

Aubrée Romain
08/12/2025

Beauvallet

Groupe 1

Bloc 1 TP 10

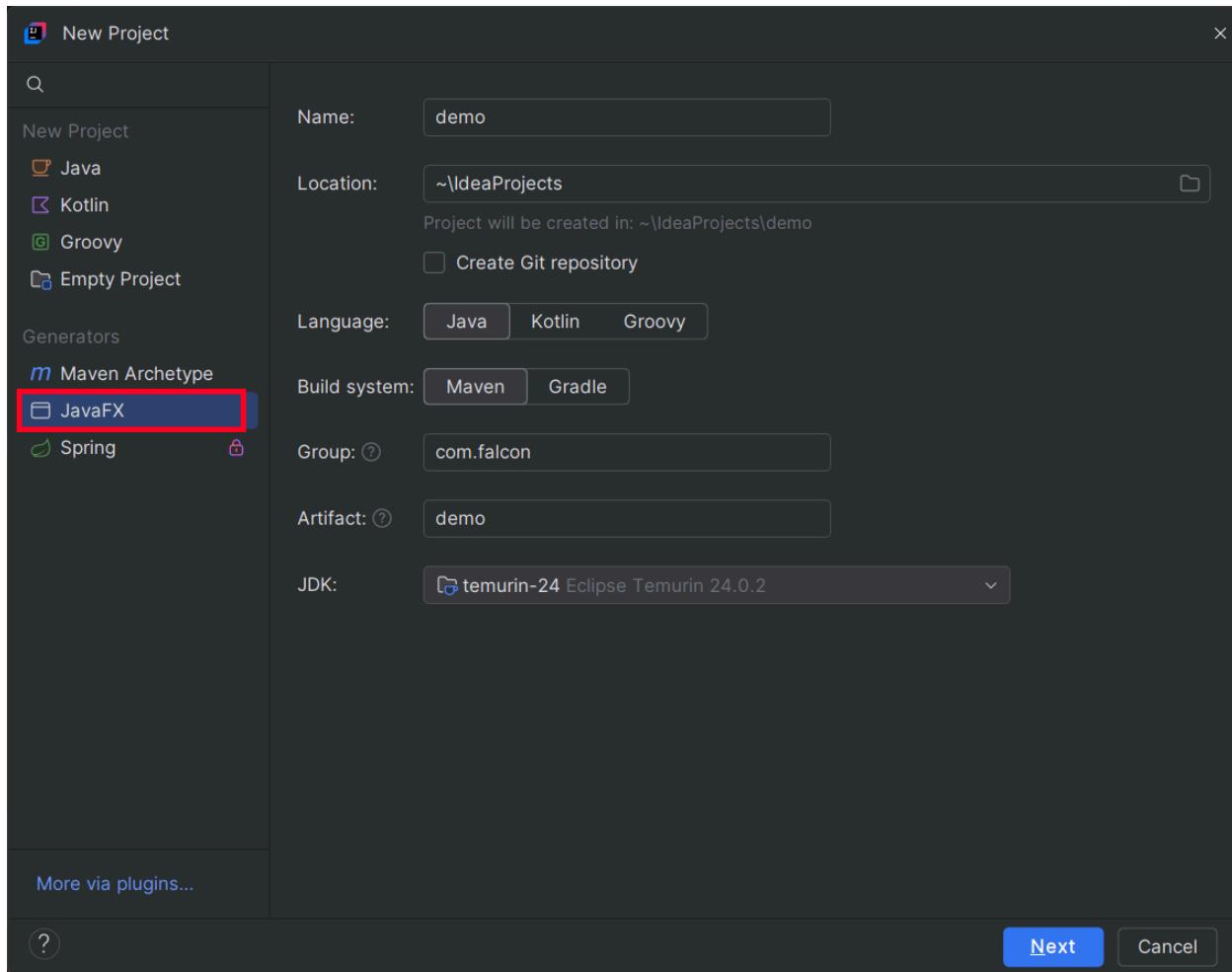
Création du projet	2
SceneBuilder	3
Code Java	5
Conclusion	7

Création du projet

Le but de ce TP était de faire une interface de connexion (Login) fonctionnelle avec **JavaFX** et **SceneBuilder**, incluant une vérification de mot de passe, des retours visuels utilisateur et une navigation vers un tableau de bord.

J'ai utilisé **IntelliJ IDEA Community**.

J'ai créé un projet sous **JavaFX**.

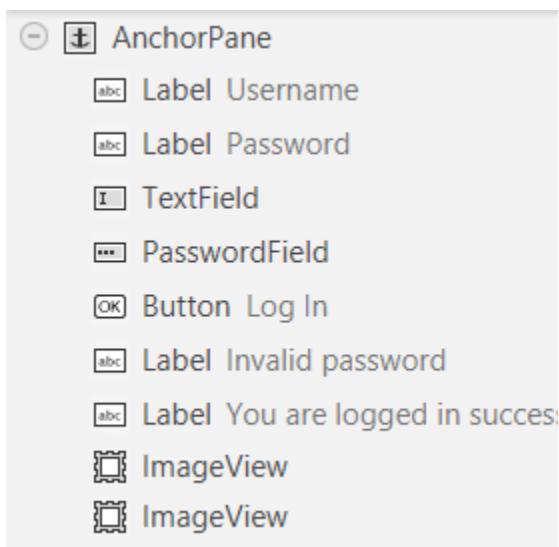


Ensuite, j'ai installé le logiciel **SceneBuilder** qui permet de fabriquer les "fenêtres" avec une interface graphiques

SceneBuilder

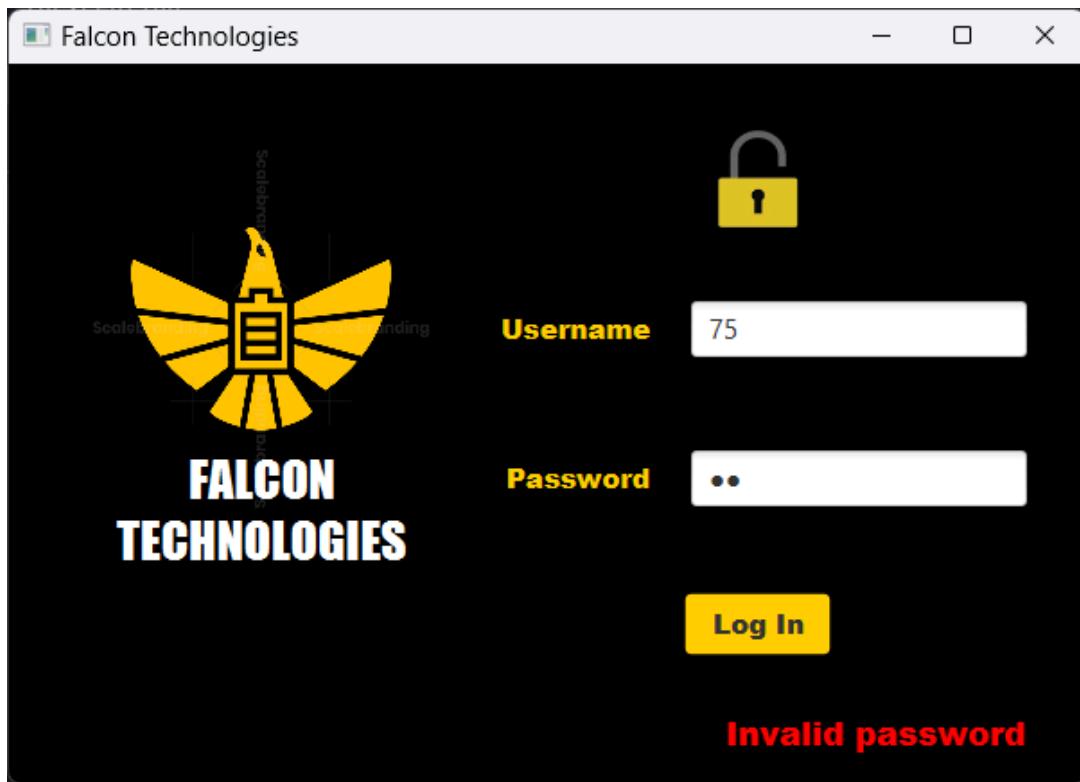
Dans le projet IntelliJ, j'ai ouvert “hello-view.fxml” avec **SceneBuilder** et j'ai supprimé les composants déjà présents par défaut.

J'ai ajouter les éléments :

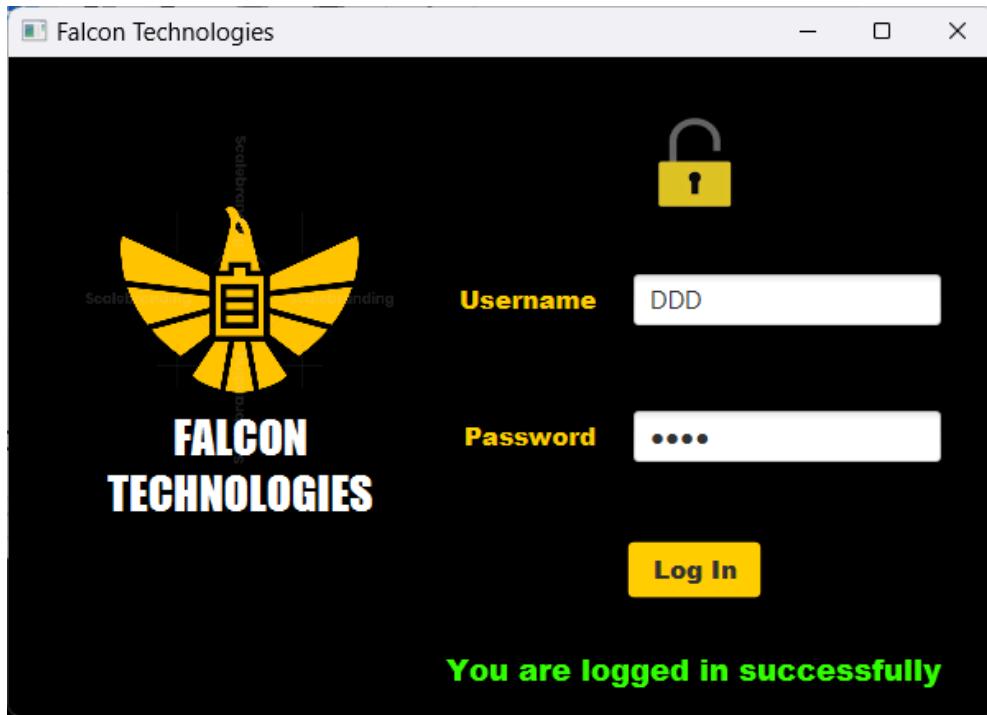


Les labels “*Invalid password*” et “*You are logged in successfully*” sont par défaut pas visible, le code dans IntelliJ permet de l'afficher si l'utilisateur rentre un mauvais mot de passe.

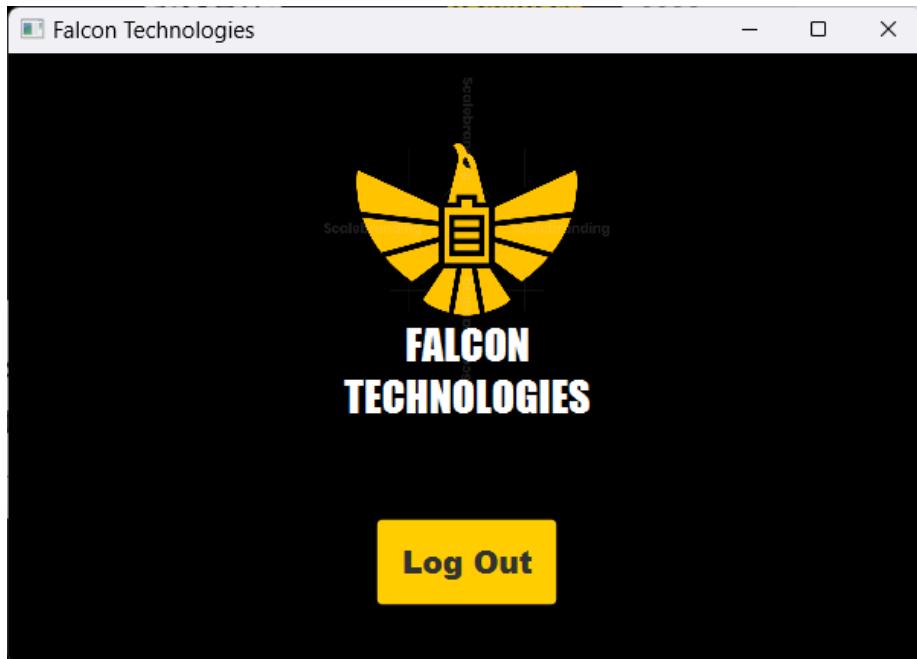
Mot de passe invalide :



Mot de passe valide :



Ensuite, j'ai créé une 2éme fenêtre qui s'appelle “*dashboard.fxml*” qui affichera cette fenêtre (elle redirige à l'ancienne fenêtre si on clique sur “*Log Out*”).



Code Java

Cette partie va ouvrir la fenêtre, définir sa taille, lui appliquer le nom (qui sera en haut à gauche).

```
package com.falcon.falconlogin;

import javafx.application.Application;
import javafx.fxml.FXMLLoader;
import javafx.scene.Scene;
import javafx.stage.Stage;

import java.io.IOException;

public class HelloApplication extends Application {
    @Override
    public void start(Stage stage) throws IOException {
        FXMLLoader fxmlLoader = new FXMLLoader(HelloApplication.class.getResource("hello-view.fxml"));
        Scene scene = new Scene(fxmlLoader.load(), 600, 400);
        stage.setTitle("Falcon Technologies");
        stage.setScene(scene);
        stage.show();
    }
}
```

```
package com.falcon.falconlogin;

import javafx.application.Application;

public class Launcher {
    public static void main(String[] args) {
        Application.launch(HelloApplication.class, args);
    }
}
```

Cette partie va checker si la partie “*password*” contient “1234”, elle rendra visible le texte qui dira que le mot de passe est correct, sinon elle rendra visible l’autre texte qui dit qu’il n’est pas bon, j’ai rajouté un délai de 2 secondes quand le mot de passe est correct puis, elle ouvre l’autre fenêtre “*dashboard.fxml*”, si l’utilisateur clique sur “*Log Out*”, Cela va ouvrir l’ancienne fenêtre.

```
package com.falcon.falconlogin;

import javafx.animation.PauseTransition;
import javafx.event.ActionEvent;
import javafx.fxml.FXML;
import javafx.fxml.FXMLLoader;
import javafx.scene.Node;
import javafx.scene.Parent;
import javafx.scene.Scene;
import javafx.scene.control.Button;
import javafx.scene.control.Label;
import javafx.scene.control.PasswordField;
import javafx.scene.control.TextField;
import javafx.stage.Stage;
import javafx.util.Duration;

import java.io.IOException;

public class HelloController {

    @FXML
    private TextField username;
    @FXML
    private PasswordField password;
    @FXML
    private Label successLabel;
    @FXML
    private Label warningLabel;
    @FXML
    private Button loginButton;
    @FXML
    private Button disconnectButton;

    @FXML
    protected void handleLogin(ActionEvent event) {
        String pass = password.getText();

        if (pass.equals("1234")) {
            warningLabel.setVisible(false);
            successLabel.setVisible(true);

            PauseTransition pause = new PauseTransition(Duration.seconds(2));
            pause.setOnFinished( ActionEvent e -> {
                try {
                    switchScene(event, fxmleFile: "dashboard.fxml");
                } catch (IOException ex) {
                    ex.printStackTrace();
                }
            });
            pause.play();
        } else {
            successLabel.setVisible(false);
            warningLabel.setVisible(true);
        }
    }
}
```

```

@FXML
protected void handleLogout(ActionEvent event) throws IOException {
    switchScene(event, fxmlFile: "hello-view.fxml");
}

private void switchScene(ActionEvent event, String fxmlFile) throws IOException { 2 usages
    FXMLLoader loader = new FXMLLoader(getClass().getResource(fxmlFile));
    Parent root = loader.load();
    Stage stage = (Stage) ((Node) event.getSource()).getScene().getWindow();
    Scene scene = new Scene(root);
    stage.setScene(scene);
    stage.show();
}
}

@FXML
protected void handleLogout(ActionEvent event) throws IOException {
    switchScene(event, fxmlFile: "hello-view.fxml");
}

private void switchScene(ActionEvent event, String fxmlFile) throws IOException { 2 usages
    FXMLLoader loader = new FXMLLoader(getClass().getResource(fxmlFile));
    Parent root = loader.load();
    Stage stage = (Stage) ((Node) event.getSource()).getScene().getWindow();
    Scene scene = new Scene(root);
    stage.setScene(scene);
    stage.show();
}
}

```

Conclusion

Introduction à la création de fenêtre avec “SceneBuilder” pour créer des interfaces personnalisées.