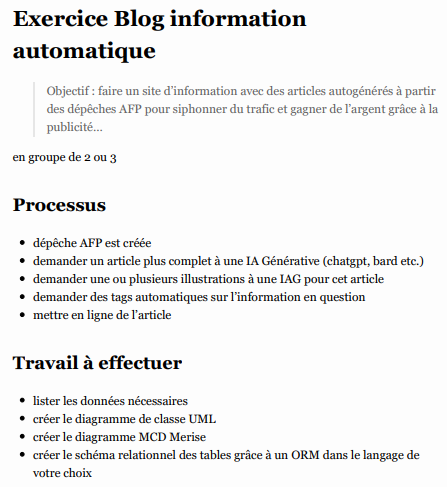
Pierre-Antoine LARGE

Corentin Clero

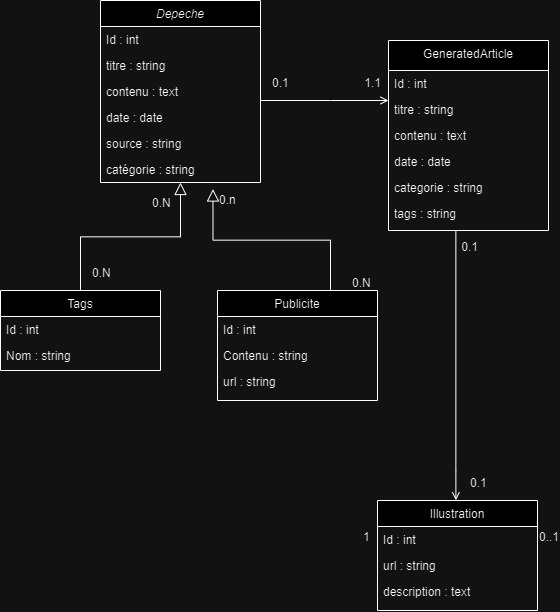
**Enoncé :**

****

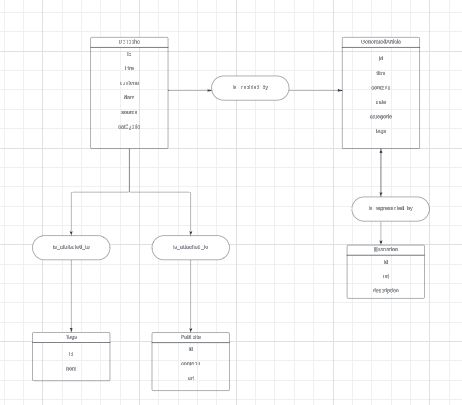
**Lister les donnés :**

* Dépêches AFP :
  + Identifiant de la dépêche
  + Titre
  + Contenu
  + Date de publication
  + Source (AFP)
  + Catégorie (politique, économie, sport, etc.)
* Articles générés :
  + Identifiant de l'article
  + Titre
  + Contenu généré
  + Date de génération
  + Catégorie
  + Tags
* Illustrations :
  + Identifiant de l'illustration
  + URL de l'illustration
  + Description
  + Article associé
* Tags :
  + Identifiant du tag
  + Nom du tag
  + Article associé
* Publicités :
  + Identifiant de la publicité
  + Contenu de la publicité
  + URL de la publicité
  + Article associé

**Diagramme UML :**



**Diagramme MCD :**



**Schéma relationnel (python) :**

from sqlalchemy import create\_engine, Column, Integer, String, Text, Date, ForeignKey

from sqlalchemy.ext.declarative import declarative\_base

from sqlalchemy.orm import relationship, sessionmaker

Base = declarative\_base()

class DepêcheAFP(Base):

\_\_tablename\_\_ = 'depeches\_afp'

id = Column(Integer, primary\_key=True)

titre = Column(String, nullable=False)

contenu = Column(Text, nullable=False)

date = Column(Date, nullable=False)

source = Column(String, default="AFP")

categorie = Column(String, nullable=False)

class ArticleGenere(Base):

\_\_tablename\_\_ = 'articles\_genere'

id = Column(Integer, primary\_key=True)

titre = Column(String, nullable=False)

contenu = Column(Text, nullable=False)

date = Column(Date, nullable=False)

categorie = Column(String, nullable=False)

tags = Column(String)

depeche\_id = Column(Integer, ForeignKey('depeches\_afp.id'))

depeche = relationship('DepêcheAFP', back\_populates='articles')

class Illustration(Base):

\_\_tablename\_\_ = 'illustrations'

id = Column(Integer, primary\_key=True)

url = Column(String, nullable=False)

description = Column(Text)

article\_id = Column(Integer, ForeignKey('articles\_genere.id'))

article = relationship('ArticleGenere', back\_populates='illustrations')

class Tags(Base):

\_\_tablename\_\_ = 'tags'

id = Column(Integer, primary\_key=True)

nom = Column(String, nullable=False)

article\_id = Column(Integer, ForeignKey('articles\_genere.id'))

article = relationship('ArticleGenere', back\_populates='tags')

class Publicite(Base):

\_\_tablename\_\_ = 'publicites'

id = Column(Integer, primary\_key=True)

contenu = Column(Text, nullable=False)

url = Column(String)

article\_id = Column(Integer, ForeignKey('articles\_genere.id'))

article = relationship('ArticleGenere', back\_populates='publicites')

DepêcheAFP.articles = relationship('ArticleGenere', order\_by=ArticleGenere.id, back\_populates='depeche')

ArticleGenere.illustrations = relationship('Illustration', order\_by=Illustration.id, back\_populates='article')

ArticleGenere.tags = relationship('Tags', order\_by=Tags.id, back\_populates='article')

ArticleGenere.publicites = relationship('Publicite', order\_by=Publicite.id, back\_populates='article')

# Création de la base de données

engine = create\_engine('sqlite:///blog\_information.db')

Base.metadata.create\_all(engine)

# Création d'une session

Session = sessionmaker(bind=engine)

session = Session()