

Projet Algorithmique et programmation java avec Mme AUBONNET Tatiana



STAG



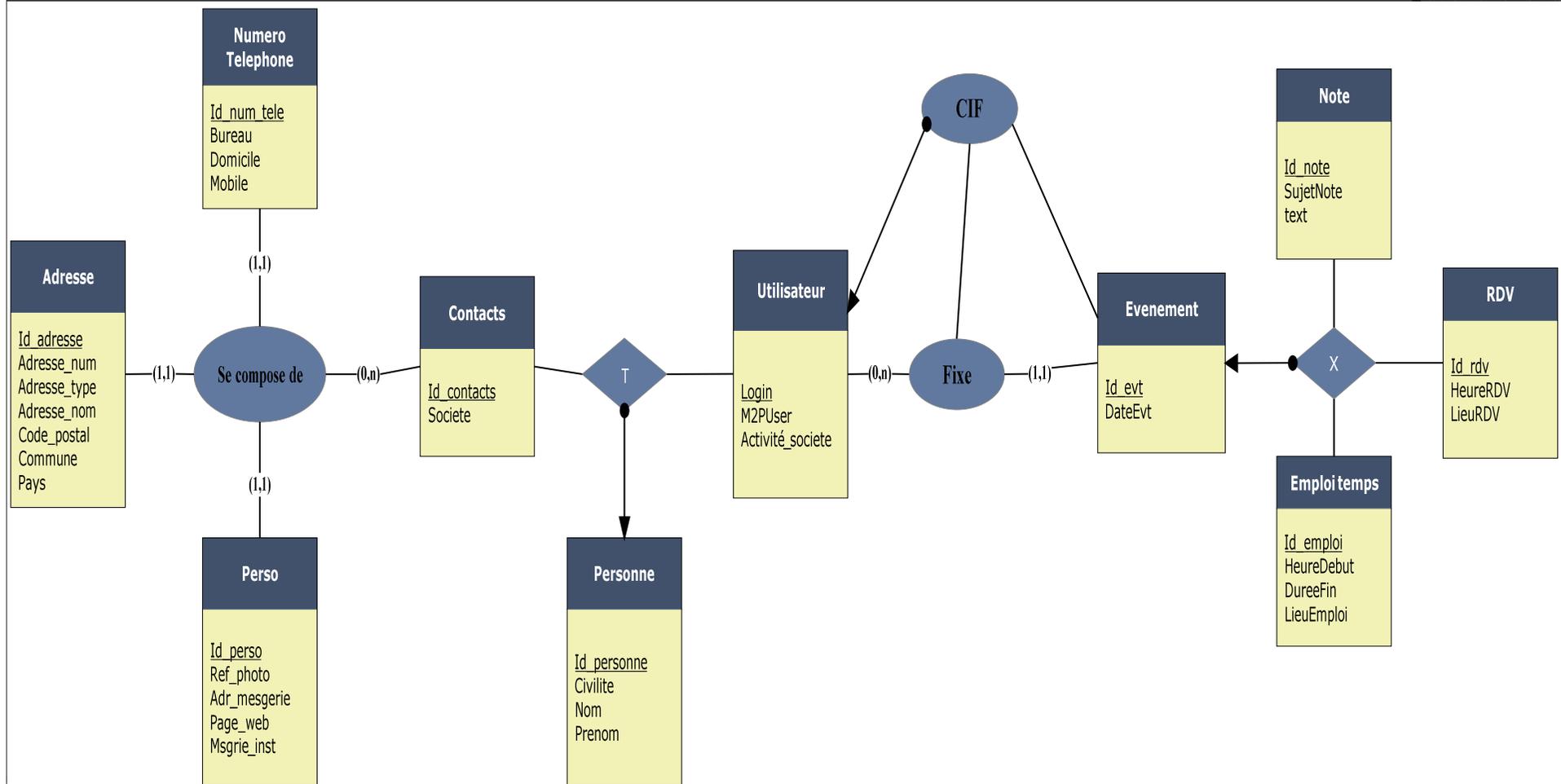
INTRODUCTION

STAG est une application qui gère:

- Les contacts
- Les rendez-vous
- Les notes
- Calendrier

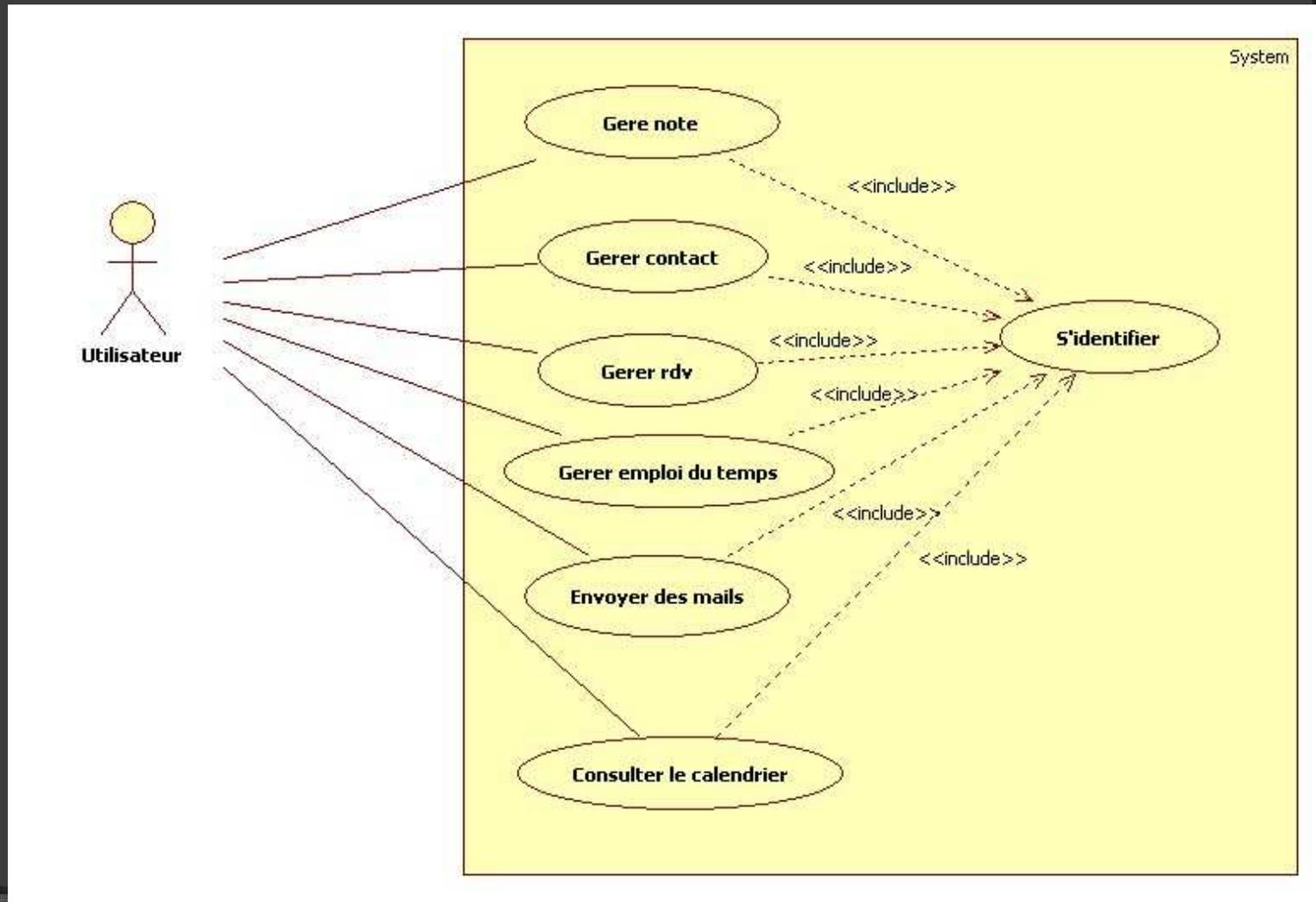
CONCEPTION

Model Logique de données



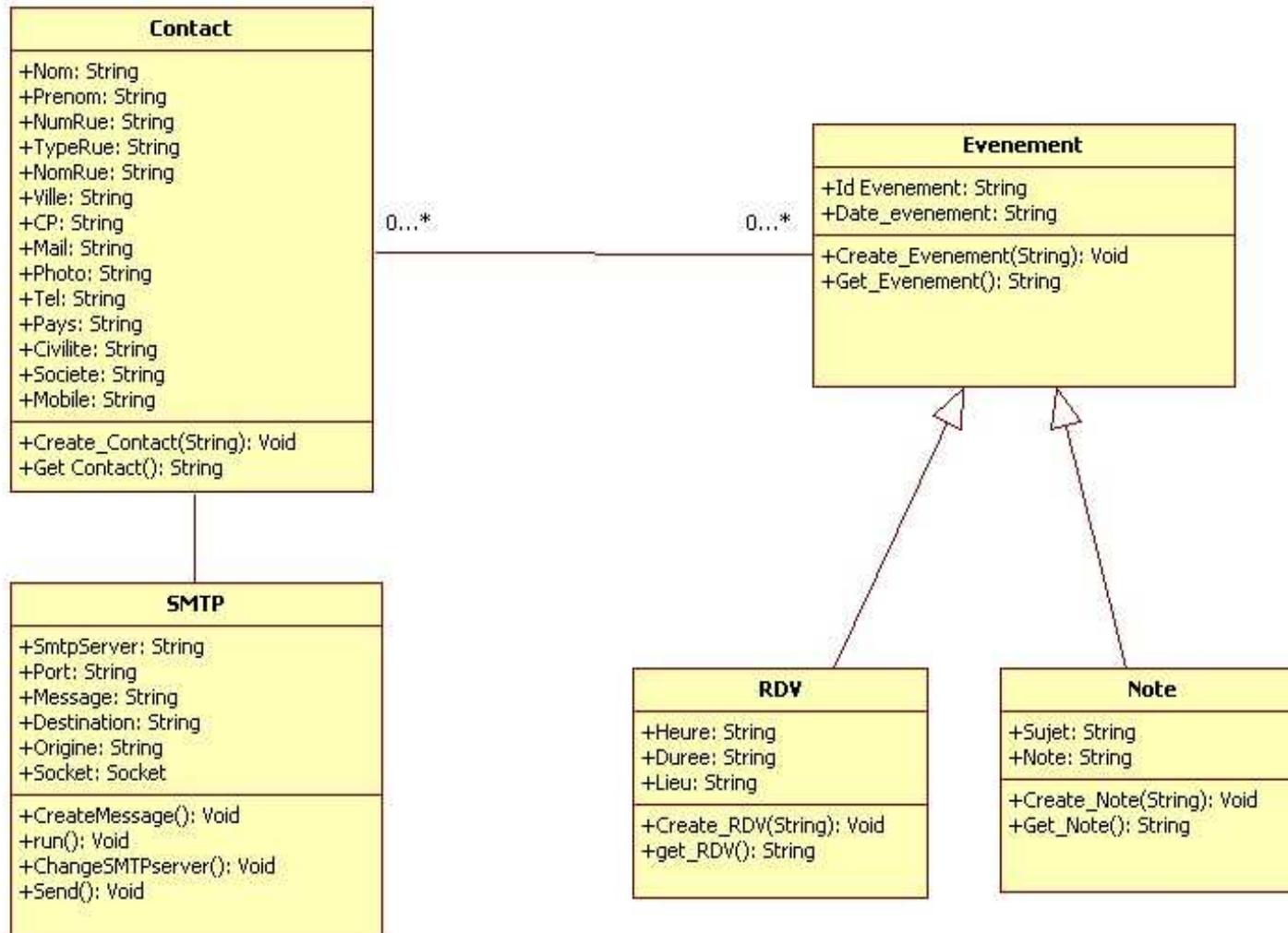
CONCEPTION

Diagramme des cas d'utilisation



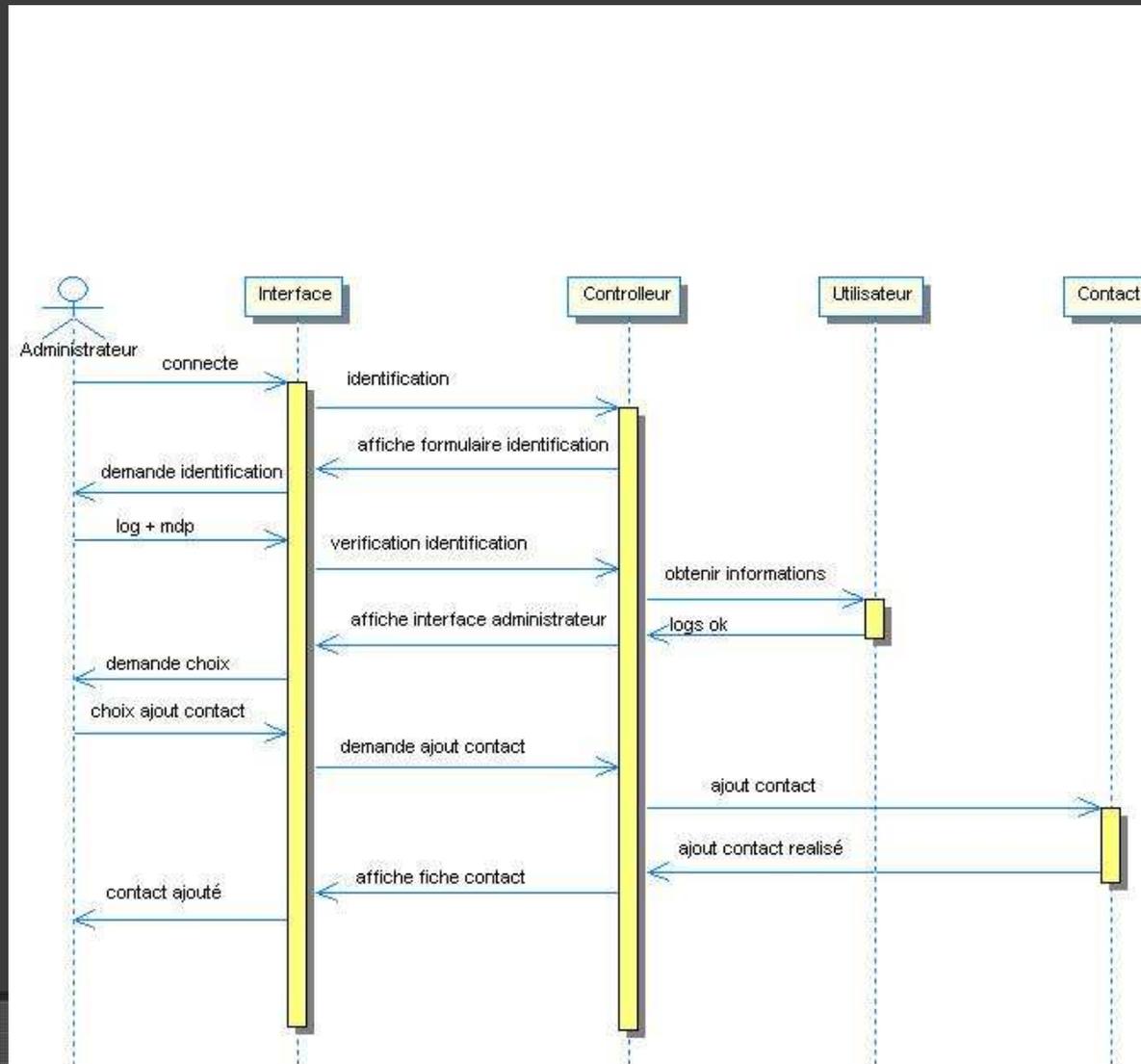
CONCEPTION

Diagramme de classe



CONCEPTION

Diagramme de séquence



REALISATION DU PROJET

Problèmes rencontrés:

- ⦿ Création base de données via Java
 - ⦿ Image de fond

Création d'une base de données via du code Java

```
public void Create_Database() throws SQLException, ClassNotFoundException, IOException
{
    if(Verify_DB())
    {
        Class.forName("com.mysql.jdbc.Driver");
        String url = "jdbc:mysql://localhost:3306/mysql";
        String user = "root";
        String pass = "";
        // on commence par se connecter à la base factice
        Connection factice = DriverManager.getConnection(url,user,pass);
        // on crée la base et on récupère une Connection
        Connection connection = createMySQLDatabase(factice,user,pass,this.Field_login.getText());
        // on peut finalement fermer notre Connection factice qui ne nous sers plus à rien
        factice.close();
        connection.close();
        // on ferme la bonne connection avec la nouvelle base de données
        connection.close();
        structure_database(this.Field_login.getText());
    }
}
```

```
public static Connection createMysqlDatabase(Connection factice, String user, String pass, String nomBase) throws SQLException
{
    Connection connection = null;
    Statement statement = null;
    try
    {
        statement = factice.createStatement();
        statement.execute("CREATE DATABASE "+nomBase);
        String url = factice.getMetaData().getURL();
        url = url.substring(0, url.lastIndexOf("/"));
        url += "/" + nomBase;
        connection = DriverManager.getConnection(url, user, pass);
    }
    catch(SQLException e)
    {
        SQLException sqle = new SQLException("Création de la base impossible");
        sqle.setNextException(e);
        throw sqle;
    }
    finally
    {
        try(statement.close(); catch(Exception e){})
    }
    return connection;
}
```

Image de fond

```
AuthenticationIHM fenetre_log = new AuthenticationIHM(); // frame d'authentification
/* Charge l'image en fonde de la JFrame */
IHMfond panel = new IHMFond(new ImageIcon("src\\stag\\icone\\fond.jpg").getImage());
fenetre_log.getContentPane().add(panel);
```

Image de fond

```
public IHMFond(Image img)
{
    this.img = img;
    Dimension size = new Dimension(img.getWidth(null), img.getHeight(null));
    setPreferredSize(size);
    setMinimumSize(size);
    setMaximumSize(size);
    setSize(size);
    setLayout(null);
}

@Override
public void paintComponent(Graphics g)
{
    g.drawImage(img, 0, 0, null);
}
```

Les différentes classes



IHM

- ContactIHM
- AuthentificationIHM
- ListeContactIHM
- CreationLoginIHM
- MenuIHM

ENTITY

- Contact
- SMTP

CONTROLEUR

- Controleur

LIFECYCLE

- Gestion_DB

Les exceptions

```
try
{
    /* Connection à la base de données STAG avec log : root et pwd : 090988 */
    connexion = DriverManager.getConnection("jdbc:mysql://localhost:3306/"+login, "root", dbpassword);
} catch (SQLException SQLException)
{
    /* affichage message d'erreur */
    JOptionPane.showMessageDialog(null, "Erreur sur la connexion à la base de donnée !");
}
```

Surcharge de méthode:

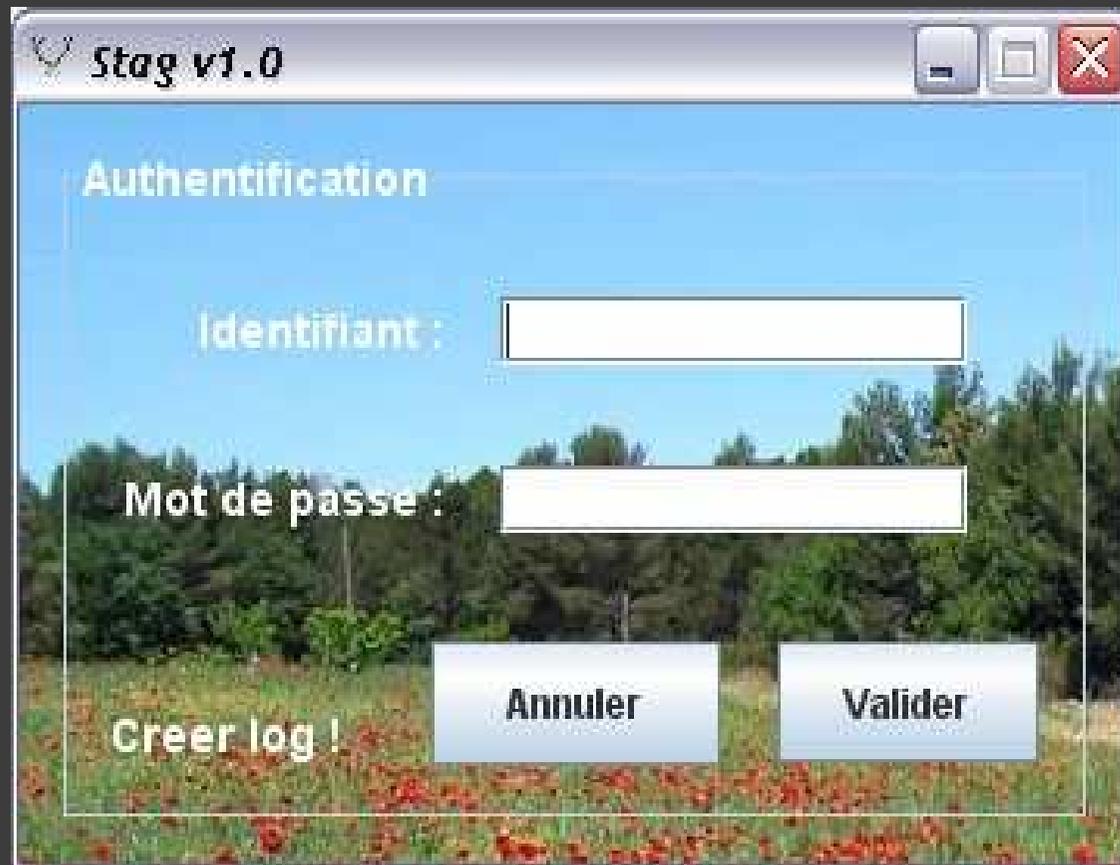
- Frame de type ContactIHM

```
/* fonction servant à mettre l'icone */
public static void Set_icone(String path, ContactIHM frame)
{
    File icone = new File(path);
    if(icone.exists())
    {
        ImageIcon icon = new ImageIcon(path);
        frame.setIconImage(icon.getImage());
    }
    else
    {
        /* affichage message d'erreur */
        JOptionPane.showMessageDialog(null, "Erreur de chargement sur l'icone de la frame !");
    }
}
```

- Frame de type CreationLogIHM

```
public static void Set_icone(String path, CreationLogIHM frame)
{
    File icone = new File(path);
    if(icone.exists())
    {
        ImageIcon icon = new ImageIcon(path);
        frame.setIconImage(icon.getImage());
    }
    else
    {
        /* affichage message d'erreur */
        JOptionPane.showMessageDialog(null, "Erreur de chargement sur l'icone de la frame !");
    }
}
}
```

Les différents menus et fonctionnalités: Interface d'authentification



Interface principale



Création d'un contact



Stag v1.0

Coordonnées

Identité

Nom :

Prenom :

Civilite : Homme Femme

Adresse

N°Rue : Type Rue :

Nom Rue :

Ville : Code Postal :

Pays :

Infos complémentaire

Mail :

Tel :

Mobile :

Societe :

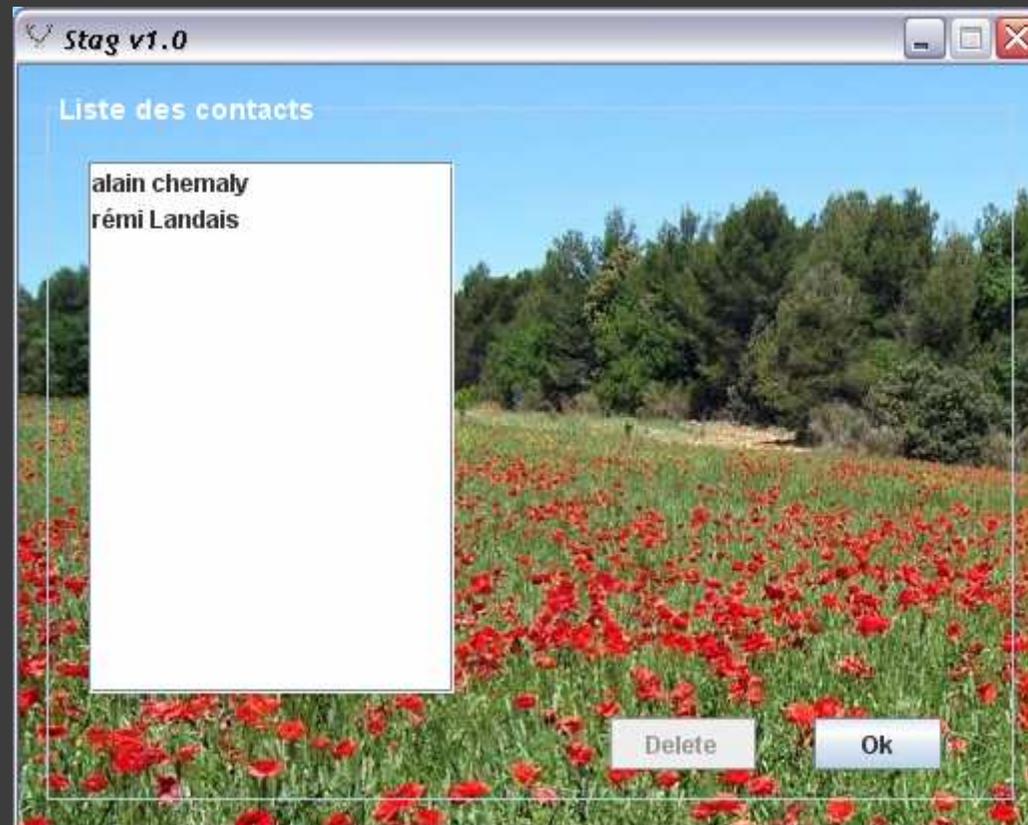
Photo

Photo

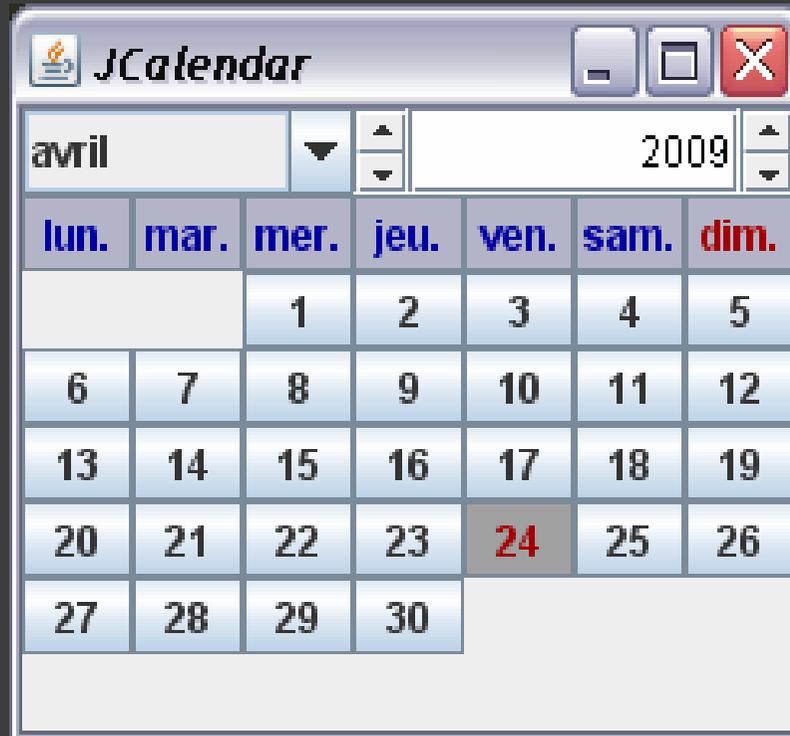
Annuler Enregistrer

A screenshot of a web application window titled "Stag v1.0". The window contains a contact creation form. The form is divided into several sections: "Identité" with fields for "Nom", "Prenom", and "Civilite" (radio buttons for "Homme" and "Femme"); "Adresse" with fields for "N°Rue", "Type Rue" (a dropdown menu showing "Rue"), "Nom Rue", "Ville", "Code Postal", and "Pays"; and "Infos complémentaire" with fields for "Mail", "Tel", "Mobile", and "Societe". There is also a "Photo" section with a large image placeholder and a "Photo" button. At the bottom right, there are two buttons: "Annuler" and "Enregistrer". The background of the form is a photograph of a field of red poppies under a blue sky.

Modification d'un contact



Calendrier



The screenshot shows a window titled "JCalendar" with standard window controls (minimize, maximize, close). Below the title bar, there is a dropdown menu for the month, currently set to "avril", and a text field for the year, set to "2009". The calendar grid below shows the days of the week: "lun.", "mar.", "mer.", "jeu.", "ven.", "sam.", and "dim.". The dates are arranged in a grid, with the 24th of April highlighted in red. The days of the week are color-coded: "lun." is blue, "mar." is blue, "mer." is blue, "jeu." is blue, "ven." is blue, "sam." is blue, and "dim." is red.

lun.	mar.	mer.	jeu.	ven.	sam.	dim.
		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			



Travail restant

Client SMTP

Module création de note et rendez vous

Gestion d'alertes pour les rendez vous

Merci pour votre attention

