Mohamed chili 212131053038 L3

Exercice 1 :

```java
package Tp5;

public class validationmotdepass {

    private static final int MIN_LENGTH = 8;
    private static final int MAX_LENGTH = 14;

    public boolean valide(String password) {
        return validateLength(password) && validateCharacters(password);
    }

    private boolean validateLength(String password) {
        return password.length() >= MIN_LENGTH && password.length() <=
MAX_LENGTH;
    }

    private boolean validateCharacters(String password) {
        return hasDigit(password) && hasLowerCase(password) &&
hasUpperCase(password) && hasSpecialCharacter(password);
    }

    private boolean hasDigit(String password) {
        return password.matches(".*[0-9].*");
    }

    private boolean hasLowerCase(String password) {
        return password.matches(".*[a-z].*");
    }

    private boolean hasUpperCase(String password) {
        return password.matches(".*[A-Z].*");
    }

    private boolean hasSpecialCharacter(String password) {
        return password.matches(".*[^a-zA-Z0-9].*");
    }
}
```

```java
package Tp5;

import org.junit.jupiter.api.Test;

import static org.junit.jupiter.api.Assertions.*;
```

```java
public class PasswordValidatorTest {

    private validationmotdepass validateur = new PasswordValidator();

    @Test
    public void testerMotDePasseInvalides() {
        assertAll(
                () -> assertFalse(validateur.valide("koko"), "Manque de
chiffres et de caractères spéciaux"),
                () -> assertFalse(validateur.valide("12345678"), "Manque de
lettres"),
                () -> assertFalse(validateur.valide("lala"), "Manque de
chiffres ou de caractères spéciaux"),
                () -> assertFalse(validateur.valide("hoos4"), "Manque de
caractère spécial")
        );
    }

    @Test
    public void testerMotDePasseValide() {
        assertTrue(validateur.valide("nvas2@_"), "Mot de passe valide avec
tous les types de caractères");
    }
}
```