

# MINISTERE DE L'ENSEIGNEMENT SUPERIEUR ET DE LA RECHERCHE SCIENTIFIQUE Institut supérieur de Gestion de Tunis \*\*\*\*\*\*\*\*



# RAPPORT DU PROJET:GESTION D'UNE CONFERENCE SCIENTIFIQUE

ELABORE PAR:
 MOHAMMED ARBI GHANMI
 BOUDHRAA INTISSAR
 2BIS1

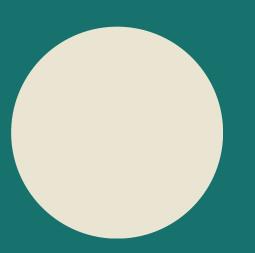
## PLAN

O1 introductin

O2 Conception et Modélisation

03 Développement









## INTRODUCTION:

L'objectif de ce projet est de développer une de gestion de conférence application scientifique qui facilite la coordination et l'organisation de tels événements. L'application sera utilisée par l'administrateur (le président de la conférence), les membres du comité scientifique et les utilisateurs (participants, auteurs, etc.). Voici un aperçu du le développement de plan pour cette application:



## CONCEPTION ET MODÉLISATION:

#### **ANALYSE DES BESOINS:**



## Administrateur (Président de la Conférence) :

- Créer, modifier et supprimer des conférences.
- Gérer les informations des conférenciers invités.
- Affecter les articles soumis aux membres du comité scientifique pour évaluation.
- Consulter l'état des articles soumis.
- Consulter les informations sur les participants inscrits.
- Consulter les statistiques générales de la conférence (nombre total d'articles soumis, taux d'acceptation, etc.).
- Imprimer l'appel à participation et la liste des participants inscrits.



#### Membres du Comité Scientifique :

- Consulter les informations sur la conférence.
- Consulter les articles qui leur ont été affectés pour évaluation.
- Évaluer les articles soumis (accepter ou rejeter).
- Consulter les informations sur les conférenciers invités.
- Consulter les statistiques générales de la conférence.

## CONCEPTION ET MODÉLISATION:

#### **ANALYSE DES BESOINS:**



## **Utilisateurs (Auteurs d'Articles et Participants):**

- Consulter les informations sur la conférence.
- Soumettre des articles avant la date limite de soumission.
- Suivre l'état de leurs articles soumis.
- S'inscrire à la conférence avant la date limite d'inscription.
- Payer les frais d'inscription selon les modalités spécifiées.
- Consulter les informations sur les conférenciers invités.
- Consulter les statistiques générales de la conférence.

## RELATIONS ENTRE LES TABLES:

- Les tables article, auteur, inscription et utilisateur sont liées à la table conference via l'ID de la conférence (id\_conf).
- Les tables **article** et **utilisateur** sont liées via l'ID de l'utilisateur (**id\_user**).
- Les tables **auteur** et **article** sont liées via l'ID de l'article (**id\_article**).
- Les tables **conferencier** et **conference** sont liées via l'ID de la conférence (**id\_conf**).

Ces tables et leurs relations fournissent une base solide pour stocker et gérer toutes les informations nécessaires à la gestion d'une conférence scientifique.

#### **Table utilisateur:**

- ette table stocke les informations sur les utilisateurs de l'application.
- Chaque utilisateur a un identifiant unique (id\_user).
- Les types d'utilisateurs sont enregistrés dans la colonne **type**.
- Les autres informations telles que le nom, prénom, email, etc., sont également stockées.

#### **Table article**

- Cette table contient les détails des articles soumis pour la conférence.
- Chaque article a un identifiant unique (id\_article).
- L'état de l'article (**etat**) est enregistré sous forme d'énumération (NA, UE, ACC, REJ).
- L'ID de l'utilisateur qui a soumis l'article est stocké dans id\_user.

#### table auteur:

- Cette table relie les auteurs aux articles soumis.
- Chaque auteur a un identifiant unique (id\_auteur).
- Les informations sur l'auteur (nom, prénom, email, institution) sont stockées.
- LLes liens avec l'article soumis et la conférence sont établis via id article et id\_conf



#### **Table conference:**

- ette table stocke les informations sur les utilisateurs de Cette table contient les détails des conférences organisées.
- Chaque conférence a un identifiant unique (id\_conf).
- Les informations telles que le titre, le président, l'institution organisatrice, la date, le lieu, etc., sont stockées.
- Les sujets de la conférence sont également enregistrés dans la colonne topics.

#### **Table conferencier:**

- Cette table stocke les informations sur les conférenciers invités pour une conférence donnée.
- Chaque conférencier a un identifiant unique (id\_conferencier).
- Les détails tels que le nom, le titre de présentation, l'institution et le pays d'origine sont enregistrés.

#### **Table inscription:**

- Cette table contient les informations sur les participants inscrits à une conférence.
- Chaque participant a un identifiant unique (id\_inscrit).
- Les détails tels que le nom, l'email, l'institution, la méthode de paiement sont enregistrés.
- Les liens avec l'utilisateur et la conférence sont établis via id\_user et id\_conf.

## **DÉVELOPPEMENT:**

#### 1. CRÉATION D'UNE NOUVELLE CONFÉRENCE :

private void jButton3ActionPerformed(java.awt.event.ActionEvent evt) { return; // Sortir de la méthode sans effectuer l'inscription String institution = jTextFieldl.getText(); String methodePaiement = (String) jComboBox1.getSelectedItem(); // Récupérer l'email de l'utilisateur String emailUtilisateur = getEmailUtilisateur(idUtilisateur); // Remplacer par l'email de l'utilisateur récupéré de la base de données // Vérifier si l'email de l'utilisateur est valide if (emailUtilisateur == null) { JOptionPane.showMessageDialog(this, "Impossible de récupérer l'email de l'utilisateur.", "Erreur", JOptionPane.ERROR\_MESSAGE); return; if (emailExisteDansInscription(emailUtilisateur)) { JOptionPane.showMessageDialog(this, "L'utilisateur est déjà inscrit.", "Erreur", JOptionPane.ERROR MESSAGE); MenuUtilisateur NewWindow = new MenuUtilisateur(); NewWindow.setVisible(true); // Insérer l'inscription dans la base de données insererInscription(nomPrenomUtilisateur, emailUtilisateur, institution, methodePaiement); // Retourner à la page MenuUtilisateur MenuUtilisateur menuUtilisateur = new MenuUtilisateur(); menuUtilisateur.setVisible(true); this.dispose(): private void jButton6ActionPerformed(java.awt.event.ActionEvent evt) {

MenuUtilisateur NewWindow = new MenuUtilisateur();

NewWindow satVisible(true)

- Permettre à l'administrateur de créer une nouvelle conférence en saisissant toutes les informations nécessaires.
- Implémenter les fonctionnalités de modification et de suppression des conférences.

#### 2 GESTION DES CONFÉRENCIERS INVITÉS :

- Permettre à l'administrateur d'ajouter, consulter, supprimer ou modifier des conférenciers invités.
- Implémenter les fonctionnalités de gestion des informations des conférenciers invités.

```
private void jButton2ActionPerformed(java.awt.event.ActionEvent ext) {
     int selectedIndex = jList1.getSelectedIndex();
if (selectedIndex != -1) {
      // An item is selected, retrieve id article
     String selectedItem = jList1.getSelectedValue();
        System.out.println("ligne: "+selectedItem);
     String[] parts = selectedItem.split(" ");
       System.out.println("partie: "+parts[2])
     String etat = selectedItem.substring(selectedItem.lastIndexOf(" ") );
       System.out.println("last index: "+selectedItem.lastIndexOf(" "));
      etat = etat.trim();
       System.out.println("etat: "+etat);
      if(!("Id".equals(parts[0].trim()))){
     if("NA".equals(etat)){
      idArticle = Integer.parseInt(parts[0]);
      // Clear the list
     DefaultListModel<String> listModel = new DefaultListModel<>();
      jList1.setModel(listModel);
      // Populate the list with data of users who have type "com sc"
      try (Connection connection = DriverManager.getConnection(JDBC_URL, USERNAME, PASSWORD)
           Statement statement = connection.createStatement();
           ResultSet resultSet = statement.executeQuery("SELECT * FROM utilisateur WHERE typ
          listModel.addElement("Id
                                                                                   Nom
          while (resultSet.next()) {
             // Retrieve user information
              int userId = resultSet.getInt("id user");
              String nom = resultSet.getString("nom");
              String prenom = resultSet.getString("prenom");
              // Add the user information to the list
              listModel.addElement(userId +
          jList1.setModel(listModel); // Set the populated model to the JList
          jButton3.setVisible(true);
```

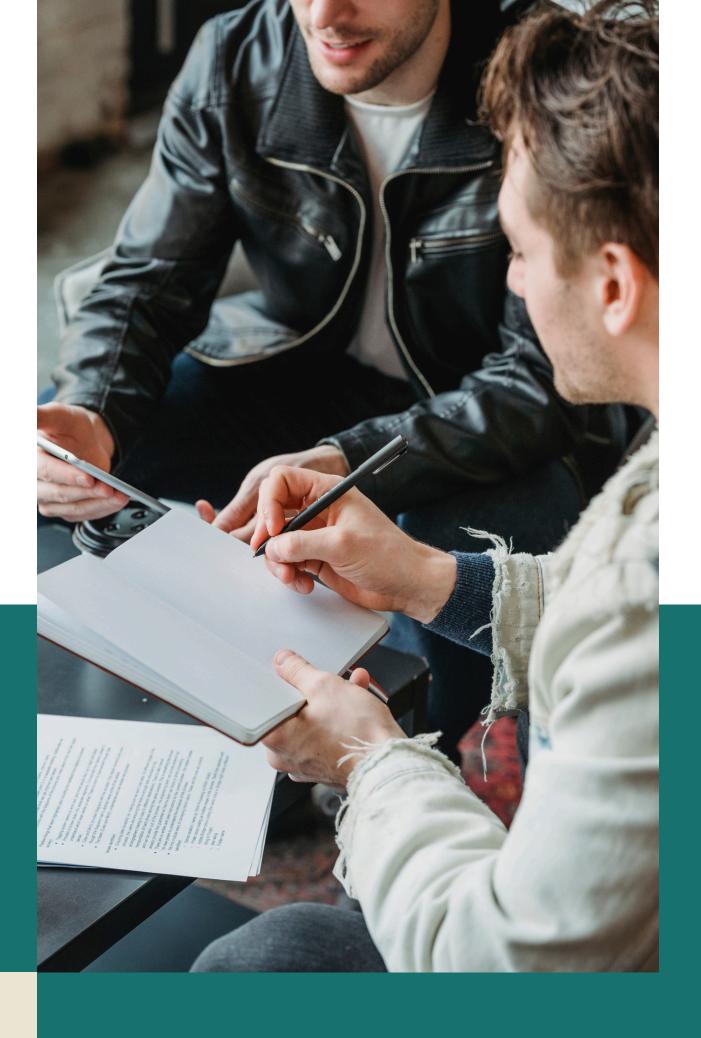
## **DÉVELOPPEMENT:**

#### • 3 SOUMISSION D'ARTICLES:

- Permettre aux utilisateurs (auteurs) de soumettre des articles en fournissant toutes les informations requises.
- Mettre en place le suivi des articles soumis et de leur état (non attribué, évaluation en cours, accepté, rejeté).

#### • 4 ÉVALUATION DES ARTICLES :

- Permettre aux membres du comité scientifique d'évaluer les articles qui leur ont été attribués.
- Mettre en place un système de décision pour chaque article (acceptation ou rejet).



## **DÉVELOPPEMENT:**

- 5 INSCRIPTIONS DES PARTICIPANTS :
- Permettre aux utilisateurs de s'inscrire à la conférence en fournissant leurs informations personnelles et de paiement.
- Implémenter les vérifications de date limite pour l'inscription.