2025

ADMISSION AND EMPLOYABILITY

SMART ADMIT

By DataInsight

esprit
Se former autrement

DataInsight Members

Sarra Trabelsi
Ayoub Sellami
Yassine Messai
Iheb Jounaidi
Aymen Bahdoudi

4 ERP - BI 7

esprit
Se former autrement

Acknowledgment

On behalf of the Data Insight team, we would like to sincerely thank our supervisors and the ESPRIT faculty for giving us the opportunity to work on the Admission Esprit project. It has been an enriching experience that allowed us to apply our skills to a meaningful and innovative challenge

SUMMARY

SUMMARY

INTRODUCTION

To become a leading institution in higher education, a school must excel not only in academic offerings but also in attracting top-performing students and ensuring their smooth integration into the professional world. At ESPRIT, this strategic vision is central to its mission. However, recent challenges have emerged that threaten this trajectory. The admission process has become increasingly fragmented and inefficient, while the connection between academic training and labor market outcomes has weakened. In parallel, the institution has observed a decline in the employability rate of its alumni and a regression in the overall quality of admitted students. These trends highlight the urgent need to rethink and modernize both the admission process and post-graduation follow-up. SmartAdmit was born as a response to these challenges. It aims to provide a smart, data-driven, and centralized platform that optimizes candidate evaluation and enhances alumni tracking, ultimately reinforcing ESPRIT's leadership position in the educational landscape.

PROBLEM STATEMENT

While the current admission process at ESPRIT is conducted through electronic submission, it still lacks automation in the selection and evaluation phases. Each application must be processed individually, which makes the system time-consuming, resource-intensive, and prone to inconsistencies. The absence of automated screening or intelligent ranking mechanisms limits efficiency and scalability, especially during peak admission periods. Moreover, ESPRIT does not have an integrated platform to systematically track alumni after graduation and that includes their employment status, industries of integration, and career progression. This results in a limited understanding of the real-world impact of ESPRIT's educational programs and weakens the school's ability to maintain long-term engagement with alumni and industry partners. These challenges are further compounded by a recent decline in the academic and professional quality of graduates, along with decreasing employment rates. Altogether, these issues point to the urgent need for a reimagined, data-driven approach to both admission management and alumni tracking.

“

How can we improve the university admission process to enhance employability and address the imbalance between supply and demand, unequal access, slow procedures, and the mismatch between academic programs and labor market needs?

EXISTING SOLUTIONS : STUDY AND CRITIQUE

Several commercial platforms caurrently address admission workflows or alumni tracking separately. Examples include CRM-based systems for admissions and platforms like LinkedIn Alumni or third-party career tracking services. However, most of these tools lack deep integration with the institution's internal databases and do not offer dynamic, AI-driven support for decision-making. Moreover, many of these solutions are built for large international universities with high implementation costs and rigid architectures that do not fit the local context or resource constraints of a school like ESPRIT. The gap between admission and employability tracking remains wide, resulting in siloed decision-making and missed opportunities for data-driven optimization.

EXISTING SOLUTIONS MAIN DISADVANTAGES :

Disadvantages:

- Limited consideration of individual student potential or employability.
- Lack of flexibility to adapt to labor market needs.
- Inequalities in access for students from different regions or backgrounds.
- Risk of saturation in certain programs and shortages in others.

PROPOSED SOLUTION: SMARTADMIT

SMARTADMIT

Smart Admissions. Smarter Futures.

"Select the future, automate excellence."

04

PROPOSED SOLUTION: SMARTADMIT

SmartAdmit is a unified, intelligent digital platform designed to optimize the full lifecycle of the student journey , from application to post-graduation. The system automates and enhances admission management using advanced sorting, filtering, and scoring algorithms based on machine learning, while also establishing a continuous channel for alumni data collection and engagement.

SMARTADMIT
Smart Admissions. Smarter Futures.

"Select the future, automate excellence."

KEY FEATURES INCLUDE:

- Centralized admission dashboard with real-time tracking.
- Intelligent profile matching and ranking based on customizable criteria.
- Integration with ESPRIT's internal systems and external labor market databases.
- Alumni follow-up system with automated surveys and employment tracking.
- Dynamic analytics dashboards to support strategic decision-making.

METHODOLOGY

The development of SmartAdmit followed an Agile methodology, allowing our team to remain flexible, responsive, and aligned with real user needs throughout the project lifecycle.

Our methodology unfolded across five key phases:

- Needs Assessment : We began by conducting a thorough analysis of ESPRIT's current admission process and alumni tracking mechanisms.
- Problem Definition : Based on the assessment, we defined the core problems: lack of automation in admission management, lack of data on alumni, and poor decision support tools for administrators. These findings shaped the functional priorities of SmartAdmit.
- Agile Design & Development : We structured the development process into sprints, each delivering incremental features. Using Agile principles, we held regular team meetings, reviewed sprint outcomes, and adjusted our backlog based on feedback. Key tools used included: Kanban boards for task tracking User stories to model administrator and student interactions Iterative testing to validate each feature quickly
- Testing & Validation : After each sprint, we conducted validation sessions simulating real admission workflows. These included: End-to-end testing of candidate scoring and filtering Verification of data visualization dashboards Testing of the alumni tracking module with dummy data
- Business and Strategic Alignment : Throughout the project, we ensured that the solution aligned with ESPRIT's strategic priorities: student quality, operational efficiency, and long-term employability. We developed a set of KPIs to evaluate impact and integrated a scalable design that can be adapted to other institutions in the future.

BUISNESS AND DATA UNDERSTANDING

SmartAdmit is designed around a value-centric business model that addresses the core pain points of academic institutions like ESPRIT based on the institution own DATA.

BUISNESS UNDERSTANDING

The primary objective of SmartAdmit is to support ESPRIT in addressing two strategic challenges:

1. Optimizing the admission process: making candidate evaluation faster, more objective, and better aligned with program-specific criteria.
2. Improving alumni tracking: collecting and analyzing post-graduation employment data to measure educational outcomes and support continuous improvement.

BUISNESS UNDERSTANDING

Strategic Priority	Business Impact
FASTER ADMISSION DECISION-MAKING	REDUCES ADMINISTRATIVE WORKLOAD AND IMPROVES APPLICANT EXPERIENCE
FAIR AND PERSONALIZED CANDIDATE EVALUATION	INCREASES QUALITY OF ADMITTED STUDENTS AND REDUCES MISMATCH RISKS
DATA-DRIVEN ADMISSION AND EMPLOYABILITY INDICATORS	ENHANCES INSTITUTIONAL CREDIBILITY AND MARKET ALIGNMENT
STRENGTHENING ALUMNI-EMPLOYER TIES	BUILDS LONG-TERM PARTNERSHIPS AND IMPROVES GRADUATE VISIBILITY

DATA UNDERSTANDING

The data used in SmartAdmit is sourced from both historical admission records and alumni LinkedIn profiles. The aim was to understand the profiles that lead to successful academic outcomes and professional integration.

DATA UNDERSTANDING

DATA SOURCES:

- ADMISSION DATA (STRUCTURED): CANDIDATE INFORMATION (NAME, AGE, DIPLOMA, SCORE, ETC.) APPLICATION TIMESTAMPS AND DECISIONS ACADEMIC
- ALUMNI DATA (SEMI-STRUCTURED/UNSTRUCTURED): LINKEDIN SCRAPING, EMPLOYER DATA, JOB TITLES, SECTORS

data understanding

1. Table: specialites.csv

- **Columns**:
  - **INFO**: Code for the Computer Science specialty.
  - **ELECTRO**: Code for the Electromechanical specialty.
  - **GENICIV**: Code for the Civil Engineering specialty.
  - **TELCOMM**: Code for the Telecommunications specialty.

2. Table: diplomes.csv

- **Columns**:
  - **DII**: Diploma of Engineer in Computer Science.
  - **DIEM**: Diploma of Engineer in Electromechanics.
  - **DIGC**: Diploma of Engineer in Civil Engineering.
  - **DITC**: Diploma of Engineer in Telecommunications.

3. Table: pieces.csv

- **Columns**:
  - **DIP**: Diploma.
  - **EN**: Birth certificate.
  - **PI**: Identity photo.
  - **Pass**: Passport.
  - **DB**: Baccalaureate diploma.
  - **DM**: Medical file.
  - **CIN**: National identity card.

4. Table: institutions.csv

- **Columns**:
  - **code_inst**: Code for the institution.
  - **institution**: Name of the institution.

5. Table: countries.csv

- **Columns**:
  - **code_pays**: Code for the country.
  - **pays**: Name of the country.
  - **nationalite**: Associated nationality.

6. Table: governorates.csv

- **Columns**:
  - **code_gouv**: Code for the governorate.
  - **gouvernorat**: Name of the governorate.
  - **code_pays**: Code for the country (here, Tunisia).

7. DIPLOMES_SPECIALITES

- **Description**: This table links diplomas to their respective specialties.
- **Fields**:
  - **code_diplome**: Unique identifier for the diploma.
  - **code_spec**: Identifier for the specialty.

8. ANNEE_UNIVERSITAIRES

- **Description**: This table stores academic years.
- **Fields**:
  - **annee_deb**: Start year.
  - **annee_fin**: End year.

9. CURSUS

- **Description**: This table defines different study programs.
- **Fields**:
  - **code_cursus**: Unique identifier for the program.
  - **cursus**: Description of the program.

10. DEMARRAGE_ANNEE_UNIV

- **Description**: Table that manages the start of academic years for different institutions and programs.
- **Fields**:
  - **id_annee_univ**: Unique identifier for each record.
  - **code_inst**: Code of the institution.
  - **code_cursus**: Reference to the program.
  - **code_diplome** and **code_spec**: References to the diplomas and specialties.
  - **annee_deb** and **annee_fin**: Period of the academic year.
  - **date_demarrage_au**: Start date.
  - **date_cloture_au**: End date.

11. CONCOURS

- **Description**: Table that records competitions for admission.
- **Fields**:
  - **id_concours**: Unique identifier for the competition.
  - **date_ouvrerture**: Opening date for applications.
  - **date_cloture**: Closing date for applications.

12. PIECES_EXIGEES

- **Description**: This table defines the documents required for each competition.
- **Fields**:
  - **id_concours**: Reference to the competition.
  - **code_piece**: Code for the required document.
  - **langue_piece**: Language of the document.
  - **nb**: Number of required documents.

13. VILLES

- **Description**: This table stores information about cities.
- **Fields**:
  - **code_postal**: Postal code of the city.
  - **ville**: Name of the city.
  - **code_gouv**: Code of the governorate.

14. CANDIDATS

- **Description**: Table that records the personal information of candidates.
- **Fields**:
  - **id_et**: Unique identifier for the student (candidate).
  - **nom_et** and **pnom_et**: Name and surname of the candidate.
  - **date_nais_et**: Date of birth.
  - **lieu_nais_et**: Place of birth.
  - **sexe**: Gender of the candidate.

15. NATURE_BAC

- **Description**: This table identifies types of baccalaureate.
- **Fields**:
  - **nature_bac**: Unique identifier for the type of bac.
  - **bac**: Description of the type of bac.

- **Description**: This table contains information about schools that issue baccalaureates.
- **Fields**:
  - **code_etab_bac**: Unique identifier for the establishment.
  - **etab_bac**: Name of the establishment.

17. BAC

- **Description**: Records the baccalaureate results for each candidate.
- **Fields**:
  - **id_et**: Reference to the candidate.
  - **nature_bac**: Reference to the type of bac.

18. PARENTS

- **Description**: Table that records the information of candidates' parents.
- **Fields**:
  - **id_et**: Reference to the candidate.
  - **cin_pere** and **cin_mere**: Identifiers for the parents.

19. CANDIDATURES

- **Description**: Records the applications of students for different competitions
- **Fields**:
  - **id_et**: Reference to the candidate.
  - **id_concours**: Reference to the competition.

20. DOSSIERS

- **Description**: This table manages the documents submitted by candidates.
- **Fields**:
  - **id_et**: Reference to the candidate.
  - **id_concours**: Reference to the competition.

KEY INSIGHTS FROM DATA EXPLORATION

Dimension	Observations
CANDIDATE DIVERSITY	WIDE VARIATION IN ACADEMIC BACKGROUND, AGE, AND DIPLOMA TYPE
ADMISSION DECISION DELAYS	NO AUTOMATION – ALL EVALUATION DONE MANUALLY
WEAK ALUMNI LINKAGE	POST-GRADUATION DATA FRAGMENTED, NO CENTRALIZED VIEW
LIMITED SUCCESS PREDICTORS	ADMISSION CRITERIA NOT CORRELATED WITH POST-GRADUATION SUCCESS

BUSINESS-TO-DATA TRANSLATION

TO BRIDGE THE GAP BETWEEN BUSINESS NEEDS AND TECHNICAL IMPLEMENTATION, WE TRANSLATED THE ABOVE BUSINESS PRIORITIES INTO DATA-DRIVEN OBJECTIVES:

Business Question	Data Question
WHICH CANDIDATES ARE MOST LIKELY TO SUCCEED?	WHAT FEATURES CORRELATE WITH ACADEMIC PERFORMANCE AND EMPLOYABILITY?
ARE OUR ADMISSION DECISIONS OPTIMAL AND FAIR?	HOW DO ADMITTED VS. REJECTED CANDIDATES PERFORM POST-ADMISSION?
ARE WE LOSING CONTACT WITH OUR ALUMNI?	HOW MANY ALUMNI ARE EMPLOYED, AND IN WHICH SECTORS?
ARE SOME DIPLOMAS OR PROFILES SYSTEMATICALLY UNDERPERFORMING?	WHAT PROFILES HAVE LOW POST-GRADUATION SUCCESS RATES?

BUISNESS MODEL CANVAS

KEY PARTNERS

Universities administration , IT department , Academic departments , Alumni associations , Employer networks and recruiters , Public/private sponsors and education sector agencies

KEY ACTİVİTİES

Platform development
AI integration
Data processing
Customer support
Regulatory monitoring
Digital communication

KEY RESOURCES

Technical team
Cloud servers
Educational databases
Academic network
Admissions/HR experts
Analytics tools

VALUE PROPOSİTİONS

Fast and automated admission
Fewer human errors
Customizable criteria
Graduate tracking
Fairness and transparency
Useful predictions
Cost reduction
Improved institutional reputation

CUSTOMER RELATİONSHİP

Self-service portal
Multichannel support
Personalized follow-up
User training
Feedback and improvement

CHANNEL

Social media
Webinars and fairs
Official partnerships
Targeted emailing

CUSTOMER SEGMENTS

Medium-sized public /private universities in Greater Tunis

COST STRUCTURE

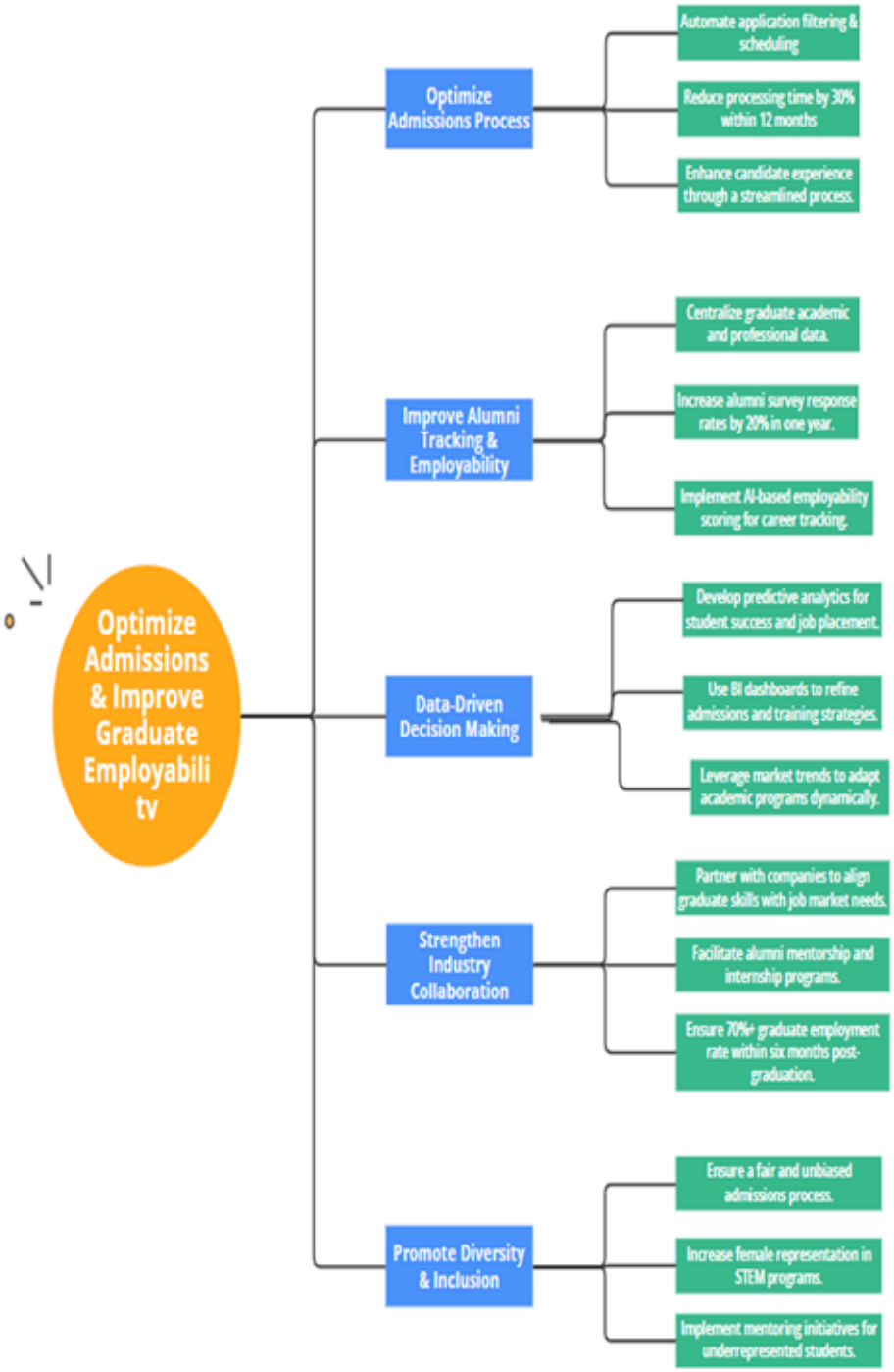
GDPR compliance and security
Salaries
Marketing
Partnerships
Ongoing R&D

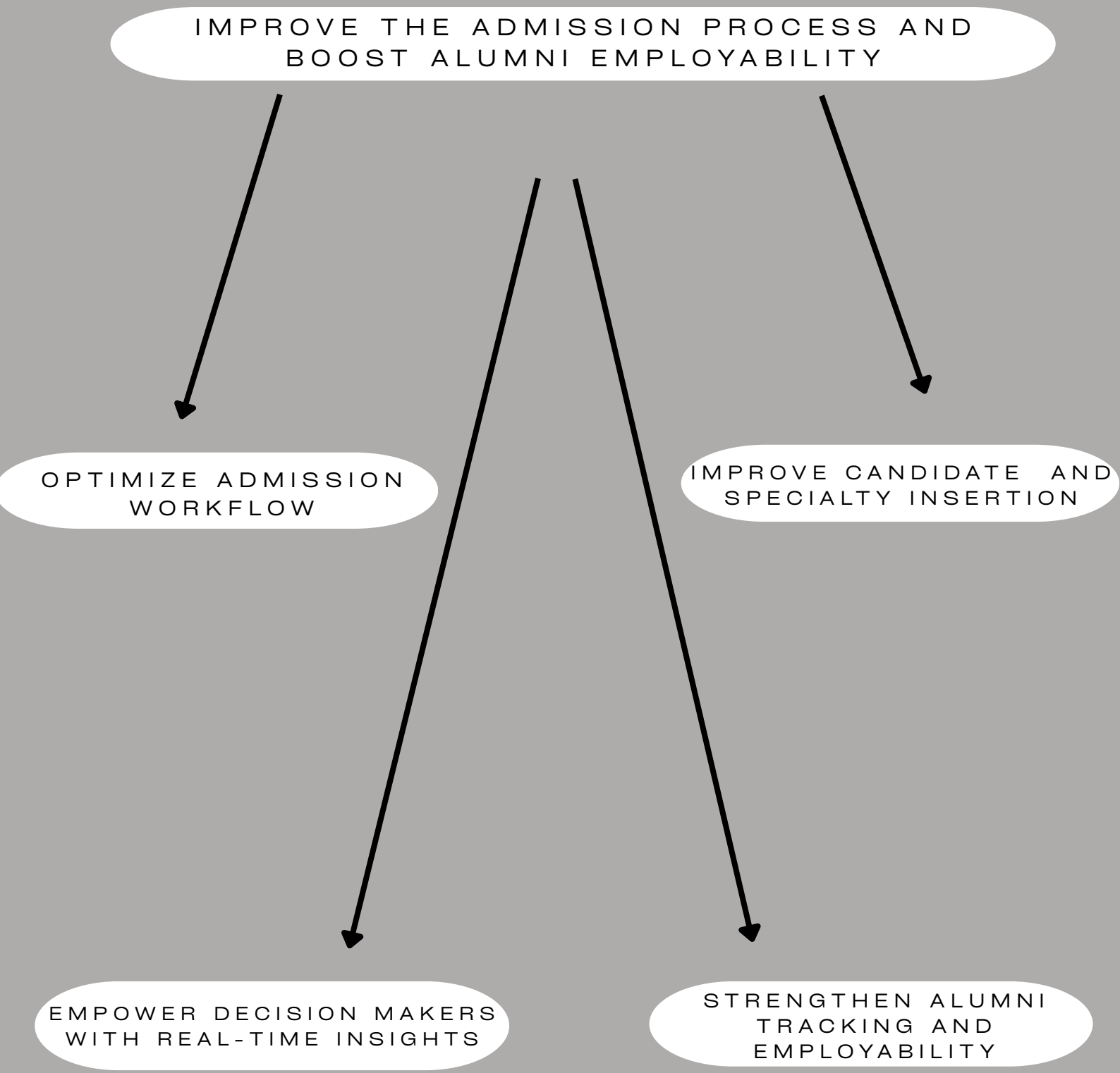
REVENUE STREAM

Annual subscriptions
Premium modules
Personalized services
Grants / project calls
Private partnerships

01. Customer Segments:
Medium-sized private universities in Greater Tunis
02. Value Propositions:
Transparent and intelligent admission system Enhanced employability tracking Reduced administrative workload Improved institutional reputation
03. Channels:
Web platform Institutional system integrations Internal dashboards and reporting tools
04. Customer Relationships:
Long-term collaboration with academic institutions Training and onboarding support Customizable modules and user feedback loops
05. Revenue Streams:
Licensing fees Implementation , customization services Data analytics and reporting services
06. Key Resources:
Development team Data infrastructure and storage Internal academic and student records External labor market data sources
07. Key Activities:
Data integration and pipeline management Admission automation workflows Alumni data collection and analytics System maintenance and support
08. Key Partners:
ESPRIT administration , IT department , Academic departments , Alumni associations , Employer networks and recruiters , Public/private sponsors and education sector agencies
09. Cost Structure: Software development and maintenance Data infrastructure and hosting Support services and updates Marketing and stakeholder engagement

OBJECTIFS TREE




IMPROVE THE ADMISSION PROCESS AND BOOST ALUMNI EMPLOYABILITY
OPTIMIZE ADMISSION WORKFLOW
IMPROVE CANDIDATE AND SPECIALTY INSERTION
EMPOWER DECISION MAKERS WITH REAL-TIME INSIGHTS
STRENGTHEN ALUMNI TRACKING AND EMPLOYABILITY

OPTIMIZE ADMISSION WORKFLOW

REDUCE PROCESSING TIME PER APPLICATION

KPI:AVERAGE TIME IN APPLICATION HANDLING(DAYS)

MINIMIZE MANUAL ERRORS IN APPLICATION HANDLING

INCREASE TRANSPARENCY IN DECISION-MAKING

KPI: PERCENTAGE OF APPLICATION PROCESSED WITH SCORING CRITERIA

IMPROVE CANDIDATE AND SPECIALTY INSERTION

PREDICT CANDIDATE SUCCESS

MACHINE LEARNING SCRIPT BASED ON SELECTED FEATURES

KPI : PREDICTION ACCURACY RATE OF ML MODEL FOR CANDIDATE SELECTION

RECOMMEND SUITABLE SPECIALTIES

DEPP LEARNING SCRIPT BASED ON SELECTED FEATURES

KPI : RECOMMANDATION ACCURACY AND POST-ENROLLEMENT SUCCESS IN THE SUGGESTED SPECIALTIES

STRENGTHEN ALUMNI TRACKING AND EMPLOYABILITY

MONITOR POST-GRADUATION EMPLOYMENT STATUS

KPI : EMPLOYMENT RATE WITHIN A SPECIFIC PERIODE POST GRADUATION

IDENTIFY ALUMNI PROFILE AND PERFORMANCE

KPI :PERCENTAGE OF ALUMNIS POSTS , PORMOTION AND AVERAGE SALARY

PREDECTION OF EMPLOYABILITY RATES FOR NEXT YEARS

MACHINE LEARNING SCRIPT BASED ON SELECTED FEATURES

KPI : PREDICTION ACCURACY RATE OF ML MODEL FOR EMPLOYABILITY PREDECTION

EMPOWER DECISION MAKRES WITH REAL-TIME INSIGHTS

PROVIDE DYNAMIC DASHBOARDS AND VISUALS ANALYTICS FOR ADMISSION DECISION MAKING

PROVIDE DYNAMIC DASHBOARDS AND VISUALS ANALYTICS FOR EMLOYABILITY DECISION MAKING

KEY DECISION MAKERS

ADMISSIONS MANAGER

EMPLOYABILITY MANAGER

09

ADMISSIONS MANAGER

THIS ROLE REQUIRES RAPID YET PRECISE DECISION-MAKING ON:

- ADMISSION THRESHOLDS
- RESOURCE ALLOCATION FOR INTERVIEWS
- EARLY DETECTION OF DROPOUT-PRONE PROFILES
- GENDER AND REGIONAL BALANCE IN ENROLLMENT.

SMARTADMIT ENABLES THESE DECISIONS WITH PREDICTIVE ANALYTICS AND REAL-TIME DASHBOARDS, REDUCING MANUAL WORK AND BIAS

EMPLOYABILITY MANAGER

THIS ROLE FOCUSES ON MONITORING AND IMPROVING POST-GRADUATION OUTCOMES:

- ALUMNI INSERTION TRACKING
- STRATEGIC PARTNERSHIPS WITH TOP RECRUITERS
- SALARY BENCHMARKING
- FOLLOW-UP OF GRADUATES IN UNDERPERFORMING SECTORS

SMARTADMIT PROVIDES CENTRALIZED, LONGITUDINAL DATA THAT SUPPORTS BOTH REPORTING AND PROACTIVE MEASURES TO BOOST EMPLOYMENT OUTCOMES.

DASHBOARD AND KPI'S

SMARTADMIT WAS BUILT AROUND STRATEGIC KPIS DIRECTLY EXTRACTED AND VISUALIZED THROUGH CUSTOMIZED DASHBOARDS. THESE INDICATORS EMPOWER ESPRIT'S DECISION-MAKERS TO ACT BASED ON REAL-TIME, STRUCTURED INSIGHTS.

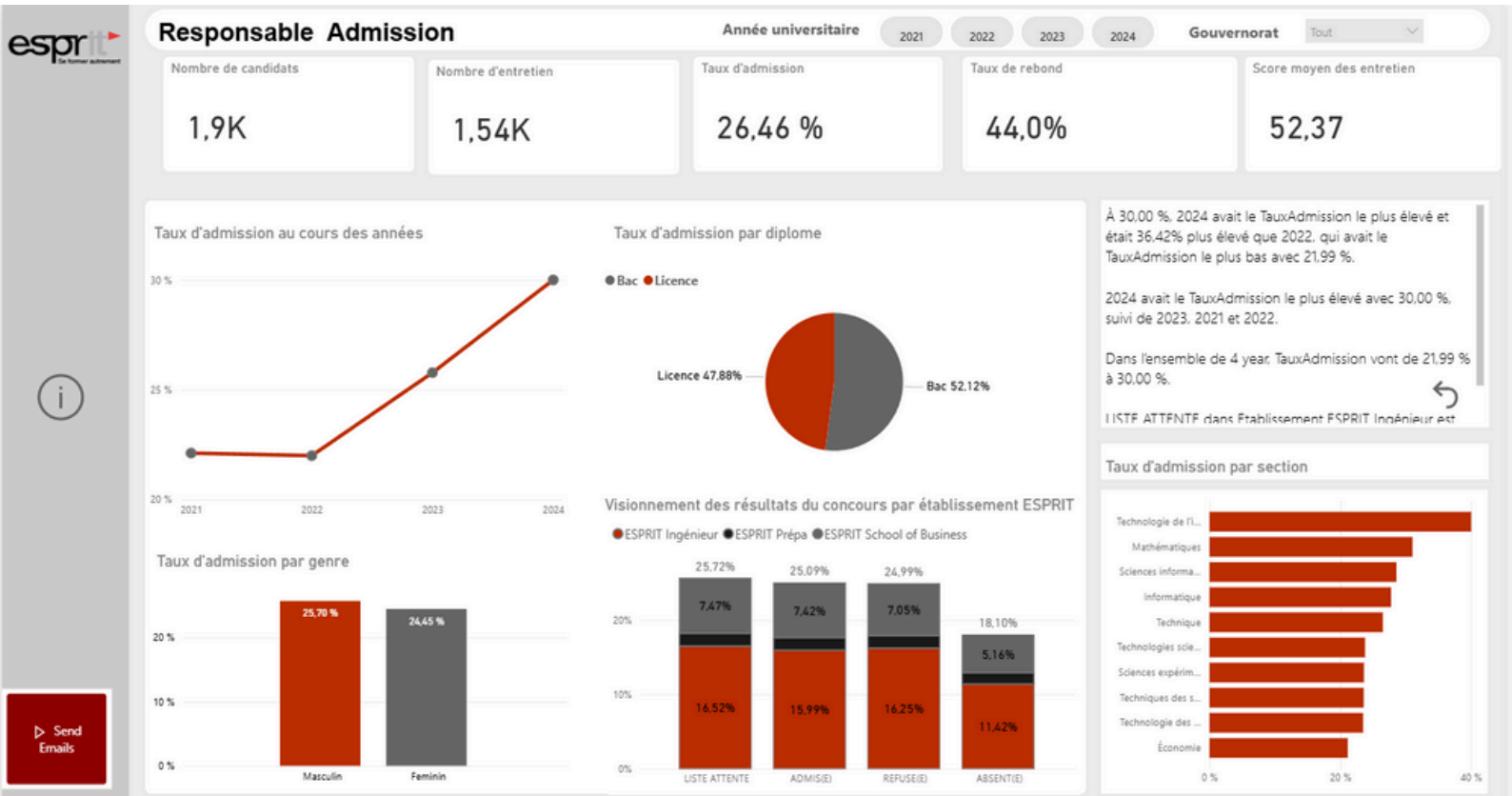
DECISION MAKER : ADMISSION MANAGER

DASHBOARD 01

KPI'S

- ADMISSION RATE PER YEAR
- BOUNCE RATE
- AVERAGE INTERVIEW SCORE
- CONVERSION RATE: APPLICATIONS → INTERVIEWS → ENROLLMENTS ADMISSION BY SECTION & DEGREE
- CLUSTERING OF CANDIDATES BASED ON SCORES AND PROFILES (MACHINE LEARNING VIEW)

THESE KPIS HELP THE ADMISSIONS TEAM:
→OPTIMIZE SELECTION CRITERIA.
→SPOT PATTERNS IN DROPOUT OR MISMATCH.
→IMPROVE GENDER EQUITY AND REGIONAL BALANCE. →ANTICIPATE PROGRAM DEMAND BY DIPLOMA AND SECTION.



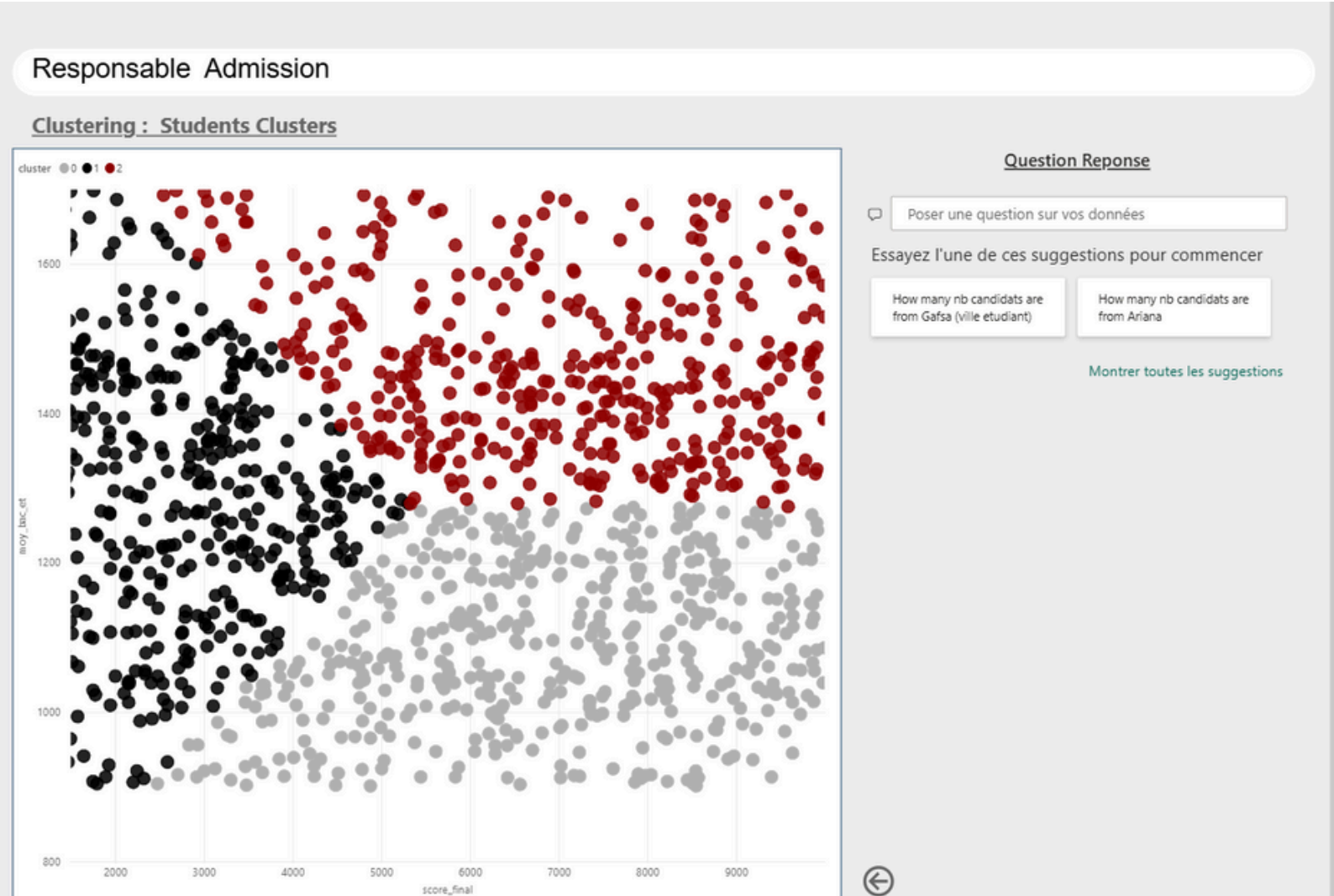
DASHBOARD 02

- CLUSTERING OF CANDIDATES (ML-BASED SEGMENTATION) :

DEFINITION: GROUPING CANDIDATES BASED ON SIMILARITIES IN ATTRIBUTES (FINAL SCORE, INTERVIEW SCORE, REGION, BACKGROUND).
TECHNIQUE: THIS IS UNSUPERVISED MACHINE LEARNING, TYPICALLY USING K-MEANS

STEPS:
- DEFINE FEATURES
- STANDARDIZE VALUES. APPLY CLUSTERING ALGORITHM
- VISUALIZE CLUSTERS TO INTERPRET PATTERNS

OUTPUT:
- CLUSTER 0: LOW-PERFORMING
- CLUSTER 1: MEDIUM
- CLUSTER 2: HIGH-PERFORMING



THE FORMULAS AND METHODS FOR EACH OF THE KPIS :

- ADMISSION RATE = (NUMBER OF ADMITTED CANDIDATES / TOTAL NUMBER OF APPLICANTS) × 100
- BOUNCE RATE = (NUMBER OF INCOMPLETE APPLICATIONS / TOTAL NUMBER OF APPLICATIONS STARTED) × 100
- AVERAGE INTERVIEW SCORE = (SUM OF ALL INTERVIEW SCORES) / (NUMBER OF INTERVIEWED CANDIDATES)
- CONVERSION RATE 1 = (NUMBER OF INTERVIEWS CONDUCTED / TOTAL APPLICATIONS) × 100
- GLOBAL CONVERSION RATE = (NUMBER OF ADMITTED CANDIDATES / TOTAL APPLICATIONS) × 100
- SECTION/DEGREE RATE = (NUMBER OF ADMITTED CANDIDATES IN GROUP / TOTAL ADMITTED CANDIDATES) × 100

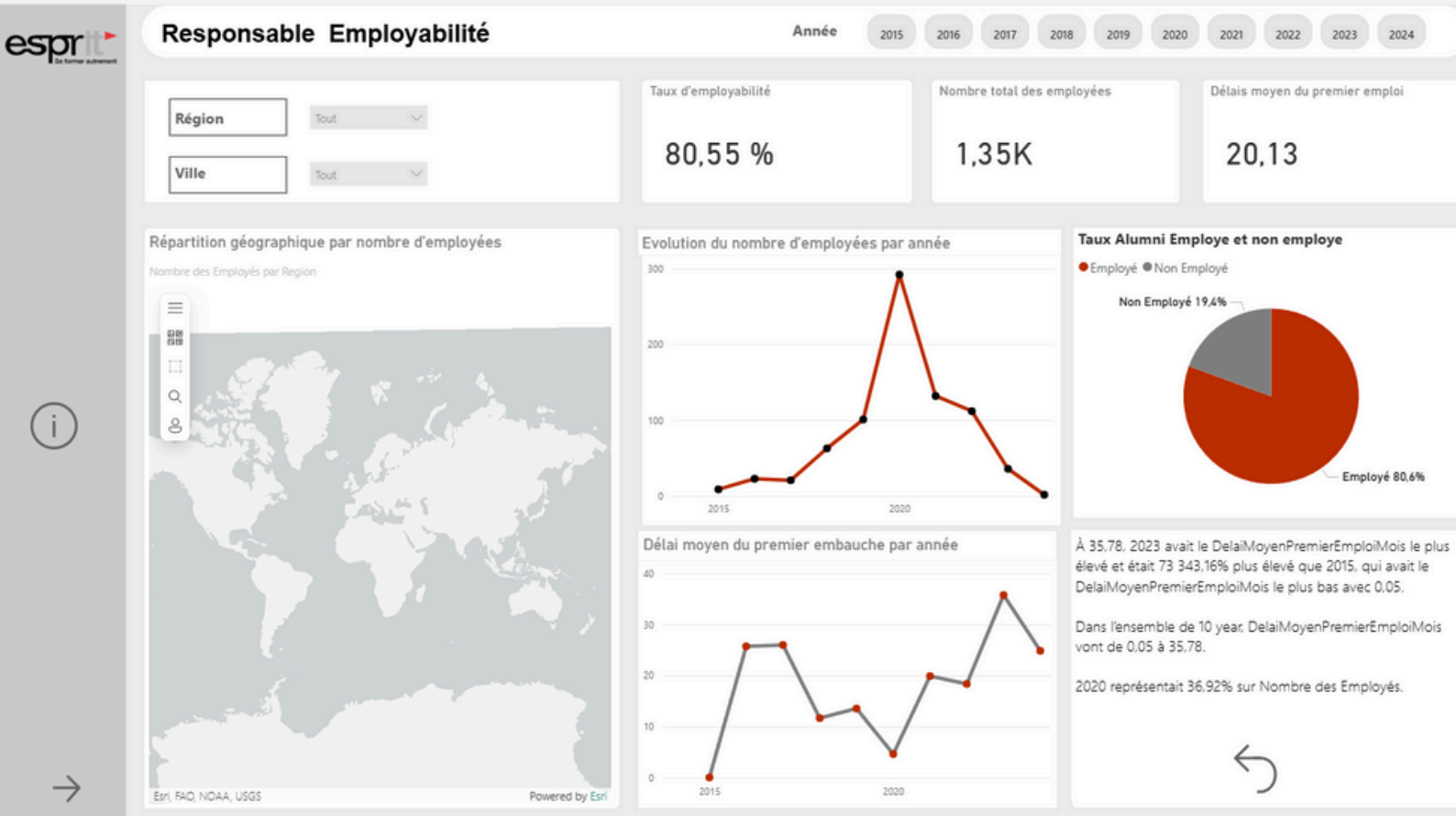
DECISION MAKER : EMPLOYABILITY MANAGER

DASHBOARD 01

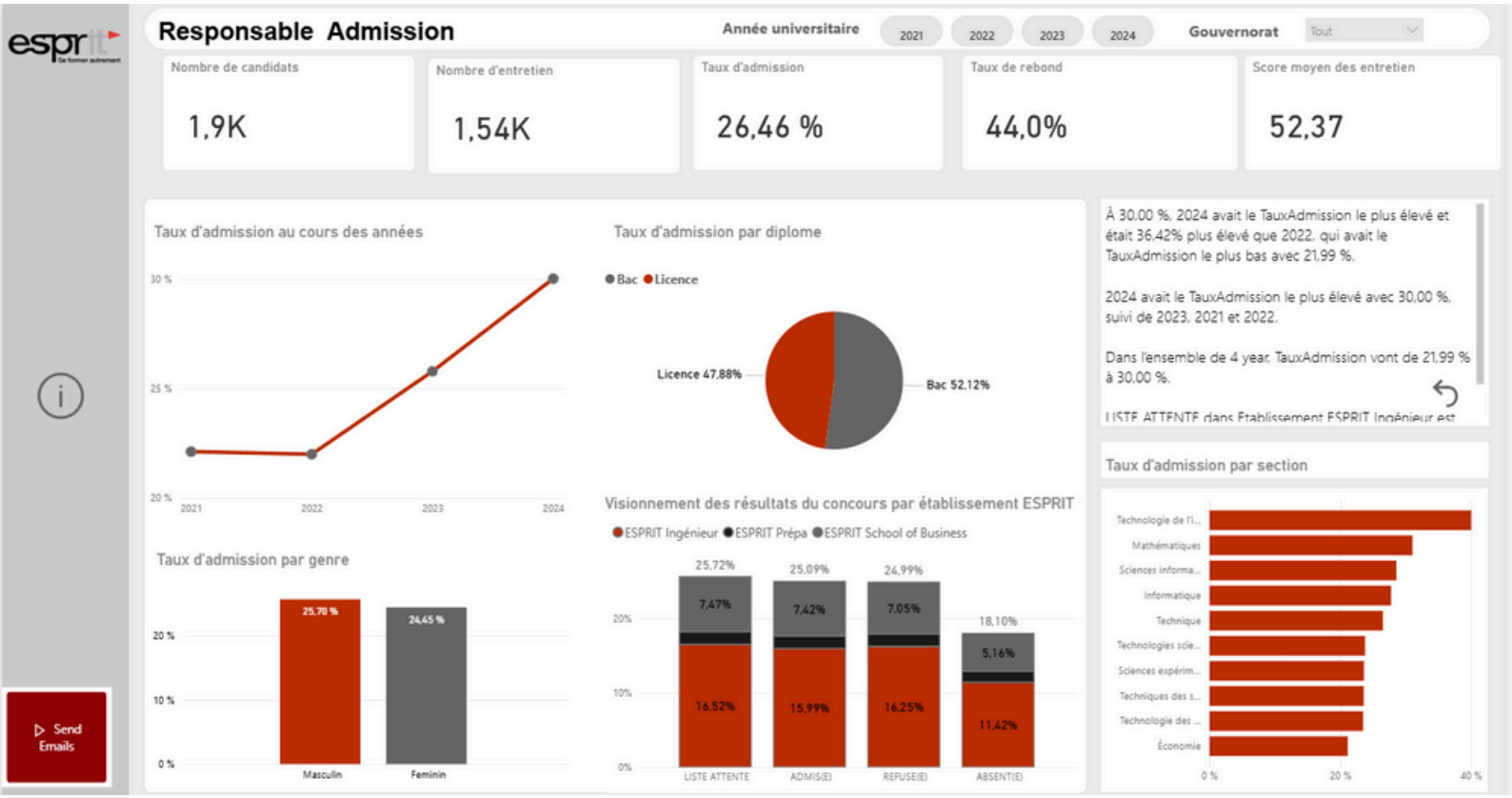
KPI'S

- EMPLOYMENT RATE
- AVERAGE DELAY TO FIRST JOB
- IMMEDIATE EMPLOYMENT RATE
- EMPLOYMENT BY SPECIALTY AND INDUSTRY
- GEOGRAPHICAL DISTRIBUTION OF ALUMNI
- TOP SALARIES BY COMPANY
- TOP POSITIONS HELD BY GRADUATES
- THESE INDICATORS ALLOW EMPLOYABILITY TEAMS TO:

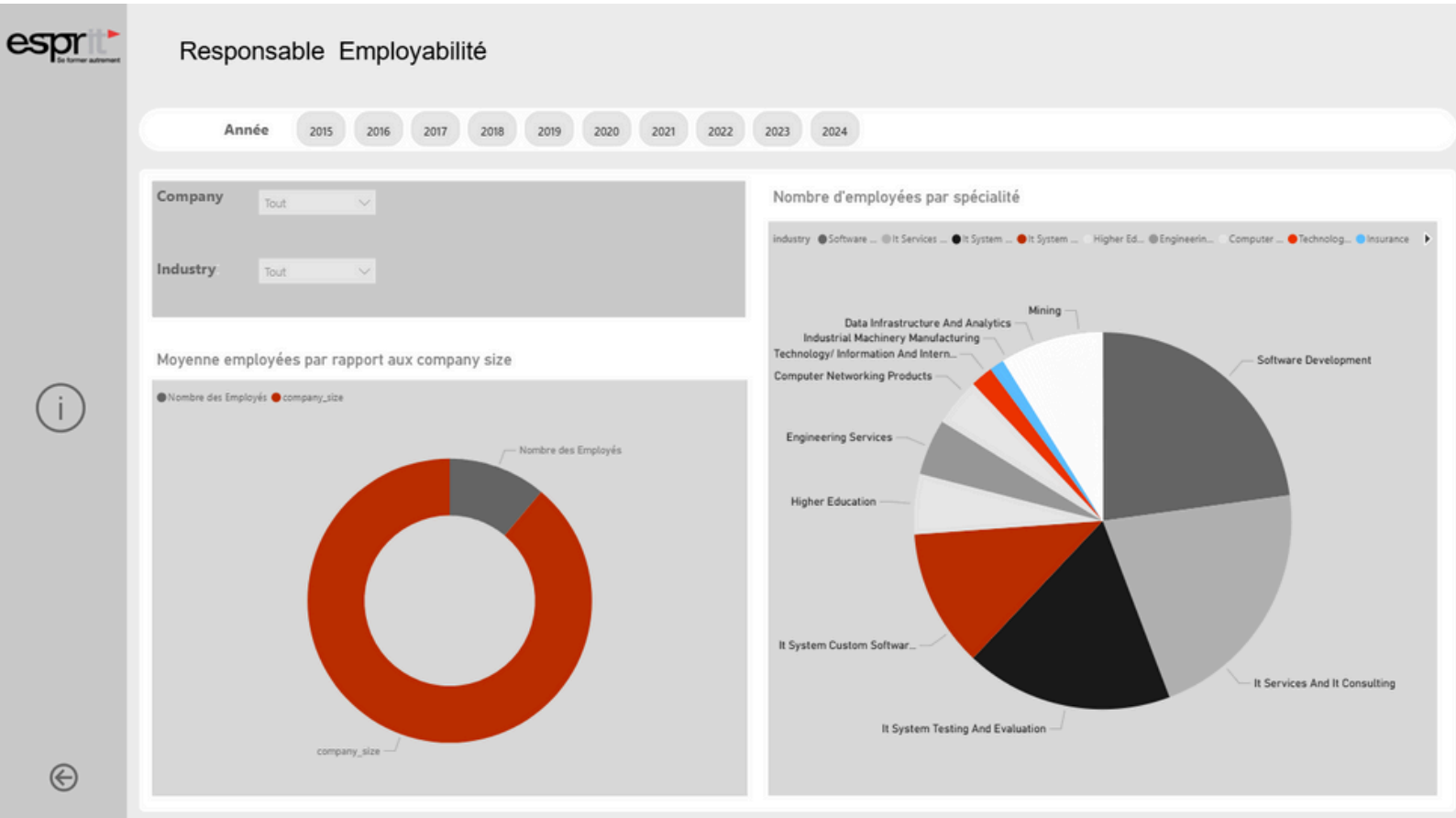
→EVALUATE TIME-TO-EMPLOYMENT TRENDS.
→MEASURE THE EFFICIENCY OF ACADEMIC-TO-JOB TRANSITIONS.
→IDENTIFY SECTORS WITH HIGH ABSORPTION CAPACITY.
→TARGET HIGH-PERFORMING COMPANIES FOR PARTNERSHIP.
→ADJUST ORIENTATION OR CURRICULUM BASED ON REAL-WORLD FEEDBACK.



DASHBOARD 02



DASHBOARD 03



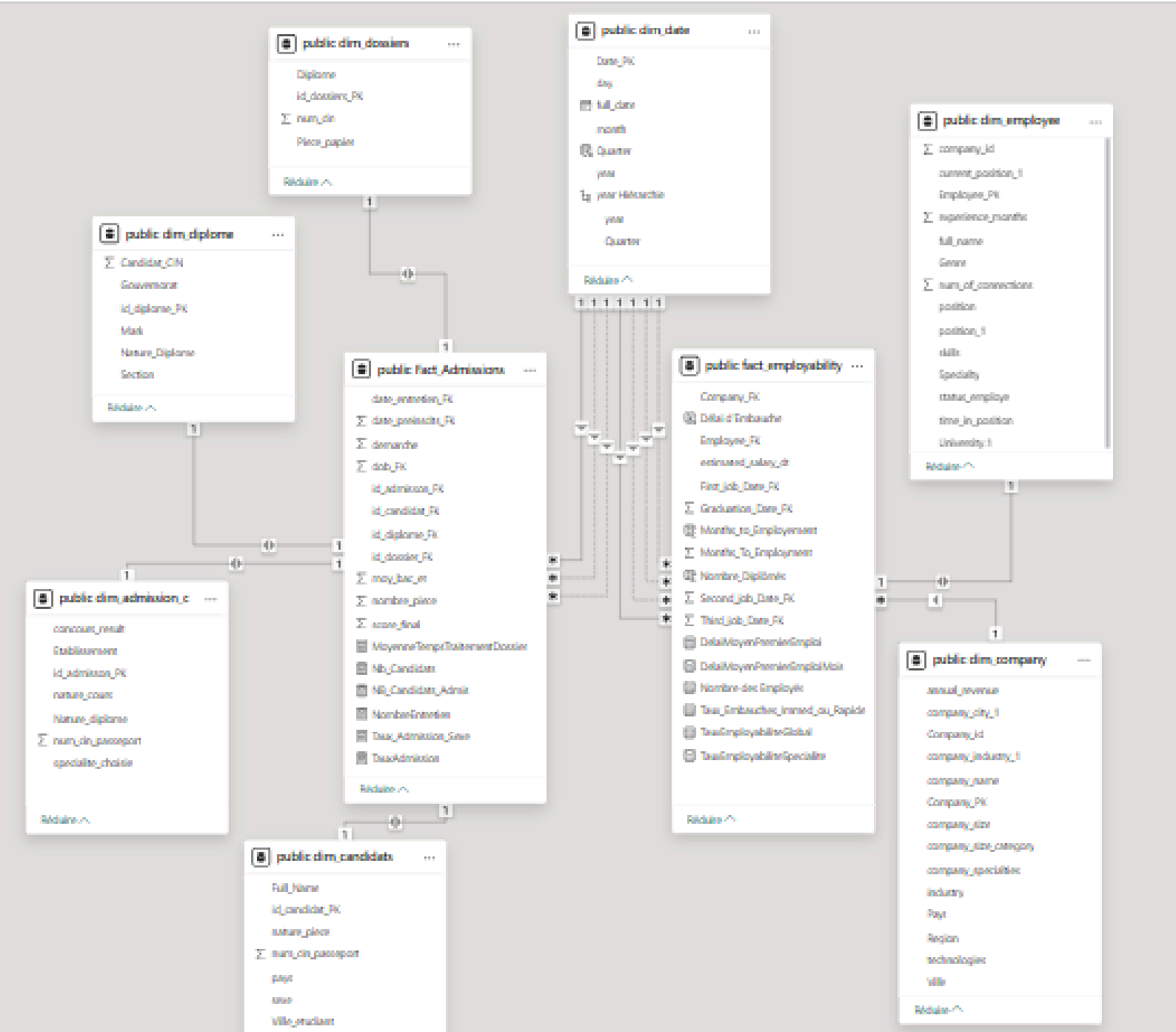
THE FORMULAS AND METHODS FOR EACH OF THE KPIS :

- EMPLOYMENT RATE = (NUMBER OF EMPLOYED GRADUATES / TOTAL GRADUATES TRACKED) × 100
- AVERAGE DELAY = (SUM OF ALL DELAYS TO FIRST EMPLOYMENT) / (NUMBER OF EMPLOYED GRADUATES)
- IMMEDIATE EMPLOYMENT RATE = (NUMBER OF IMMEDIATELY EMPLOYED GRADUATES / TOTAL GRADUATES) × 100
- PERCENTAGE = (NUMBER OF EMPLOYED GRADUATES IN GROUP / TOTAL EMPLOYED GRADUATES) × 100
- REGIONAL EMPLOYMENT SHARE = (NUMBER OF EMPLOYED GRADUATES IN REGION / TOTAL EMPLOYED GRADUATES) × 100
- AVERAGE SALARY (COMPANY X) = (SUM OF SALARIES OF ALUMNI IN COMPANY X) / (NUMBER OF ALUMNI IN COMPANY X)
- MAX DELAY = MAX(DELAY TO FIRST JOB FOR ALL EMPLOYED GRADUATES)
- TOTAL EMPLOYED = COUNT(GRADUATES WHERE EMPLOYMENT_STATUS = "EMPLOYED")

DESIGN AND MODELING

To support dual analytics for admission optimization and alumni employability, the SmartAdmit Data Warehouse is structured around two central fact tables and a set of well-normalized dimension tables. This structure enables powerful insights across the entire student lifecycle—from application to job placement.

DWH MODEL


public: dim_dossiers
Diplome
id_dossiers_PK
num_cin
Piece_papier
Réduire
public: dim_date
Date_PK
day
full_date
month
Quarter
year
year Hiérarchie
year
Quarter
Réduire
public: dim_employe
company_id
current_position_1
Employee_PK
experience_months
full_name
Genre
num_of_connections
position
position_1
skills
Specialty
status_employe
time_in_position
University 1
Réduire
public: dim_diplome
Candidat_CIN
Gouvernorat
id_diplome_PK
Mark
Nature_Diplome
Section
Réduire
public: fact_Admissions
date_entretien_FK
date_publication_FK
demarche
dob_FK
id_admission_FK
id_candidat_FK
id_diplome_FK
id_dossier_FK
moy_bac_en
nombre_piece
score_final
MoyenneTempsTraitementDossier
Nb_Candidats
NB_Candidats_Admis
NombreEntretien
Taux_Admission_Genre
TauxAdmission
Réduire
public: fact_employability
Company_FK
Délai d'Embauche
Employee_FK
estimated_salary_dt
First_job_Date_FK
Graduation_Date_FK
Months_to_Employement
Months_To_Employement
Nombre_Diplômés
Second_job_Date_FK
Third_job_Date_FK
DelaiMoyenPremierEmploi
DelaiMoyenPremierEmploiMois
Nombre des Employés
Taux_Embauches_Immed_ou_Rapide
TauxEmployabiliteGlobal
TauxEmployabiliteSpécialité
Réduire
public: dim_admission_c
concours_result
Etablissement
id_admission_PK
nature_cours
Nature_diplome
num_cin_passeport
spécialité_choisie
Réduire
public: dim_candidats
Full_Name
id_candidat_PK
nature_piece
num_cin_passeport
pays
sexe
Ville_etudiant
public: dim_company
annual_revenue
company_city_1
Company_id
company_industry_1
company_name
Company_PK
company_size
company_size_category
company_specialities
industry
Pays
Region
technologies
Ville
Réduire

DWH MODEL

FACT TABLES :

1. FACT_ADMISSION

TRACKS ALL APPLICATION-LEVEL INFORMATION AND DECISIONS.

KEY FIELDS:

- ID_ADMISSION_PK
- ID_CANDIDAT_FK
- ID_FILIERE_FK
- ID_DIPLOME_FK
- DATE_ADMISSION SCORE
- RESULTAT_ADMISSION
- MOTIF_REFUS
- ID_DATE_FK

->USE CASES:

EVALUATE ADMISSION TRENDS BY YEAR/DIPLOMA ANALYZE PERFORMANCE BY PROGRAM IDENTIFY FACTORS LEADING TO REJECTION

2. FACT_EMPLOYABILITY :

CAPTURES ALUMNI JOB OUTCOMES AND EMPLOYABILITY INDICATORS. KEY FIELDS:

- ID_EMPLOYABILITE_PK
- ID_CANDIDAT_FK
- ID_EMPLOI_FK
- ID_COMPANY_FK
- STATUT_EMPLOI
- SALAIRE
- DATE_PREMIER_EMPLOI
- DELAI_EMBAUCHE
- MATCHING_DIPLOME_EMPLOI
- ID_DATE_FK

-> USE CASES:

TRACK EMPLOYMENT RATE BY DIPLOMA MEASURE TIME TO EMPLOYMENT POST-GRADUATION EVALUATE JOB MATCHING QUALITY

DWH MODEL

DIMENSION TABLES:

Dimension	Description
dim_candidat	Candidate personal details: gender, age, region, nationality
dim_filiere	Academic program details: department, type of degree
dim_diplome	Diploma prior to admission: type, sector
dim_date	Date hierarchy: day, month, quarter, year
dim_emploi	Job attributes: title, sector, seniority level
dim_company	Employer data: name, sector, location, size
dim_domaine	Sector/domain classification for prior diplomas and jobs

SCHEMA LOGIC

The dim_date is shared by both fact tables to allow for consistent time-based reporting.
The dim_candidat is a key shared entity that links both admission and employability, enabling full life-cycle analysis such as:
“Do candidates who graduated from a technical baccalaureate perform better in employability outcomes?”

Analytical Scenarios Supported :
Admission analysis by diploma, filière, and region
Employment success rates by academic background or domain
Profile scoring models based on past employability success
Time-to-hire indicators across different academic programs
Program effectiveness: matching jobs to diploma fields

BI SOLUTION ARCHITECTURE

The SmartAdmit Business Intelligence (BI) architecture was designed to provide end-to-end analytical visibility across the full student lifecycle—from initial application to alumni employability. It integrates multiple data sources, centralizes them in a star-schema DWH, and provides insights through dashboards and predictive tools.

ARCHITECTURE OVERVIEW

The architecture is structured into six functional layers, ensuring modularity, performance, and strategic alignment with ESPRIT's goals.

1/DATA SOURCES LAYER

Application Data :
CSV / Excel files
Application platforms
Student Academic Records
Admission results, rejection reasons, program assignment
Alumni Data :
LinkedIn scrapped profiles

2/ ETL LAYER (DATA INTEGRATION)

DATA IS EXTRACTED, CLEANED, TRANSFORMED, AND LOADED INTO THE DATA WAREHOUSE USING PYTHON SCRIPTS AND ETL TOOLS (TALEND).
KEY TASKS: MERGE CANDIDATE, DIPLOMA, AND EMPLOYMENT DATA
NORMALIZE AND ENRICH FIELDS
GENERATE SURROGATE KEYS FOR FACT/DIMENSION INTEGRATION

ARCHITECTURE OVERVIEW

3/DATA WAREHOUSE LAYER (POSTGRESQL)

The DWH uses a star schema design centered on:

fact_admission: admission process metrics

fact_employability: post-graduation outcomes

Dimensions)

4/ BUSINESS INTELLIGENCE LAYER (VISUALIZATION & ANALYTICS):

USING POWER BI, DASHBOARDS PROVIDE REAL-TIME INSIGHTS FOR ADMINISTRATORS, ACADEMIC COORDINATORS, AND STRATEGY TEAMS.

ARCHITECTURE OVERVIEW

5/ MACHINE LEARNING & PREDICTIVE MODELING LAYER

Optional models built using Python (scikit-learn, XGBoost):

Admission Prediction

Employability predicition

Specialty Recommandation

6/ SMARTADMIT FRONTEND (USER INTERFACE)

- FRONTEND: ANGULAR
- AUTHENTICATION: FIREBASE
- CONNECTS TO APIS AND DISPLAYS DYNAMIC DATA: CANDIDATE STATUS TRACKING VISUAL SCORE BREAKDOWN PER APPLICANT REAL-TIME DASHBOARD EMBEDDING (FROM POWER BI OR RECHARTS) ALUMNI UPDATE FORMS (CAREER UPDATES, EMPLOYMENT INFO)

7/ SECURITY & GOVERNANCE

ROLE-BASED ACCESS CONTROL (ADMIN, ADMISSION OFFICER, ALUMNI MANAGER)

ANONYMIZATION OF SENSITIVE ALUMNI DATA

GDPR-ALIGNED DATA HANDLING

IMPLEMENTATION

13

DATA-DRIVEN ADMISSION SYSTEM DEVELOPMENT PROCESS :INCLUDING TOOLS AND TECHNOLOGIES

The development of SmartAdmit followed a structured, data-centric methodology tailored to ESPRIT's real operational environment. The process was carried out through the following phases :

SmartAdmit Project – Key Steps

Data Acquisition & Cleaning

- Collected internal student and admission data from ESPRIT.
- Cleaned, normalized, and enriched it with external labor market data.

Data Analysis & Centralization

- Merged all datasets.
- Analyzed for patterns in student success, admission effectiveness, and employability.

System Design

- Created system mockups.
- Designed a modular architecture to ensure scalability and performance.

Model Development

- Developed machine learning models for candidate ranking, admission scoring, and employability prediction.

Technical Implementation

- Backend/API: Flask
- Frontend: Angular
- Authentication: Firebase
- Data Processing: Talend (ETL workflows)
- Data Analysis/ML: Python
- Databases: PostgreSQL & SQL Server

Testing & Iteration

- Conducted functional and user testing using real ESPRIT data.
- Refined the system based on feedback.

Deployment & Demonstration

- Deployed the solution.
- Presented it to the ESPRIT admissions office and adapted it to institutional requirements.

RUN1/RELEASE1

DATA INTERN
UNDERSTANDING
DATA EXTRACTION
SCRAPPED DATA
STAGING AREA PROCESS

DATA EXTRACTION

DIPLOMA SCANS(LICENCE/BAC) WERE PROCESSED USING PYTHON-BASED OCR SCRIPTS (ETESSERACT), ALLOWING EXTRACTION OF:
CANDIDATE NAME, DIPLOMA TITLE, INSTITUTION MENTION, YEAR OF GRADUATION FIELD OF STUDY THESE WERE THEN NORMALIZED AND MAPPED TO EXISTING DIM_DIPLOME VALUES.

Name	Date of Birth	National ID	Grade
Alia Naccache	02/05/1999	1505871	مقبول
Anas Melki	13/12/1997	1505867	جيد جدا
Anissa Materi	30/12/1997	1505864	جيد
Atef Mamou	01/08/1998	1505857	ممتاز
Basma Abid	01/11/2001	1505860	جيد جدا
Basma Touati	21/12/1997	1505888	جيد
Baya Manaï	27/04/2002	1505858	مقبول
Bechir Zarrouk	11/06/1999	1505894	جيد
Belhssan Kateb	10/12/2001	1505850	ممتاز
Brahim Slim	08/09/2000	1505883	جيد
Darine Sfar	26/06/2001	1505880	جيد جدا
Donia Laroussi	21/01/1999	1505852	مقبول
Emna Marzouk	25/11/1999	1505861	جيد جدا
Farah Lasram	23/11/2000	1505853	ممتاز
Farouk Ouertani	22/10/1997	1505874	ممتاز
Fatma Triki	19/06/2001	1505890	ممتاز
Ghalia Jouini	16/03/1997	1505847	جيد
Ghalia Mbarek	08/05/1998	1505865	جيد جدا
Ghassen Lelouch	24/12/2000	1505854	ممتاز
Haikel Karoui	16/10/1999	1505849	مقبول

SCRAPPED DATA

PARTNERS AND COMPETITORS (WEB SCRAPING)
USING SCRAPING TOOLS (BEAUTIFULSOUP, SELENIUM), WE COLLECTED DATA FROM: ESPRIT COMPETITORS' ADMISSION CRITERIA JOB OFFERS FROM PARTNER COMPANIES COMPANY WEBSITES AND PUBLIC ALUMNI SUCCESS STORIES THIS DATA WAS USED TO ENRICH: DIM_COMPANY (FOR EMPLOYABILITY) DIM_EMPLOI AND DIM_DOMAINE (FOR JOB SECTOR ANALYSIS) BENCHMARKING KPIS IN DASHBOARDING

PARTNERS

Nom	Logo	Lien
Universite de Paris 13	https://esprit.tn/uploac	https://www.univ-paris13.fr/
Telecom Ecole de Management (TEM)	https://esprit.tn/uploac	https://www.imt-bs.eu/
Institut Superieur d'Electronique de Paris (ISEP)	https://esprit.tn/uploac	https://www.isep.fr/
Universite du Maine	https://esprit.tn/uploac	http://www.univ-lemans.fr/fr/index.html
Ecole Superieure Africaine des TICs (ESATIC)	https://esprit.tn/uploac	http://www.esatic.ci/
Universite de Nice	https://esprit.tn/uploac	http://unice.fr/
Universite de Sherbrooke	https://esprit.tn/uploac	https://www.usherbrooke.ca/
Cegep de la Pocatiere	https://esprit.tn/uploac	http://www.cegeplapocatiere.qc.ca/Cegep/Accueil.html
Institut National des Postes et TÃ©lÃ©communications	https://esprit.tn/uploac	https://www.inpt.ac.ma/
Ecole Nationale Superieure des Mines de Rabat	https://esprit.tn/uploac	http://www.enim.ac.ma/
Ecole Nationale Superieure des Mines de Douai	https://esprit.tn/uploac	http://www2.mines-douai.fr/
Universite de Limoges	https://esprit.tn/uploac	http://www.unilim.fr/
Zayed University	https://esprit.tn/uploac	http://www.zu.ac.ae/main/ar/index.asp
University of Michigan Dearborn	https://esprit.tn/uploac	http://umdearborn.edu/
Universite Française d'Egypte	https://esprit.tn/uploac	http://www.ufe.edu.eg/
Mines Mauritanie	https://esprit.tn/uploac	http://www.emim.mr/
Institut Eurecom	https://esprit.tn/uploac	http://www.eurecom.fr/fr
INSA Euro-Mediterranee	https://esprit.tn/uploac	http://www.insa-euromediterranee.org/fr/index.html
Emlyon Business School	https://esprit.tn/uploac	https://www.em-lyon.com/fr
Cegep de Victoriaville	https://esprit.tn/uploac	https://www.cegepvicto.ca/
Cisco	https://esprit.tn/uploac	http://www.cisco.com/
L'Institut 2iE (Institut International d'IngÃ©nierie de l'Ea	https://esprit.tn/uploac	https://www.2ie-edu.org/index.php/fr/

COMPETITORS

Nom
Institut Superieur Prive Tunis Dauphine Tunis
Institut Superieur Prive Polytechnique Tunis
Institut Superieur Prive Mediterraneen de Technologie
Ecole Superieure Privee Polytechnique Ibn Khaldoun Tunis
Ecole Superieure Privee d'Ingenieurs et d'Etudes Technologiques de Tunis
Ecole Superieure Privee d'ingenierie et de communication Tunis
Ecole Superieure Privee d'Informatique et d'Administration des Affaires Tunis
Ecole Superieure Privee de technologies et d'ingenierie Tunis
Ecole Superieure Privee de Technologies de l'Information
Ecole Superieure Privee de Technologie et de Management Tunis
Ecole Superieure Privee de Carthage des Sciences et d'Ingenierie Tunis
Ecole Superieure Privee Centrale Polytechnique Tunis
Ecole Superieure Internationale Privee de Tunis
Ecole Mediterraneenne Superieure Privee Polytechnique

STAGINGAREA PROCESS

THE STAGING AREA WAS DESIGNED TO COLLECT, CLEAN, AND STANDARDIZE DATA FROM BOTH INTERNAL AND EXTERNAL SOURCES BEFORE INTEGRATION INTO THE DATA WAREHOUSE. IT ENSURED DATA QUALITY AND CONSISTENCY ACROSS ALL PIPELINES.

INTERNAL DATA SOURCES :

- CANDIDATE APPLICATIONS (EXCEL, CSV)
- ACADEMIC RECORDS FROM INTERNAL DATABASES
- ALUMNI LINKEDIN PROFILE.CSV
- DIPLOMAS (OCR EXTRACTION) DIPLOMA SCANS WERE PROCESSED USING PYTHON-BASED OCR SCRIPTS

EXTERNAL DATA SOURCES :

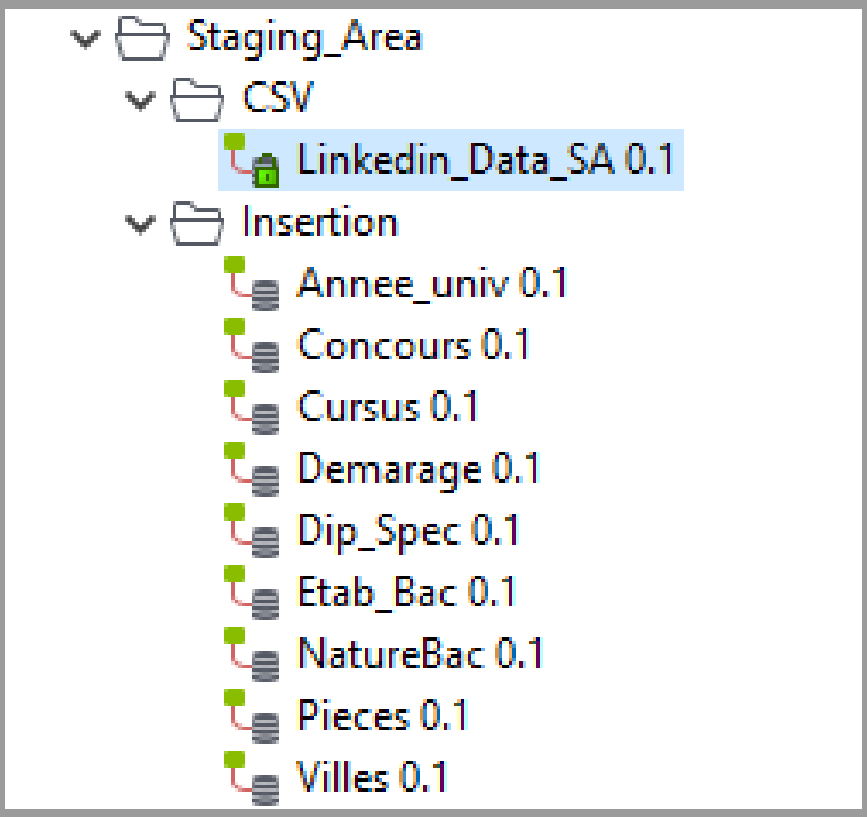
- PARTNERS AND COMPETITORS (WEB SCRAPING) USING SCRAPING TOOLS

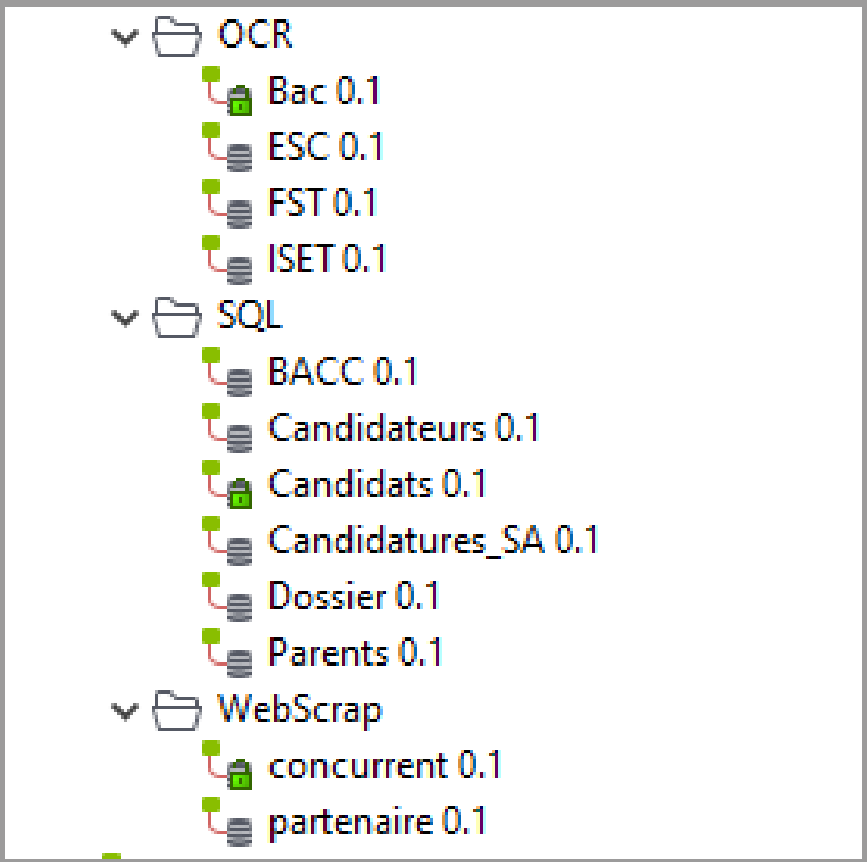
ETL STEPS IN STAGING AREA DATA INGESTION :

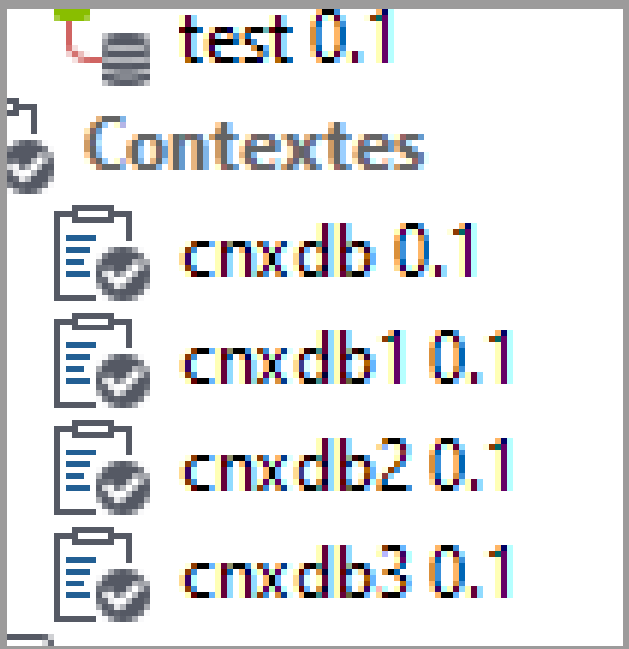
- LOAD INTERNAL FILES (CSV, XLSX)
- PARSE OCR OUTPUT INTO STRUCTURED TABLES SCRAPE AND PARSE HTML INTO JSON/CSV
- DATA STANDARDIZATION NORMALIZE DIPLOMA TYPES, JOB SECTORS, AND INSTITUTION NAMES CONVERT DATES AND SALARY VALUES INTO UNIFIED FORMATS CLEAN FREE TEXT
- TEMPORARY STORAGE STORE INTERMEDIARY TABLES IN SCHEMA STAGING_AREA

STAGINGAREA PROCESS

TALEND
SSMS

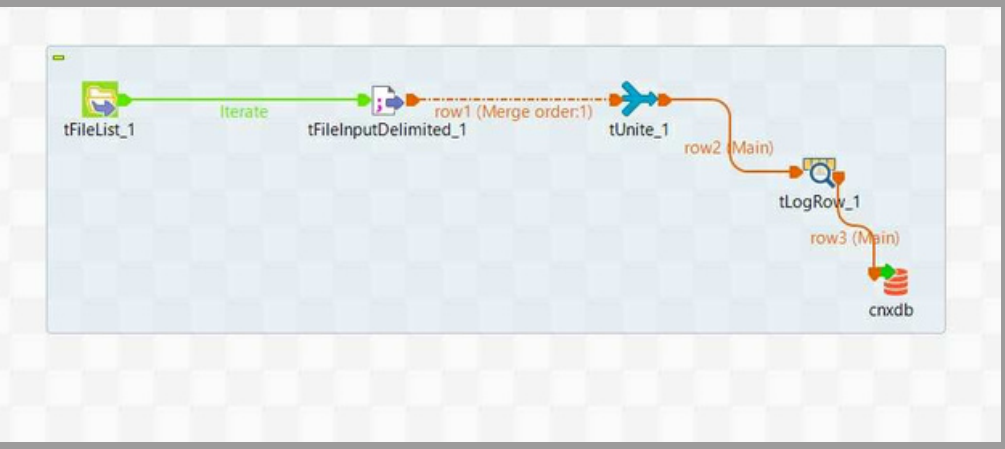


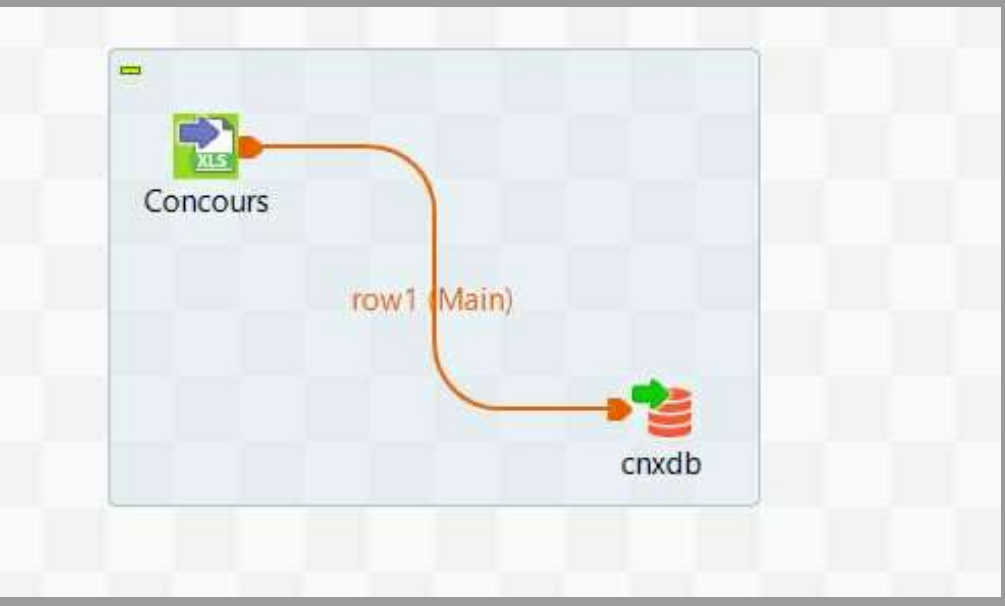


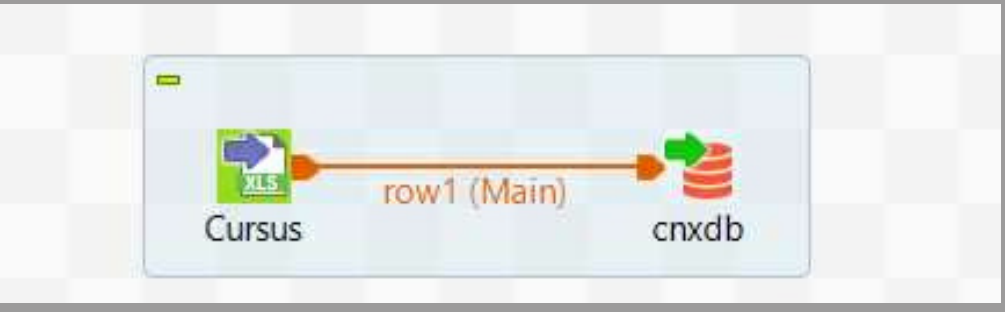


STAGINGAREA PROCESS

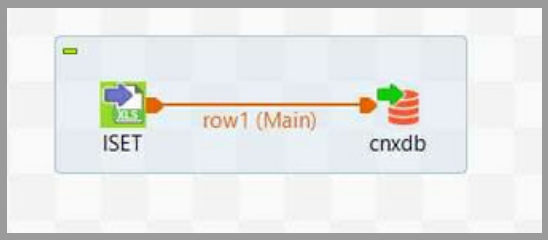
ETL
TRAITEMENT DE DONNÉES

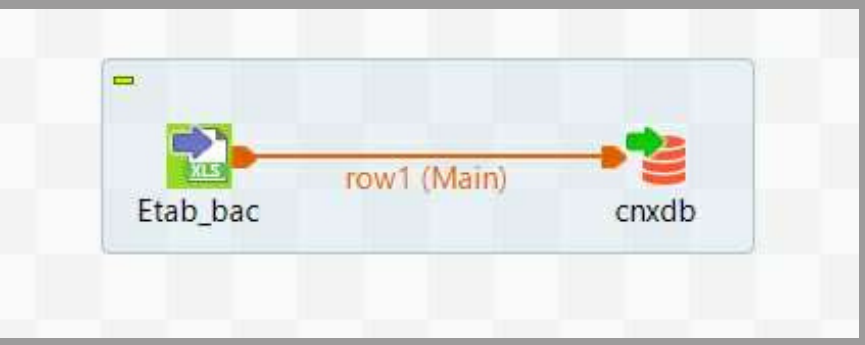


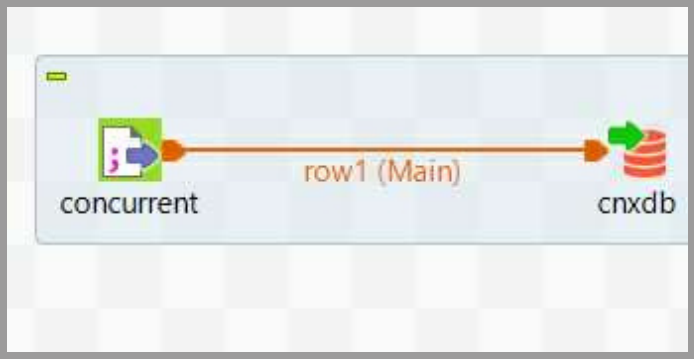




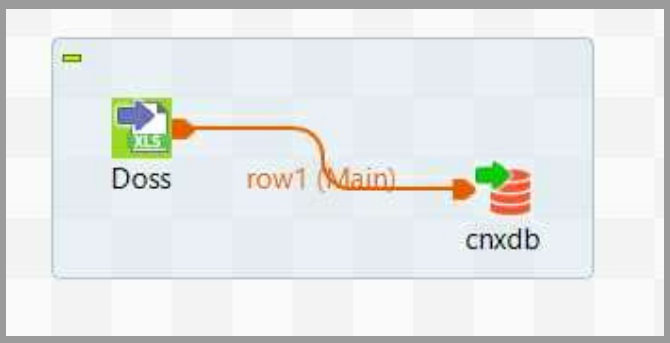

row1 (Main)
Demarage
cnxdb

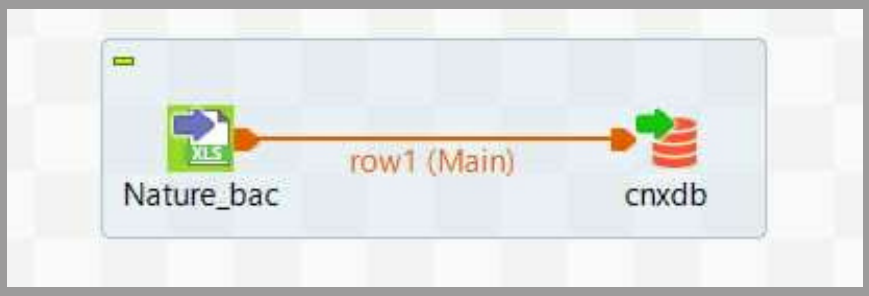

ISET
row1 (Main)
cnxdb

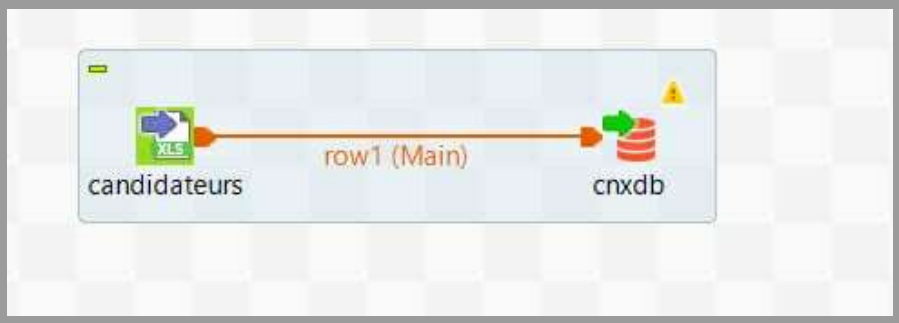

Etab_bac
row1 (Main)
cnxdb


concurrent
row1 (Main)
cnxdb


Dip_spec
row1 (Main)
cnxdb


Doss
row1 (Main)
cnxdb


Nature_bac
row1 (Main)
cnxdb


candidateurs
row1 (Main)
cnxdb

RUN2/RELEASE2

DWH PROCESS

ML SCRIPTS

DL SCRIPT

APPLICATION DEPLOIMENT

DATAWAREHOUSE PROCESS

DATA WAREHOUSE PROCESS – SMARTADMIT THE SMARTADMIT DATA WAREHOUSE (DWH) IS THE BACKBONE OF OUR ANALYTICAL SYSTEM. IT ENSURES THAT DATA FROM MULTIPLE SOURCES IS INTEGRATED, STRUCTURED, AND MADE AVAILABLE FOR ADVANCED ANALYSIS, DASHBOARDING, AND PREDICTIVE MODELING. THE DWH PROCESS FOLLOWS A STRUCTURED PIPELINE, CONSISTING OF FIVE MAIN PHASES:

DATA COLLECTION (SOURCE LAYER) :

- DATA IS GATHERED FROM BOTH INTERNAL AND EXTERNAL SOURCES . EACH SOURCE IS STAGED IN THE STAGING_AREA SCHEMA FOR PREPROCESSING.

DATA CLEANING AND TRANSFORMATION (ETL LAYER):

- RAW DATA IS TRANSFORMED INTO A CLEAN, STRUCTURED FORMAT USING CUSTOM ETL PIPELINES: REMOVE DUPLICATES, NULLS, AND INCORRECT TYPES NORMALIZE TEXTUAL FIELDS , MAP EXTERNAL CODES TO INTERNAL DIMENSIONS GENERATE SURROGATE KEYS FOR STAR SCHEMA INTEGRATION .

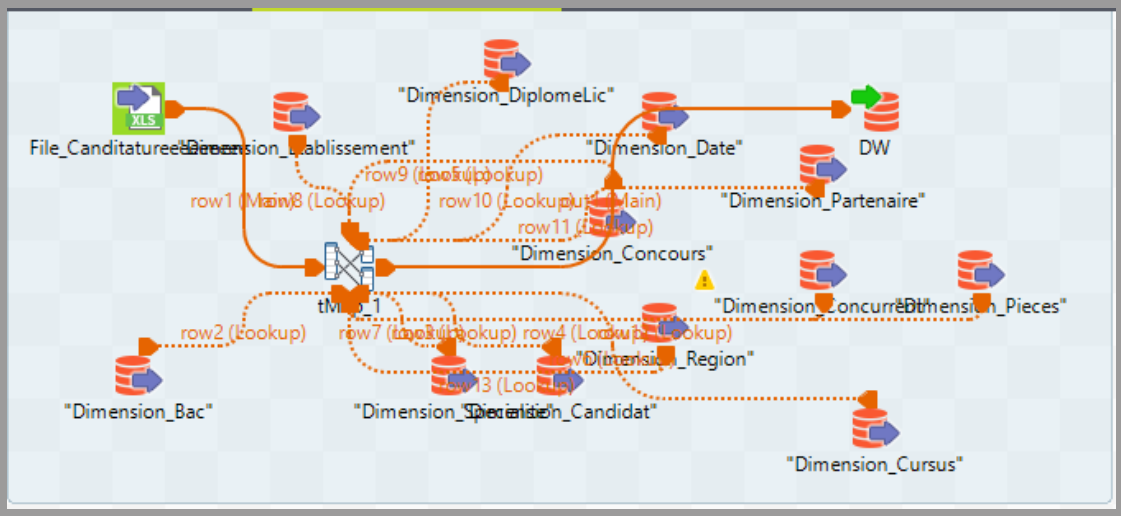
DIMENSIONAL MODELING (LOADING DIMENSIONS):

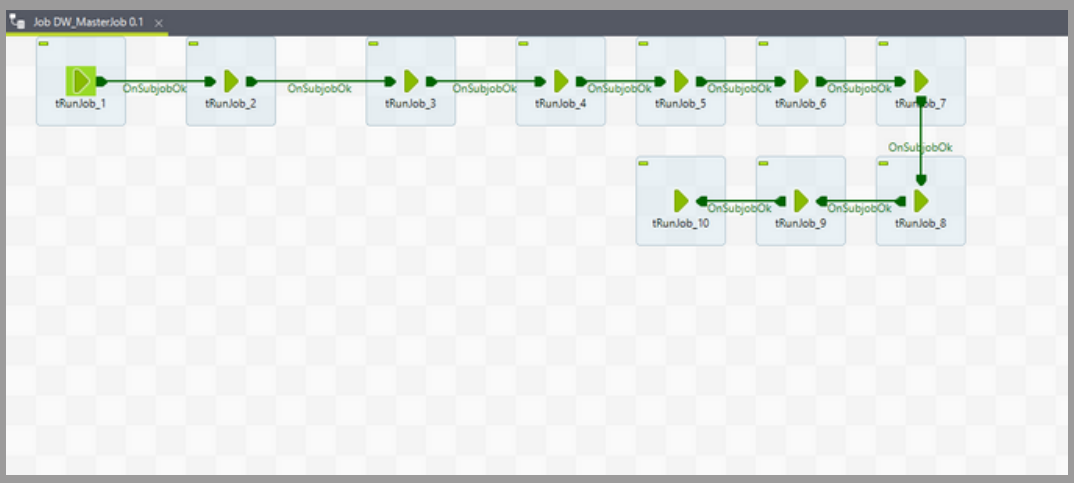
- CLEANED DIMENSION DATA IS LOADED FIRST TO ENSURE REFERENTIAL INTEGRITY

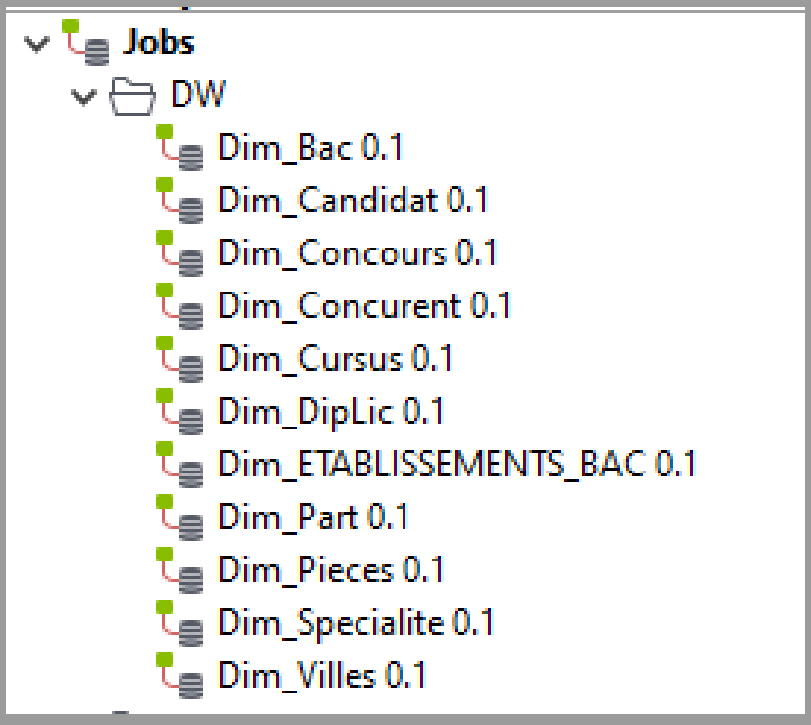
DATA ACCESS AND ANALYSIS (BI LAYER) :

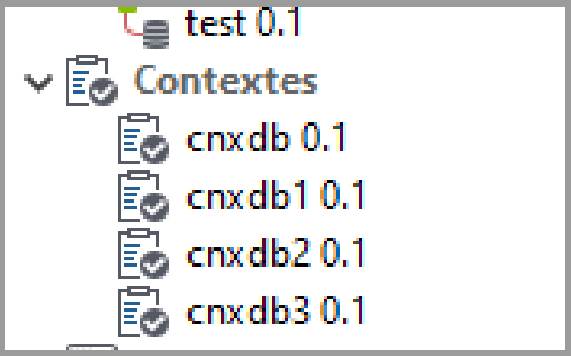
- THE DWH FEEDS THE ANALYTICAL AND VISUALIZATION TOOLS: POWER BI DASHBOARDS: FOR REAL-TIME MONITORING OF ADMISSIONS AND ALUMNI SUCCESS PYTHON PREDICTIVE MODELS: ACCESS DATA VIA POSTGRESQL CONNECTIONS SMARTADMIT PLATFORM: FETCHES INSIGHTS VIA API AND INTEGRATED CHARTS MATERIALIZED VIEWS AND INDEXED TABLES ARE USED FOR PERFORMANCE OPTIMIZATION.

DWH PROCESS


"Dimension_DiplomeLic"
"Dimension_Date"
DW
"Dimension_Partenaire"
"Dimension_Concours"
"Dimension_Pieces"
"Dimension_Region"
"Dimension_Bac"
"Dimension_Cursus"


Job DW_MasterJob 0.1
tRunJob_1
tRunJob_2
tRunJob_3
tRunJob_4
tRunJob_5
tRunJob_6
tRunJob_7
tRunJob_8
tRunJob_9
tRunJob_10
OnSubjobOk


Jobs
DW
Dim_Bac 0.1
Dim_Candidat 0.1
Dim_Concours 0.1
Dim_Concurent 0.1
Dim_Cursus 0.1
Dim_DipLic 0.1
Dim_ETABLISSEMENTS_BAC 0.1
Dim_Part 0.1
Dim_Pieces 0.1
Dim_Specialite 0.1
Dim_Villes 0.1


test 0.1
Contextes
cnxdb 0.1
cnxdb1 0.1
cnxdb2 0.1
cnxdb3 0.1

ML SCRIPTS

1/ REGRESSION LINEAIRE
2/ CLASSIFICATION
3/ CLUSTERING

1/ REGRESSION LINEAIRE

[27]:
```
# 📦 Librairies
import pandas as pd
import numpy as np

# 📥 Chargement du fichier CSV
df = pd.read_csv(r'C:\Users\Acer\Downloads\MOCK_DATA_FINAL_COMPLET.csv')

# 🔍 Aperçu
print("✅ Dimensions :", df.shape)
df.head()
```

```
✅ Dimensions : (1000, 19)
```

[27]:

	id_et	date_preinscrits	date_convocation	date_entretien	date_resultat	date_enreg	date_ne	score_final	moy_bac_et	resultat	id_concours	sexe	Note_Maths	I
0	1	04/04/2023	31/07/2023	02/08/2023	14/09/2023	14/09/2023	23/12/1999	44.52	12.09	Rejected	4	garcon	13.75	
1	2	24/06/2023	16/07/2023	02/08/2023	11/09/2023	14/09/2023	07/07/2000	27.60	17.74	Rejected	4	garcon	19.51	
2	3	30/07/2023	09/08/2023	12/09/2023	14/09/2023	06/05/2024	02/04/2003	46.64	18.63	Rejected	2	fille	17.32	
3	4	29/07/2023	26/08/2023	31/08/2023	14/09/2023	07/01/2024	05/05/2001	57.57	9.69	Waiting List	3	garcon	15.99	
4	5	29/07/2023	14/08/2023	17/08/2023	01/09/2023	14/09/2023	13/02/2001	33.85	6.46	Rejected	3	garcon	11.56	

[28]:
```
# 🎯 Colonnes à utiliser comme features
features = [
    'Note_Maths', 'Note_Physique', 'Note_Info', 'Score_Anglais',
    'Bac_Type', 'Activite_Extra', 'Interet_IA'
]

# 🏁 Variable cible : score final à prédire
target = 'score_final'

# 🧼 Nettoyage : supprimer les lignes incomplètes
df_clean = df[features + [target]].dropna()

print("✅ Données nettoyées :", df_clean.shape)
df_clean.head()
```

```
✅ Données nettoyées : (1000, 8)
```

[28]:

	Note_Maths	Note_Physique	Note_Info	Score_Anglais	Bac_Type	Activite_Extra	Interet_IA	score_final
0	13.75	10.22	8.93	83.64	Technique	Club Robotique	0	44.52
1	19.51	14.50	8.70	89.83	Sciences	Debat	0	27.60
2	17.32	18.48	18.59	62.52	Lettres	Club Robotique	1	46.64
3	15.99	16.79	8.74	81.24	Sciences	Hackathon IA	1	57.57
4	11.56	17.68	9.08	78.59	Economie	Club Robotique	0	33.85

[20]:
```
# Remplir les valeurs manquantes (par exemple, en utilisant la moyenne ou la médiane pour les variables numériques)
df_encoded.fillna(df_encoded.mean(), inplace=True)

# Vérifiez les valeurs manquantes
print(df_encoded.isnull().sum())
```

```
FK_Etudiant                    0
Age                            0
resultat_LISTE ATTENTE         0
resultat_REFUSE                0
moy_bac_et_10,04               0
                              ..
Lieu_Naissance_Zhongchuan      0
Lieu_Naissance_Zéramdine       0
Lieu_Naissance_Évry            0
Lieu_Naissance_Šilherovice     0
Lieu_Naissance_Šuto orizare    0
Length: 4661, dtype: int64
```

[12]:
```
# 📦 Installer si nécessaire : pip install xgboost
from xgboost import XGBRegressor

xgb = XGBRegressor(n_estimators=100, learning_rate=0.1, random_state=42)
xgb.fit(X_train, y_train)

y_pred_xgb = xgb.predict(X_test)

mae_xgb = mean_absolute_error(y_test, y_pred_xgb)
r2_xgb = r2_score(y_test, y_pred_xgb)
```

2/ CLASSIFICATION

```
import pandas as pd
import numpy as np
from sqlalchemy import create_engine
import matplotlib.pyplot as plt
from sklearn.model_selection import train_test_split, RandomizedSearchCV, StratifiedKFold, cross_val_score
from sklearn.preprocessing import StandardScaler, OneHotEncoder
from sklearn.impute import SimpleImputer
from sklearn.compose import ColumnTransformer
from sklearn.pipeline import Pipeline
from imblearn.pipeline import Pipeline as ImbPipeline
from imblearn.over_sampling import SMOTE
from sklearn.ensemble import RandomForestClassifier
from sklearn.metrics import classification_report, ConfusionMatrixDisplay
from sklearn.preprocessing import LabelEncoder

import warnings
warnings.filterwarnings("ignore")

# 1. Connexion à PostgreSQL
user = 'postgres'
password = 'sarra'
host = 'localhost'
port = '5432'
database = 'DWHF'
connection_string = f'postgresql+psycopg2://{user}:{password}@{host}:{port}/{database}'
engine = create_engine(connection_string)

# 2. Requête SQL
query = """
```

```
    fa.moy_bac_et,
    fa.score_final,
    da.concours_result,
    da."Nature_diplome",
    dc.sexe
FROM "Fact_Admissions" fa
JOIN dim_admission_c da ON fa."id_admisson_FK" = da."id_admisson_PK"
JOIN dim_candidats dc ON fa."id_candidat_FK" = dc."id_candidat_PK"
WHERE da.concours_result IS NOT NULL
"""

# 3. Chargement des données
df = pd.read_sql(query, engine)
df['moy_bac_et'] = df['moy_bac_et'].astype(str).str.replace(',', '.').astype(float)
df['score_final'] = df['score_final'].astype(str).str.replace(',', '.').astype(float)

#encodage cible binaire

# Nettoyage
df['concours_result'] = df['concours_result'].str.strip().str.upper()

# Encodage multiclasse
label_encoder = LabelEncoder()
df['result_encoded'] = label_encoder.fit_transform(df['concours_result'])

# Afficher les correspondances
label_mapping = dict(zip(label_encoder.classes_, label_encoder.transform(label_encoder.classes_)))
print("\n Encodage des classes :")
print(label_mapping)
```

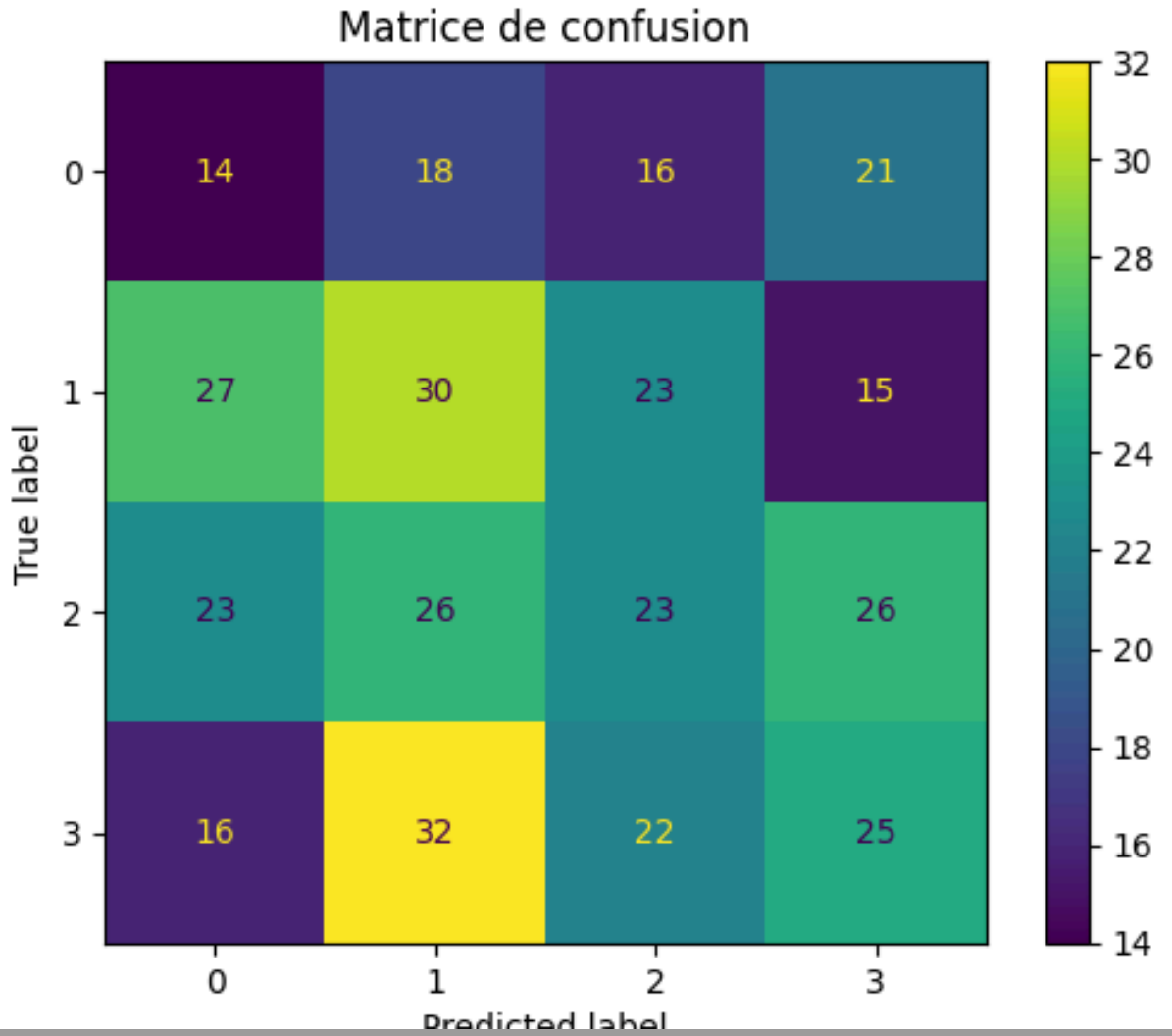
📘 Encodage des classes :
{'ABSENT(E)': np.int64(0), 'ADMIS(E)': np.int64(1), 'LISTE ATTENTE': np.int64(2), 'REFUSE(E)': np.int64(3)}

🔧 Entraînement RandomForest...
Fitting 3 folds for each of 10 candidates, totalling 30 fits

✅ Meilleurs hyperparamètres :
{'clf__n_estimators': 300, 'clf__min_samples_split': 10, 'clf__max_depth': None}

🔍 Rapport de classification :

	precision	recall	f1-score	support
0	0.17	0.20	0.19	69
1	0.28	0.32	0.30	95
2	0.27	0.23	0.25	98
3	0.29	0.26	0.27	95
accuracy			0.26	357
macro avg	0.25	0.25	0.25	357
weighted avg	0.26	0.26	0.26	357

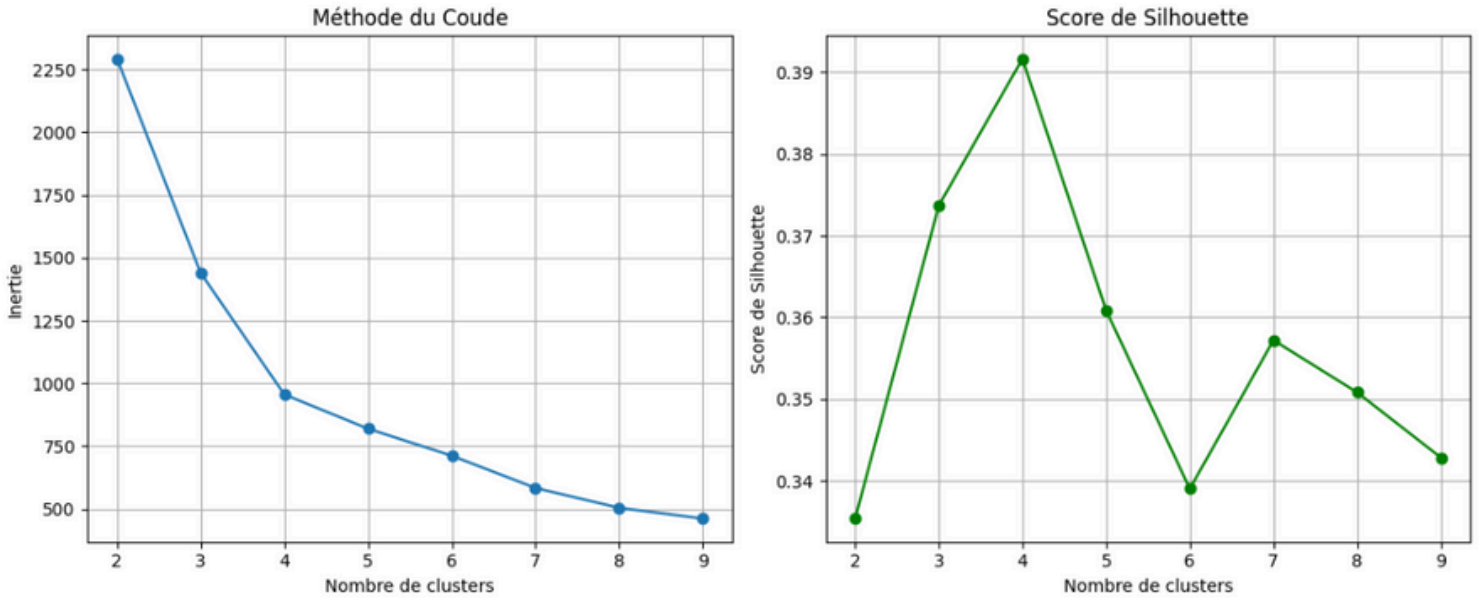


3/CLUSTERING

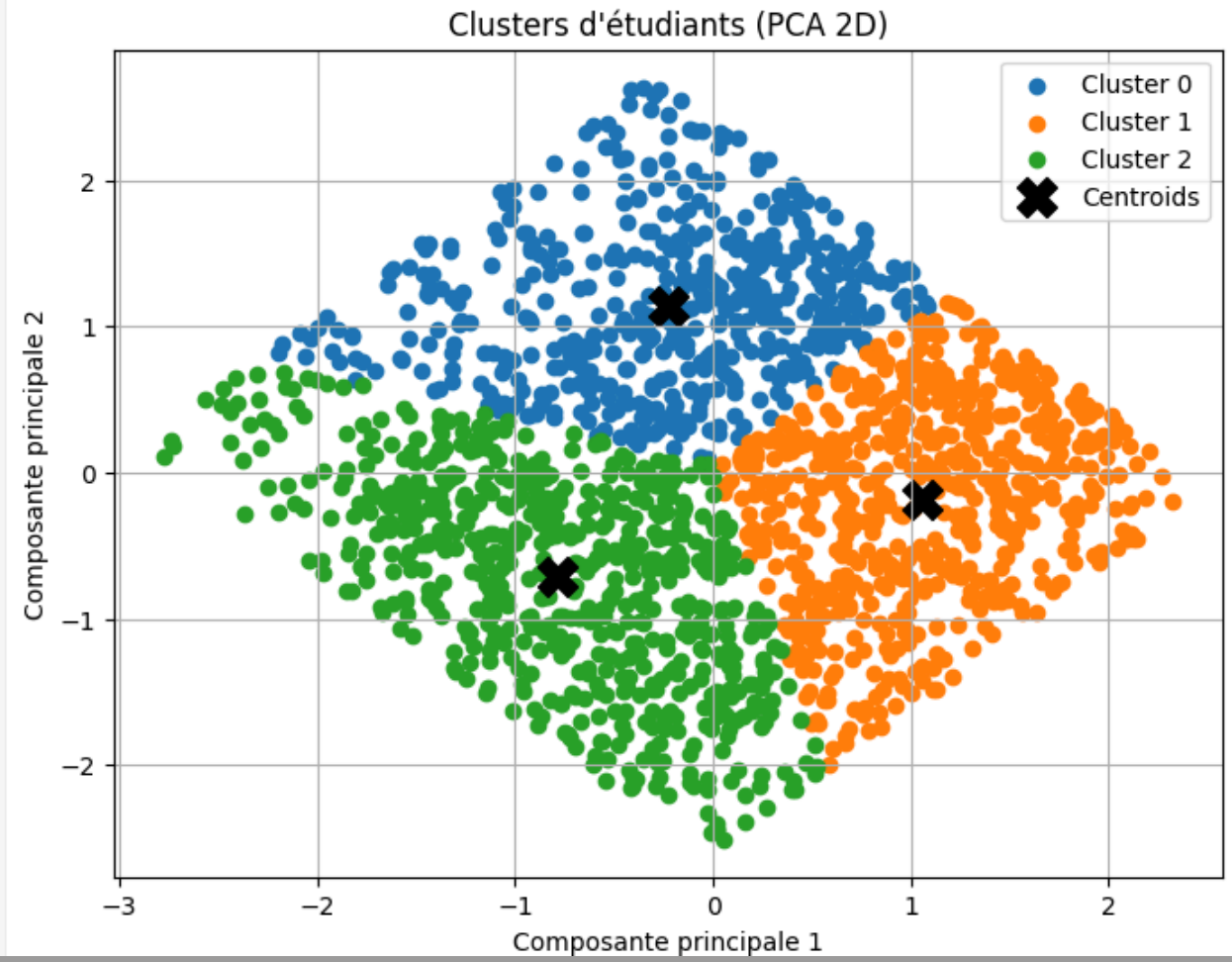
```python
import pandas as pd
import numpy as np
import matplotlib.pyplot as plt
from sklearn.preprocessing import StandardScaler
from sklearn.cluster import KMeans
from sklearn.decomposition import PCA
from sklearn.metrics import silhouette_score

# ================================
# 1️⃣ Utiliser ton DataFrame préparé
# ================================

# df déjà chargé et nettoyé par :
# - score_final
# - moy_bac_et
# - age
# - id_concours
# - nature_bac
# - code_etab_bac
# (date_ne supprimée comme demandé)

# ================================
# 2️⃣ Sélection des features pour clustering
# ================================

features_clustering = [
    'moy_bac_et',  'score_final'
]

X_cluster = df[features_clustering].copy()
```

```python
scaler = StandardScaler()
X_scaled = scaler.fit_transform(X_cluster)

# ================================
# 4️⃣ Méthode du Coude + Silhouette
# ================================

inertias = []
silhouette_scores = []
K_range = range(2, 10)

for k in K_range:
    model = KMeans(n_clusters=k, random_state=42)
    model.fit(X_scaled)
    inertias.append(model.inertia_)
    silhouette_scores.append(silhouette_score(X_scaled, model.labels_))

plt.figure(figsize=(12, 5))

plt.subplot(1, 2, 1)
plt.plot(K_range, inertias, marker='o')
plt.title('Méthode du Coude')
plt.xlabel('Nombre de clusters')
plt.ylabel('Inertie')
plt.grid(True)

plt.subplot(1, 2, 2)
plt.plot(K_range, silhouette_scores, marker='o', color='green')
plt.title('Score de Silhouette')
```

```python
plt.subplot(1, 2, 2)
plt.plot(K_range, silhouette_scores, marker='o', color='green')
plt.title('Score de Silhouette')
plt.xlabel('Nombre de clusters')
plt.ylabel('Score de Silhouette')
plt.grid(True)

plt.tight_layout()
plt.show()

print("👉 Analyse le coude + silhouette score pour choisir le meilleur K.")

# ================================
# 5️⃣ Clustering Final (K choisi)
# ================================

k_final = 3  # <-- à ajuster selon ton graphe du coude !
kmeans = KMeans(n_clusters=k_final, random_state=42)
df['cluster'] = kmeans.fit_predict(X_scaled)

print(f"✅ Clustering terminé avec {k_final} groupes.")

# ================================
# 6️⃣ Visualisation 2D avec PCA
# ================================

pca = PCA(n_components=2)
X_pca = pca.fit_transform(X_scaled)
```



Analyse le coude + silhouette score pour choisir le meilleur K.
Clustering terminé avec 3 groupes.



DL SCRIPT

RECOMONDATION SYSTEM

```python
import pandas as pd
from sklearn.model_selection import train_test_split
from sklearn.preprocessing import OneHotEncoder, MinMaxScaler

# 1. Charger le fichier enrichi
df = pd.read_csv(r'C:\Users\Acer\Downloads\MOCK_DATA_FINAL_COMPLET.csv')

# 2. Features et prétraitement
features = ['Note_Maths', 'Note_Physique', 'Note_Info', 'Score_Anglais', 'Bac_Type', 'Activite_Extra', 'Interet_IA']

X = df[features]

# 3. Encodage One-Hot pour colonnes catégorielles
categorical_cols = ['Bac_Type', 'Activite_Extra']
X = pd.get_dummies(X, columns=categorical_cols)

# 4. Normaliser les notes
scaler = MinMaxScaler()
X[['Note_Maths', 'Note_Physique', 'Note_Info', 'Score_Anglais']] = scaler.fit_transform(X[['Note_Maths', 'Note_Physique', 'Note_Info', 'Score_Anglais']])

# 5. Générer une cible Y simulée pour l'instant (exemple aléatoire)
filiere_possibles = ['Informatique', 'IA', 'Electromecanique', 'Telecom']
import numpy as np
np.random.seed(42)
Y = np.random.choice(filiere_possibles, size=len(df))

# 6. Séparer Train / Test
X_train, X_test, y_train, y_test = train_test_split(X, Y, test_size=0.2, random_state=42)
```

```python
import pandas as pd
# 2. Charger le CSV
df = pd.read_csv(r'C:\Users\Acer\Downloads\MOCK_DATA_FINAL_AVEC_SEXE.csv', sep=',')

print(f"✅ Données chargées depuis le CSV : {df.shape[0]} lignes, {df.shape[1]} colonnes.")
```

```
✅ Données chargées depuis le CSV : 1000 lignes, 12 colonnes.
```

```python
import numpy as np

# Générer des notes réalistes entre 10 et 20
np.random.seed(42)  # pour reproduction des résultats

df['Note_Maths'] = np.random.uniform(10, 20, size=len(df)).round(2)
df['Note_Physique'] = np.random.uniform(8, 20, size=len(df)).round(2)
df['Note_Info'] = np.random.uniform(5, 20, size=len(df)).round(2)
df['Score_Anglais'] = np.random.uniform(50, 100, size=len(df)).round(2)

# Générer Bac type aléatoire
bac_types = ['Sciences', 'Technique', 'Economie', 'Lettres']
df['Bac_Type'] = np.random.choice(bac_types, size=len(df))

# Générer Activités extra-scolaires
activites = ['Club Robotique', 'Hackathon IA', 'Club Sportif', 'Debat', 'Aucune']
df['Activite_Extra'] = np.random.choice(activites, size=len(df))

# Facultatif : détecter "intérêt IA" (0 ou 1 random)
df['Interet_IA'] = np.random.choice([0, 1], size=len(df))
```

[5]:
```
# Vérification rapide
print(df.head())

# Vérifier s'il manque des colonnes critiques
print(df.isnull().sum())
```

```
   id_et date_preinscrits date_convocation date_entretien date_resultat  \
0      1       04/04/2023       31/07/2023     02/08/2023    14/09/2023
1      2       24/06/2023       16/07/2023     02/08/2023    11/09/2023
2      3       30/07/2023       09/08/2023     12/09/2023    14/09/2023
3      4       29/07/2023       26/08/2023     31/08/2023    14/09/2023
4      5       29/07/2023       14/08/2023     17/08/2023    01/09/2023

   date_enreg     date_ne  score_final  moy_bac_et      resultat  id_concours  \
0  14/09/2023  23/12/1999        44.52       12.09      Rejected            4
1  14/09/2023  07/07/2000        27.60       17.74      Rejected            4
2  06/05/2024  02/04/2003        46.64       18.63      Rejected            2
3  07/01/2024  05/05/2001        57.57        9.69  Waiting List            3
4  14/09/2023  13/02/2001        33.85        6.46      Rejected            3

     sexe  Note_Maths  Note_Physique  Note_Info  Score_Anglais   Bac_Type  \
0  garcon       13.75          10.22       8.93          83.64  Technique
1  garcon       19.51          14.50       8.70          89.83   Sciences
2   fille       17.32          18.48      18.59          62.52    Lettres
3  garcon       15.99          16.79       8.74          81.24   Sciences
4  garcon       11.56          17.68       9.08          78.59   Economie

   Activite_Extra  Interet_IA
0  Club Robotique           0
1           Debat           0
```

✅ Dataset prêt pour l'entraînement :
X_train shape: (800, 14)
y_train shape: (800,)

:
```
from sklearn.preprocessing import LabelEncoder

# Créer un encodeur
le = LabelEncoder()

# Adapter sur les données
y_train_encoded = le.fit_transform(y_train)
y_test_encoded = le.transform(y_test)

# Vérifions
print(list(le.classes_))
print(y_train_encoded[:5])
```

```
[np.str_('Electromecanique'), np.str_('IA'), np.str_('Informatique'), np.str_('Telecom')]
[3 1 3 1 0]
```

Model: "sequential_1"

Layer (type)	Output Shape	Param #
dense_3 (Dense)	(None, 128)	1,920
dense_4 (Dense)	(None, 64)	8,256
dense_5 (Dense)	(None, 4)	260

Total params: 10,436 (40.77 KB)

Trainable params: 10,436 (40.77 KB)

Non-trainable params: 0 (0.00 B)

Non-trainable params: 0 (0.00 B)

```
Epoch 1/20
20/20 ━━━━━━━━━━━━━━━━━━━━ 2s 27ms/step - accuracy: 0.3559 - loss: 1.3043 - val_accuracy: 0.5125 - val_loss: 1.2495
Epoch 2/20
20/20 ━━━━━━━━━━━━━━━━━━━━ 0s 10ms/step - accuracy: 0.5607 - loss: 1.1685 - val_accuracy: 0.5562 - val_loss: 1.1386
Epoch 3/20
20/20 ━━━━━━━━━━━━━━━━━━━━ 0s 10ms/step - accuracy: 0.6509 - loss: 1.0270 - val_accuracy: 0.6250 - val_loss: 0.9874
Epoch 4/20
20/20 ━━━━━━━━━━━━━━━━━━━━ 0s 11ms/step - accuracy: 0.7255 - loss: 0.8322 - val_accuracy: 0.6625 - val_loss: 0.8233
Epoch 5/20
20/20 ━━━━━━━━━━━━━━━━━━━━ 0s 10ms/step - accuracy: 0.7881 - loss: 0.6512 - val_accuracy: 0.7437 - val_loss: 0.6825
Epoch 6/20
20/20 ━━━━━━━━━━━━━━━━━━━━ 0s 10ms/step - accuracy: 0.8399 - loss: 0.5369 - val_accuracy: 0.7563 - val_loss: 0.5619
Epoch 7/20
20/20 ━━━━━━━━━━━━━━━━━━━━ 0s 10ms/step - accuracy: 0.8760 - loss: 0.4422 - val_accuracy: 0.7688 - val_loss: 0.4986
Epoch 8/20
20/20 ━━━━━━━━━━━━━━━━━━━━ 0s 12ms/step - accuracy: 0.8837 - loss: 0.3832 - val_accuracy: 0.8250 - val_loss: 0.4333
Epoch 9/20
20/20 ━━━━━━━━━━━━━━━━━━━━ 0s 11ms/step - accuracy: 0.8882 - loss: 0.3446 - val_accuracy: 0.8500 - val_loss: 0.3846
Epoch 10/20
20/20 ━━━━━━━━━━━━━━━━━━━━ 0s 10ms/step - accuracy: 0.9082 - loss: 0.3022 - val_accuracy: 0.8687 - val_loss: 0.3663
Epoch 11/20
20/20 ━━━━━━━━━━━━━━━━━━━━ 0s 12ms/step - accuracy: 0.9242 - loss: 0.2688 - val_accuracy: 0.8625 - val_loss: 0.3398
Epoch 12/20
20/20 ━━━━━━━━━━━━━━━━━━━━ 0s 11ms/step - accuracy: 0.9419 - loss: 0.2581 - val_accuracy: 0.8750 - val_loss: 0.3133
Epoch 13/20
20/20 ━━━━━━━━━━━━━━━━━━━━ 0s 10ms/step - accuracy: 0.9423 - loss: 0.2288 - val_accuracy: 0.8750 - val_loss: 0.3042
Epoch 14/20
20/20 ━━━━━━━━━━━━━━━━━━━━ 0s 10ms/step - accuracy: 0.9454 - loss: 0.2162 - val_accuracy: 0.8813 - val_loss: 0.2873
Epoch 15/20
```

SMARTADMIT APPLICATION DEPLOYMENT STEPS

1. ENVIRONMENT SETUP
CHOOSE CLOUD PROVIDERS .
SET UP SERVERS OR CONTAINERS .
CONFIGURE STAGING AND PRODUCTION ENVIRONMENTS.
2. DATABASE CONFIGURATION
DEPLOY DATABASES (POSTGRESQL, SQL SERVER) .
CONFIGURE ACCESS, BACKUPS, AND SECURITY (ENCRYPTION, ROLES , FIREBASE).

3. BACKEND DEPLOYMENT (FLASK API + MODELS)
PACKAGE THE FLASK API AND INTEGRATE TRAINED ML MODELS.

4. FRONTEND DEPLOYMENT (ANGULAR)
BUILD THE ANGULAR APP FOR PRODUCTION (NG BUILD --PROD).
HOST STATIC FILES ON A WEB SERVER (NGINX) OR CLOUD PLATFORM (FIREBASE).

5. AUTHENTICATION SETUP (FIREBASE)
CONFIGURE AUTHENTICATION PROVIDERS AND USER ROLES.
LINK FIREBASE TO FRONTEND AND BACKEND IF NEEDED.
6. DATA INTEGRATION (TALEND ETL)
DEPLOY TALEND JOBS TO THE TALEND RUNTIME OR A DEDICATED SERVER.
SCHEDULE ETL PROCESSES .
7. API & FRONTEND INTEGRATION
TEST COMMUNICATION BETWEEN FRONTEND, BACKEND, AND EXTERNAL SERVICES.
USE POSTMAN TO VALIDATE API ENDPOINTS.
8. SECURITY & COMPLIANCE
ENFORCE HTTPS AND SECURE API ENDPOINTS.
IMPLEMENT USER ACCESS CONTROL AND GDPR COMPLIANCE.
CONFIGURE FIREWALL RULES AND MONITORING.
9. MONITORING & LOGGING
SET UP APPLICATION MONITORING .
10. TESTING IN PRODUCTION
PERFORM SMOKE TESTS AND REAL-USER TESTING IN THE PRODUCTION ENVIRONMENT.
MONITOR PERFORMANCE AND FIX BUGS IF ANY.
11. FINAL DEPLOYMENT & RELEASE
PUSH FINAL VERSIONS TO PRODUCTION.
ANNOUNCE AVAILABILITY TO STAKEHOLDERS AND TRAIN USERS IF NEEDED.

SMARTADMIT APPLICATION DEPLOYMENT STEPS

Firebase

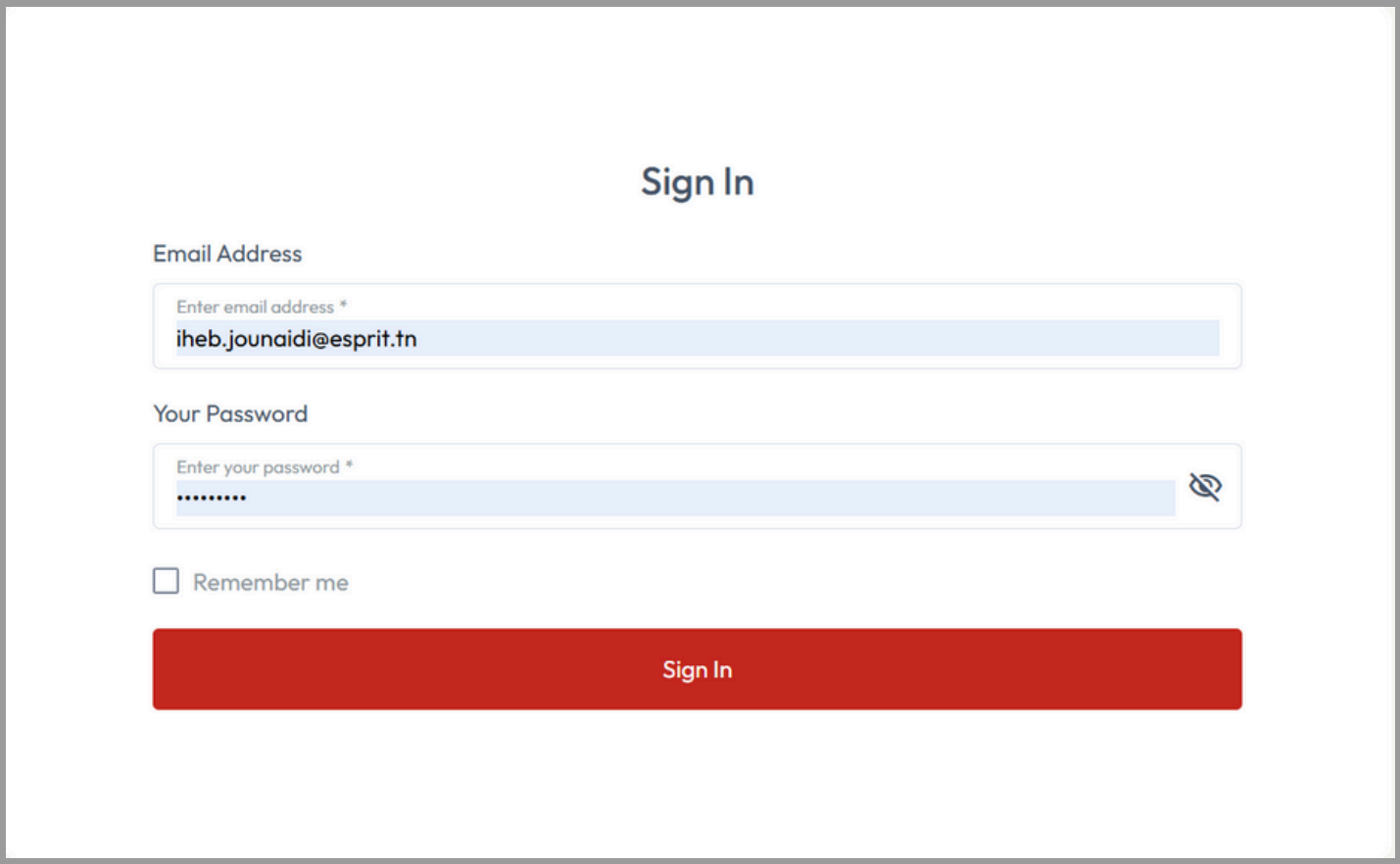
ANGULARJS

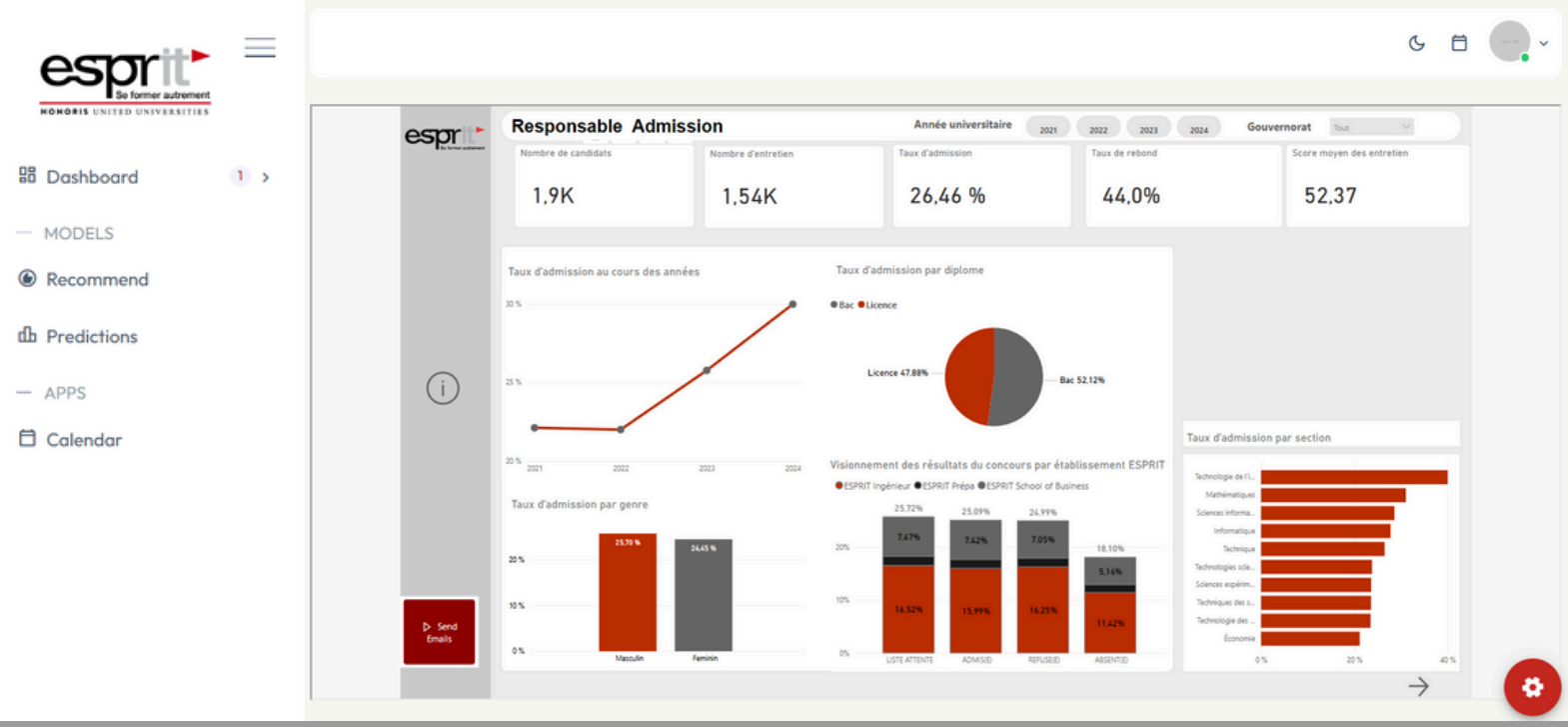
Flask

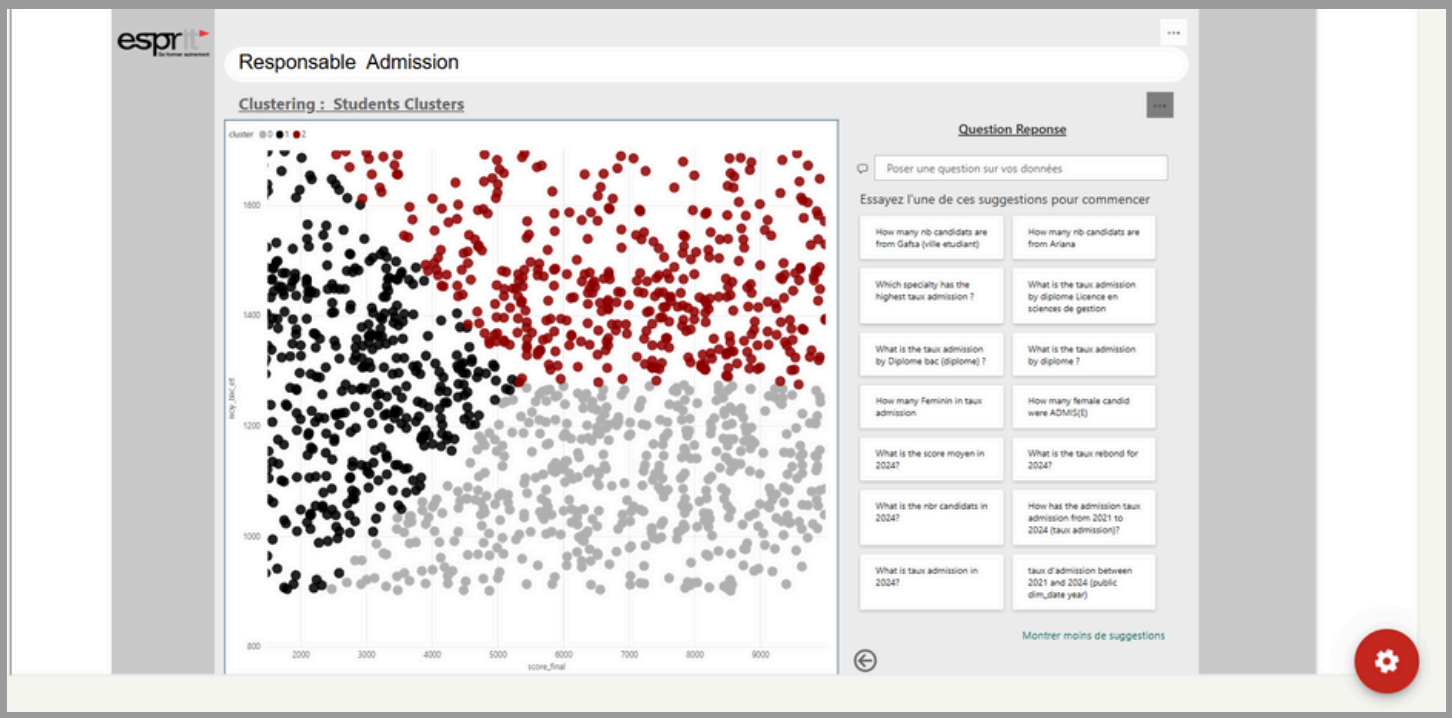
Power BI

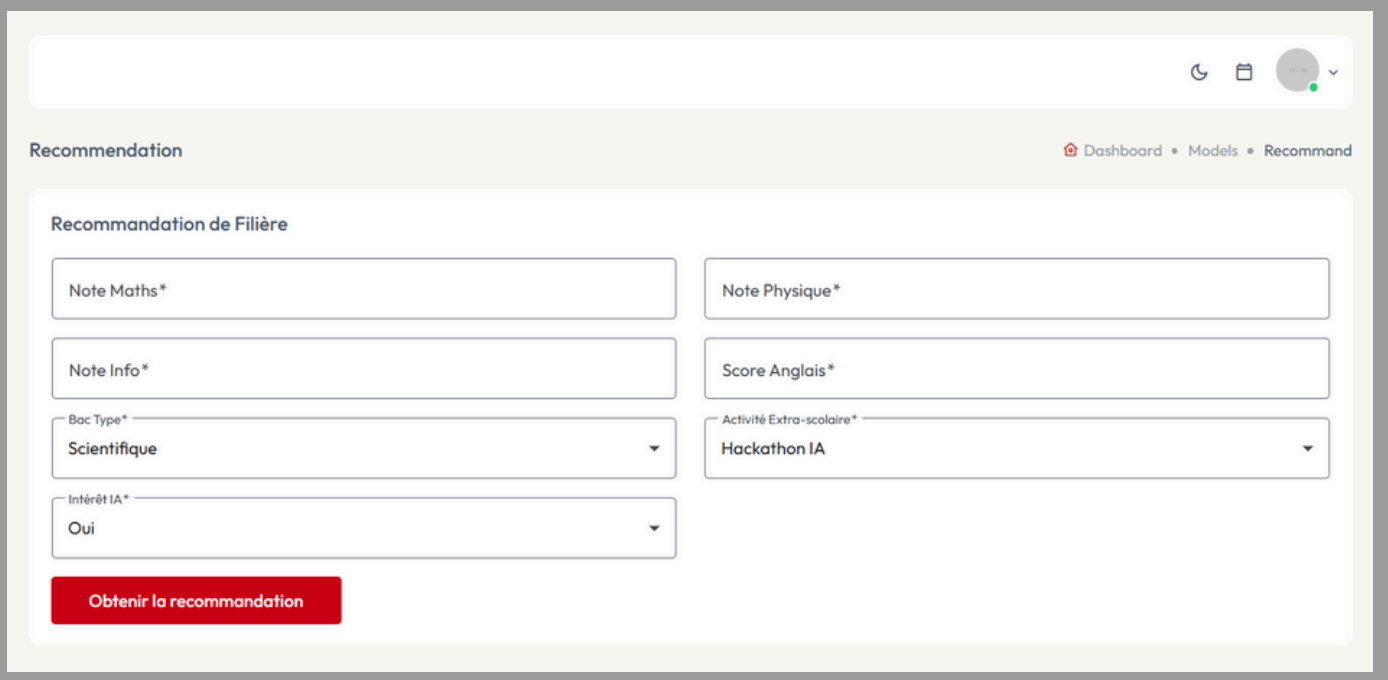
python™

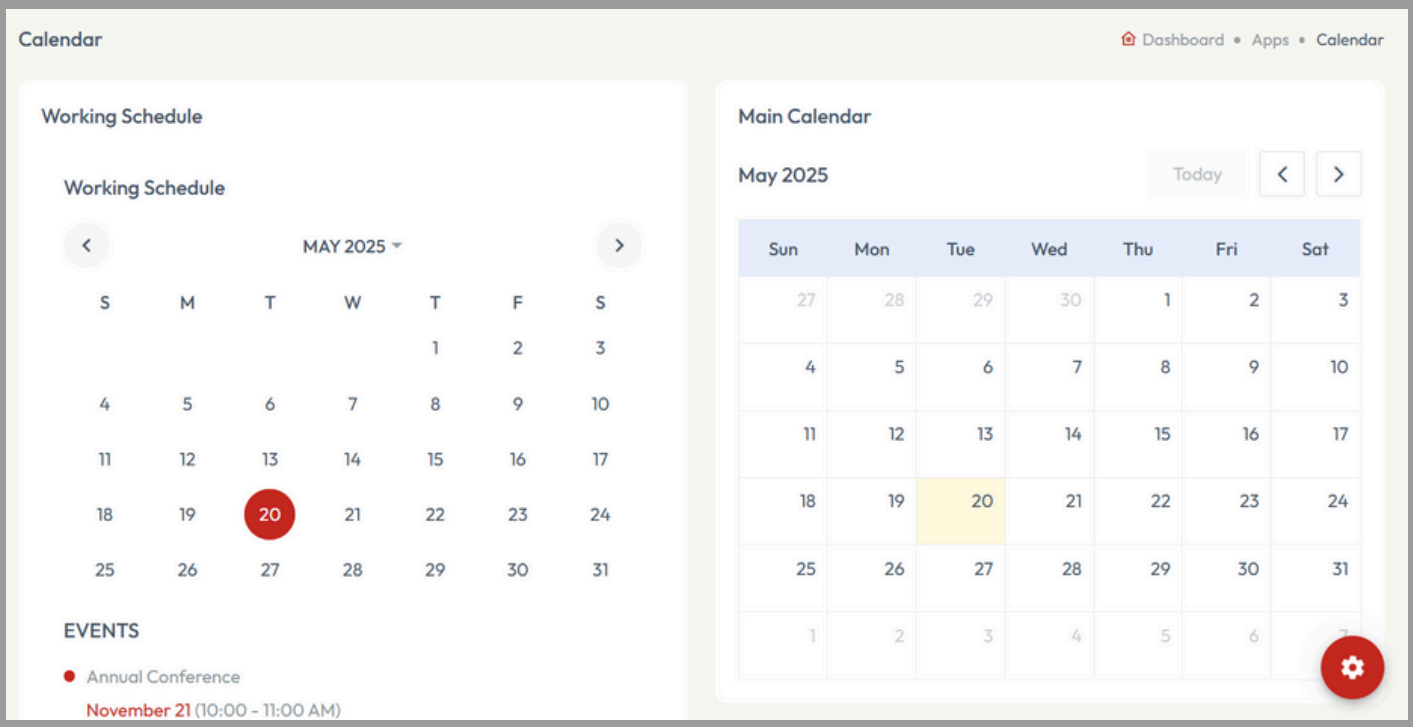
DECISION MAKER : ADMISSION MANAGER



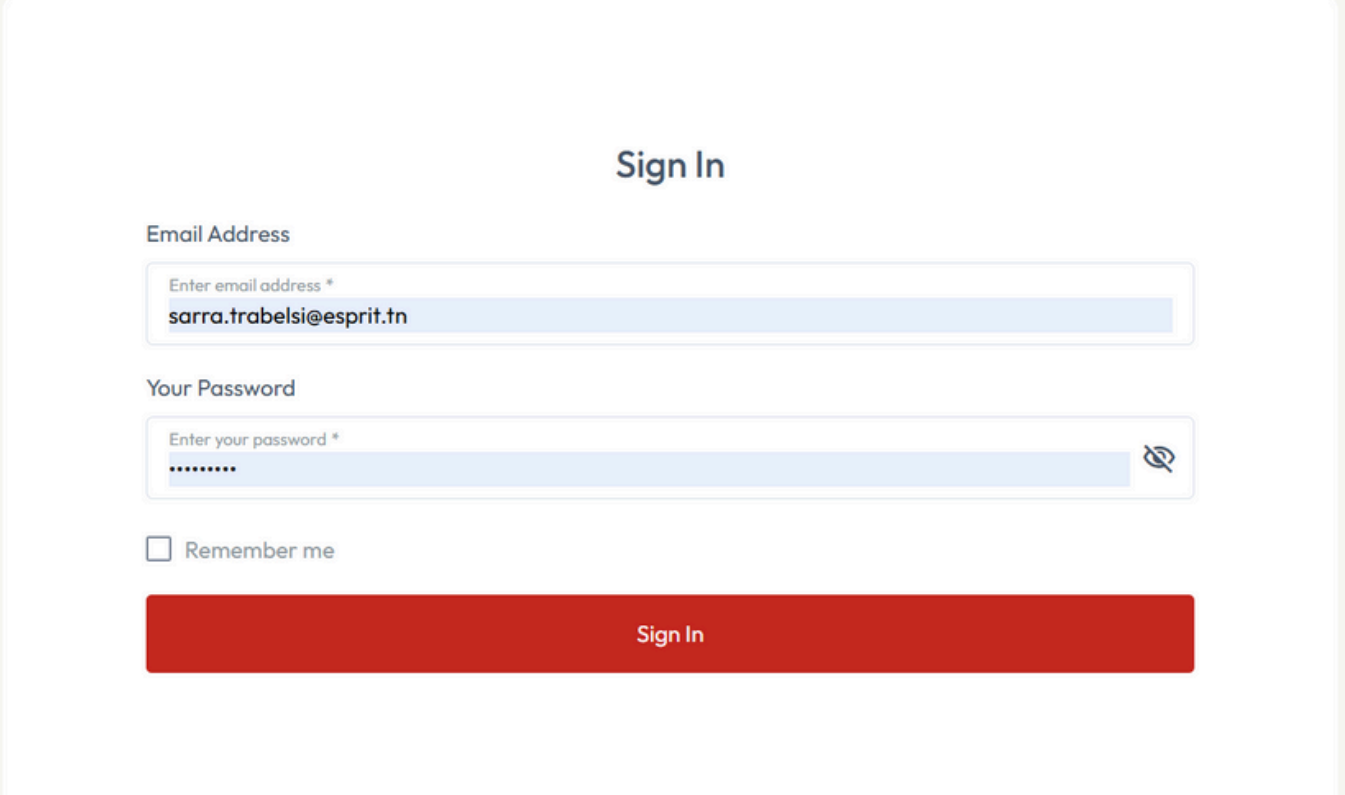

Dashboard
MODELS
Recommend
Predictions
APPS
Calendar
Responsable Admission
Année universitaire
2021
2022
2023
2024
Gouvernorat
Nombre de candidats
1,9K
Nombre d'entretien
1,54K
Taux d'admission
26,46 %
Taux de rebond
44,0%
Score moyen des entretien
52,37
Taux d'admission au cours des années
Taux d'admission par diplome
Bac
Licence
Licence 47.88%
Bac 52.12%
Taux d'admission par section
Visionnement des résultats du concours par établissement ESPRIT
ESPRIT Ingénieur
ESPRIT Prépa
ESPRIT School of Business
Taux d'admission par genre
Masculin
Feminin
Send Emails

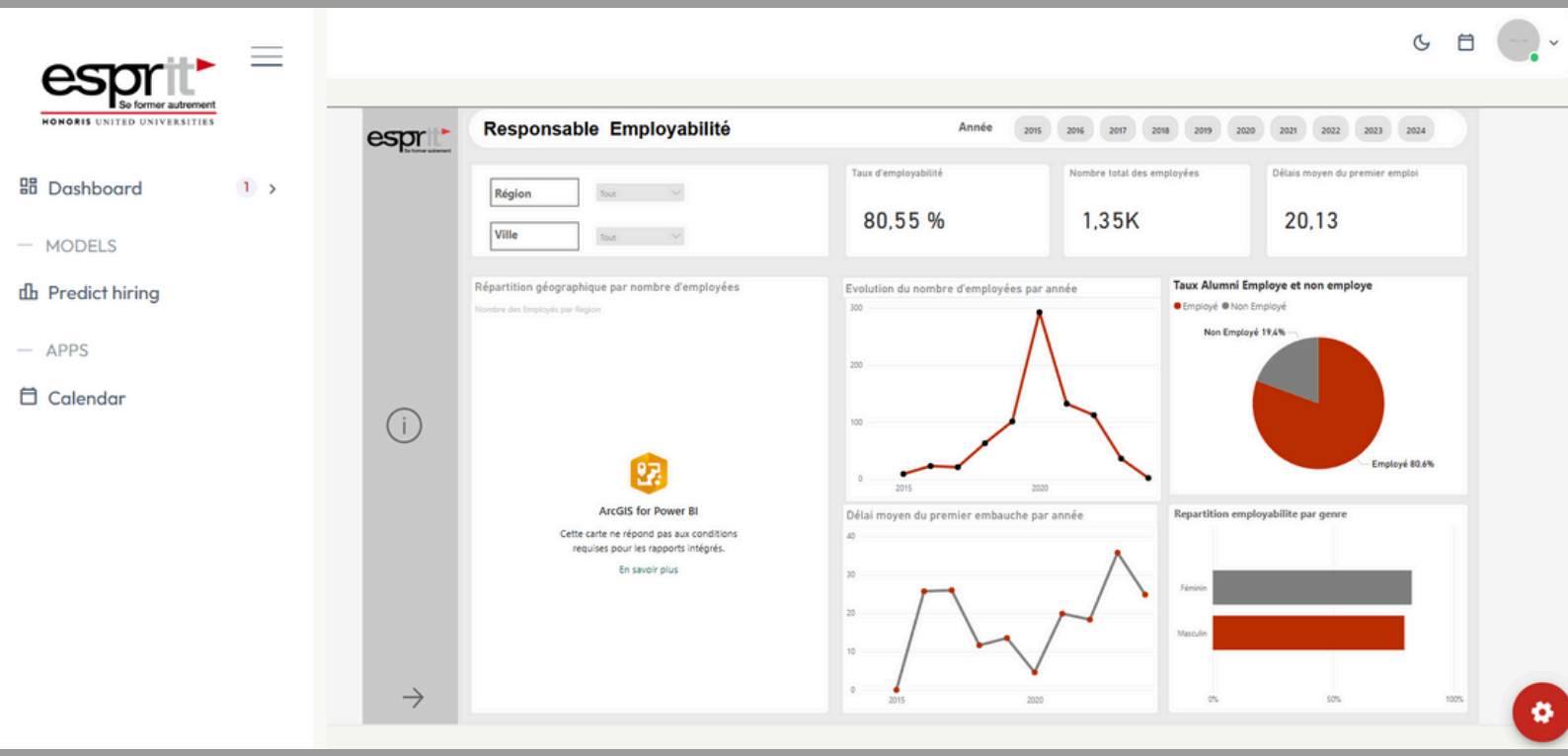

Responsable Admission
Clustering : Students Clusters
Question Reponse
Poser une question sur vos données
Essayez l'une de ces suggestions pour commencer
How many nb candidats are from Gafsa (ville etudiant)
How many nb candidats are from Ariana
Which specialty has the highest taux admission ?
What is the taux admission by diplome Licence en sciences de gestion
What is the taux admission by Diplome bac (diplome) ?
What is the taux admission by diplome ?
How many Feminin in taux admission
How many female candid were ADMIS(E)
What is the score moyen in 2024?
What is the taux rebond for 2024?
What is the nbr candidats in 2024?
How has the admission taux admission from 2021 to 2024 (taux admission)?
What is taux admission in 2024?
taux d'admission between 2021 and 2024 (public dim_date year)
Montrer moins de suggestions
score_final

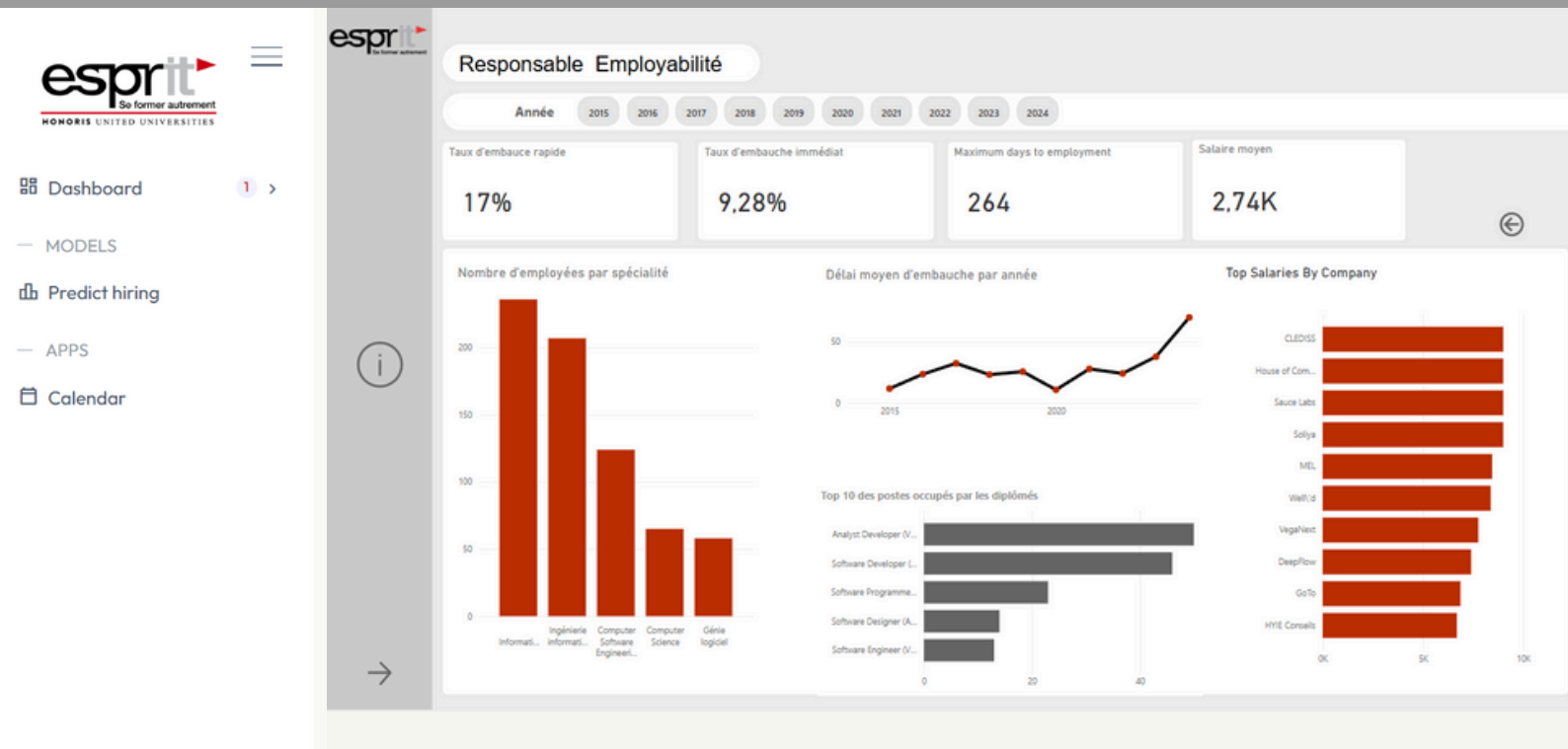

Recommendation
Dashboard • Models • Recommand
Recommandation de Filière
Note Maths*
Note Physique*
Note Info*
Score Anglais*
Bac Type*
Scientifique
Activité Extra-scolaire*
Hackathon IA
Intérêt IA*
Oui
Obtenir la recommandation

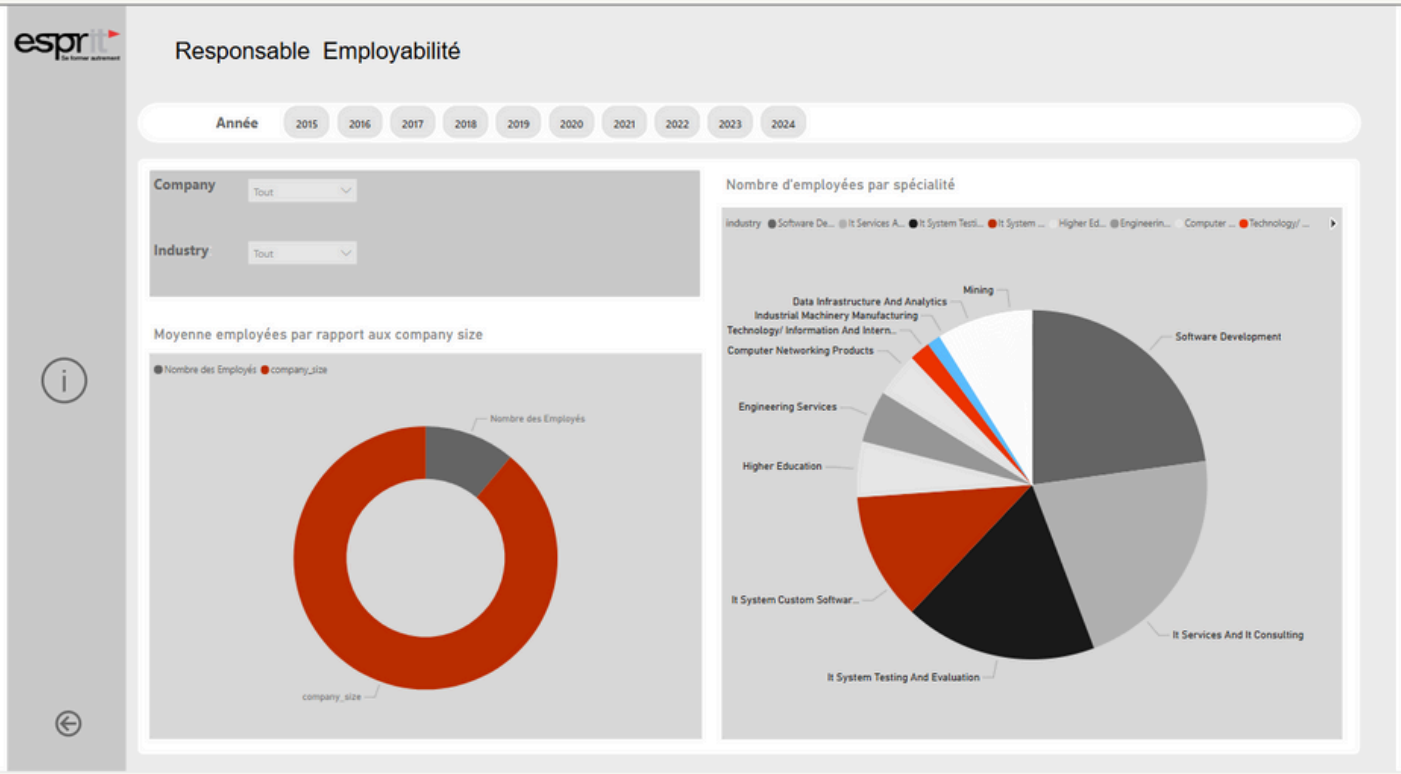

Calendar
Dashboard • Apps • Calendar
Working Schedule
Working Schedule
MAY 2025
S M T W T F S
1 2 3
4 5 6 7 8 9 10
11 12 13 14 15 16 17
18 19 20 21 22 23 24
25 26 27 28 29 30 31
EVENTS
Annual Conference
November 21 (10:00 - 11:00 AM)
Main Calendar
May 2025
Today
Sun Mon Tue Wed Thu Fri Sat
27 28 29 30 1 2 3
4 5 6 7 8 9 10
11 12 13 14 15 16 17
18 19 20 21 22 23 24
25 26 27 28 29 30 31
1 2 3 4 5 6

DECISION MAKER : EMPLOYABILITY MANAGER




esprit
Se former autrement
HONORIS UNITED UNIVERSITIES
Dashboard
MODELS
Predict hiring
APPS
Calendar
Responsable Employabilité
Année
2015 2016 2017 2018 2019 2020 2021 2022 2023 2024
Région
Ville
Tout
Taux d'employabilité
80,55 %
Nombre total des employées
1,35K
Délais moyen du premier emploi
20,13
Répartition géographique par nombre d'employées
ArcGIS for Power BI
Cette carte ne répond pas aux conditions requises pour les rapports intégrés.
En savoir plus
Evolution du nombre d'employées par année
Taux Alumni Employe et non employe
Employé
Non Employé
Non Employé 19,4%
Employé 80,6%
Délai moyen du premier embauche par année
Repartition employabilite par genre
Féminin
Masculin


esprit
Se former autrement
HONORIS UNITED UNIVERSITIES
Dashboard
MODELS
Predict hiring
APPS
Calendar
Responsable Employabilité
Année
2015 2016 2017 2018 2019 2020 2021 2022 2023 2024
Taux d'embauce rapide
17%
Taux d'embauche immédiat
9,28%
Maximum days to employment
264
Salaire moyen
2,74K
Nombre d'employées par spécialité
Délai moyen d'embauche par année
Top Salaries By Company
Top 10 des postes occupés par les diplômés


Responsable Employabilité
Année
2015
2016
2017
2018
2019
2020
2021
2022
2023
2024
Company
Tout
Industry
Tout
Moyenne employées par rapport aux company size
Nombre des Employés
company_size
Nombre d'employées par spécialité
Mining
Data Infrastructure And Analytics
Industrial Machinery Manufacturing
Technology/ Information And Intern...
Computer Networking Products
Engineering Services
Higher Education
It System Custom Softwar...
It System Testing And Evaluation
Software Development
It Services And It Consulting


Prévision des Embauches
Date de début*
jj/mm/aaaa
Date de fin*
jj/mm/aaaa
Lancer la prédiction

Calendar

Dashboard • Apps • Calendar

Working Schedule

Working Schedule

MAY 2025

S	M	T	W	T	F	S
				1	2	3
4	5	6	7	8	9	10
11	12	13	14	15	16	17
18	19	20	21	22	23	24
25	26	27	28	29	30	31

EVENTS

- Annual Conference

  November 21 (10:00 - 11:00 AM)

Main Calendar

May 2025

Today

Sun	Mon	Tue	Wed	Thu	Fri	Sat
27	28	29	30	1	2	3
4	5	6	7	8	9	10
11	12	13	14	15	16	17
18	19	20	21	22	23	24
25	26	27	28	29	30	31
1	2	3	4	5	6	7

CONCLUSION

The SmartAdmit project provided a valuable opportunity to apply data-driven solutions to real-world challenges in university admissions. By combining machine learning, modern web technologies, and labor market data, we developed a scalable and intelligent system tailored to ESPRIT's needs. This project not only enhanced our technical and analytical skills but also demonstrated the impact of AI in improving decision-making and promoting fairer, more efficient admissions processes.