté le 3 Mars 2021

[23] <https://fr.wikipedia.org/wiki/Logiciel_de_gestion_de_versions>, Logiciel de gestion de versions, consulté le 05 Juin 2021

[24] <https://fr.wikipedia.org/wiki/Langage_de_programmation>, Langage de programmation, consulté le 24 Mai 2021

[25] <https://blog.lws-hosting.com/conseils-marketing/symfony-zend-cakephp-codeigniterlequelchoisir>,consulté le 21 Juin 2021

[26] <https://ineumann.developpez.com/tutoriels/alm/agile_scrum/>, Présentation des méthodes agiles et Scrum, consulté le 5 Juillet 2021

[27] <http://bahit.fr/prod/2018/06/11/la-methode-agile-scrum/>, La méthode SCRUM AGILE, consulté le 5 Juillet 2021

GLOSSAIRE

**Codification :** Ecrire de code de façon logique et relationnel.

**Synergologie** : Marque de commerce relative à l’étude de la communication non verbale, des messages transmis par les mouvements corporels inconscients.

**Item secondaire** : Geste qui se relie de façon logique mais dans 2 temps différents.

**Sujet** : Personne qui apparait dans une vidéo.

**Architecte**: rôle créateur de code et d’item secondaire.

ANNEXES

TABLE DES MATIERES

[CURRICULUM VITAE I](#_Toc81804834)

[SOMMAIRE GENERAL IV](#_Toc81804835)

[REMERCIEMENTS VII](#_Toc81804836)

[LISTE DES FIGURES VIII](#_Toc81804837)

[LISTE DES TABLEAUX X](#_Toc81804838)

[LISTE DES ABREVIATIONS XI](#_Toc81804839)

[INTRODUCTION GENERALE 1](#_Toc81804840)

[PARTIE I : PRESENTATIONS 2](#_Toc81804841)

[Chapitre 1 Présentation de l’Ecole Nationale d’Informatique 3](#_Toc81804842)

[1.1. Information d’ordre générale 3](#_Toc81804843)

[1.2. Missions et historiques 3](#_Toc81804844)

[1.3. Organigramme institutionnel de l’ENI 5](#_Toc81804845)

[1.4. Domaines et spécialisation 6](#_Toc81804846)

[1.5. Architecture des formations pédagogiques 7](#_Toc81804847)

[1.6. Relation de l’ENI avec les entreprises et organismes 9](#_Toc81804848)

[1.7. Partenariat au niveau international 10](#_Toc81804849)

[1.8. Débouchés professionnels des diplômes 12](#_Toc81804850)

[1.9. Ressources humaines 14](#_Toc81804851)

[Chapitre 2 Présentation de MORALOGIQ 15](#_Toc81804852)

[2.1. Historique et missions 15](#_Toc81804853)

[2.2. Situation juridique 15](#_Toc81804854)

[2.3. Organigramme 15](#_Toc81804855)

[2.4. Services 15](#_Toc81804856)

[2.5. Contacts 16](#_Toc81804857)

[Chapitre 3 Description du projet 17](#_Toc81804858)

[3.1. Formulation 17](#_Toc81804859)

[3.2. Objectif et besoins d’utilisateur 17](#_Toc81804860)

[3.2.1. Objectif du projet 17](#_Toc81804861)

[3.2.2. Besoin de l’utilisateur 17](#_Toc81804862)

[3.3. Moyen nécessaire à la réalisation du projet 17](#_Toc81804863)

[3.3.1. Moyens humains 17](#_Toc81804864)

[3.3.2. Moyens matériels 19](#_Toc81804865)

[3.3.3. Moyens logiciels 20](#_Toc81804866)

[3.4. Résultats attendus 20](#_Toc81804867)

[3.5. Chronogramme des Travaux 21](#_Toc81804868)

[PARTIE II : ANALYSE ET CONCEPTION 22](#_Toc81804869)

[Chapitre 4 Analyse préalable 23](#_Toc81804870)

[4.1. Analyse de l’existant 23](#_Toc81804871)

[4.1.1. Organisation actuelle 23](#_Toc81804872)

[4.1.2. Inventaire des moyens matériels et logiciels 23](#_Toc81804873)

[4.2. Critique de l’existant 23](#_Toc81804874)

[4.3. Conception avant-projet 24](#_Toc81804875)

[4.3.1. Comparaison de solutions 24](#_Toc81804876)

[4.3.2. Choix et justification 25](#_Toc81804877)

[Chapitre 5 Analyse conceptuelle 45](#_Toc81804878)

[5.1. Étapes de l’élaboration du backlog de produit 45](#_Toc81804879)

[5.1.1. Vision globale du projet 45](#_Toc81804880)

[5.1.2. Liste détaillée des acteurs avec fiche 45](#_Toc81804881)

[5.1.3. Thème et regroupement de fonctionnalités 48](#_Toc81804882)

[5.1.4. Vision du premier release 57](#_Toc81804883)

[5.1.5. User stories 57](#_Toc81804884)

[5.1.6. Backlog de produit 58](#_Toc81804885)

[5.2. Dictionnaire des données 59](#_Toc81804886)

[5.3. Règle de gestion 64](#_Toc81804887)

[5.4. Capture des besoins fonctionnels 65](#_Toc81804888)

[5.5. Analyse 65](#_Toc81804889)

[5.5.1. Diagramme de cas d’utilisation 65](#_Toc81804890)

[5.5.2. Priorisation et description des cas d’utilisation 68](#_Toc81804891)

[5.6. Capture des besoins techniques 72](#_Toc81804892)

[5.7. Conception préliminaire 72](#_Toc81804893)

[5.7.1. Présentation de la méthode utilisée 72](#_Toc81804894)

[5.7.2. Diagrammes des séquences système 77](#_Toc81804895)

[5.7.3. Modélisation du domaine 84](#_Toc81804896)

[Chapitre 6 Conception détaillée 85](#_Toc81804897)

[6.1. Architecture du système 85](#_Toc81804898)

[6.2. Diagramme de séquence de conception 87](#_Toc81804899)

[6.3. Diagramme de classe de conception 88](#_Toc81804900)

[6.4. Diagramme de classe de conception global 95](#_Toc81804901)

[6.5. Diagramme de paquetage 95](#_Toc81804902)

[6.6. Diagramme de déploiements 96](#_Toc81804903)

[PARTIE III : REALISATION 97](#_Toc81804904)

[Chapitre 7 Mise en place de l’environnement de développement 98](#_Toc81804905)

[7.1. Installation et configuration des outils 98](#_Toc81804906)

[7.1.1. Visual Paradigm sur windows 98](#_Toc81804907)

[7.1.2. Installation de php7.4, phpMyAdmin, MySQL(MariaDB) sous Ubuntu 20 98](#_Toc81804908)

[7.1.3. PHPStorm 99](#_Toc81804909)

[7.2. Architecture de l’application 100](#_Toc81804910)

[Chapitre 8 Développement de l’application 101](#_Toc81804911)

[8.1. Création de la base de données 101](#_Toc81804912)

[8.2. Codage de d’application 106](#_Toc81804913)

[8.2.1. Code modele 106](#_Toc81804914)

[8.2.2. Code du controller 114](#_Toc81804915)

[8.3. Présentation de l’application 117](#_Toc81804916)

[CONCLUSION XI](#_Toc81804917)

[BIBLIOGRAPHIE XII](#_Toc81804918)

[WEBOGRAPHIE XIII](#_Toc81804919)

[GLOSSAIRE XIV](#_Toc81804920)

[ANNEXES XV](#_Toc81804921)

[TABLE DES MATIERES XVI](#_Toc81804922)

[RESUME XX](#_Toc81804923)

[ABSTRACT XX](#_Toc81804924)

RESUME

Dans ce présent projet, nous avons conçu et élaboré une application destinée à la gestion de la relation humaine Le projet consiste à mettre un système d’analyse de geste à partir d’une vidéo. Dans ce document se trouvent tous les détails concernés, depuis l’étude de l’existant jusqu’à la réalisation de l’application, comme la présentation des outils utilisés et la modélisation du système. Nous avons comparé les outils utilisés par rapport à d’autres et étudié l’architecture de l’application après avoir relevé les besoins de l’utilisateur avec la méthode 2TUP liée à la notation UML et on a géré le projet avec la méthode AGILE SCRUM. L’application c’est abouti avec une interface facile à manipuler et les fonctionnalités requises par la cellule que nous avons illustrée dans des captures d’écran.

**Mots-clés** : PHP, Framework, Symfony 4, analyse, vidéo, geste, modélisation, AGILE, SCRUM, UML, 2TUP.

ABSTRACT

In this present project, we have designed and developed, with the programming language PHP with its Symfony 4 Framework, an application intended for the analysis of human gestures. The project is to put a gesture analysis system from a video. In this document are all the details concerned, from the study of the existing to the realization of the application, such as the presentation of the tools used and the modeling of the system. We compared the tools used compared to others and studied the architecture of the application after having identified the user's needs with the 2TUP method linked to the UML notation and we managed the project with the AGILE SCRUM method. The application ended up with an easy-to-use interface and the functionality required by the cell that we illustrated in screenshots.

**Keywords** : PHP, Framework, Symfony 4, analysis, video, gestures, modeling, AGILE, SCRUM, UML, 2TUP.