

**LAPORAN PRATIKUM  
ALGORITMA DAN PEMROGRAMAN  
“Laporan Tugas Pekan 7 ”**

Disusun Oleh:  
Zaki Misucha Andefi  
2511533006  
Informatika Kelas B  
Dosen Pengampu : Dr. Wahyudi, S.T, M.T.  
Asisten Praktikum : Rahmad Dwirizki Olders



DEPARTEMEN INFORMATIKA  
FAKULTAS TEKNOLOGI INFORMASI  
UNIVERSITAS ANDALAS  
2025

## Soal

---

### Intruksi:

1. Buatlah program Java untuk studi kasus tunggal yang dijelaskan di bawah ini.
  2. Program wajib dibuat menggunakan Scanner untuk input data dari pengguna.
  3. Program wajib mengimplementasikan konsep:
    - Class & Object (minimal 2 class: 1 class data, 1 class utama/driver).
    - Method (termasuk *setter/getter* jika dibutuhkan, dan *helper method* untuk validasi).
    - Logika Percabangan (if-else atau switch-case).
    - Operasi dan manipulasi String (misalnya: .length(), .toUpperCase(), .toLowerCase(), .indexOf(), .concat(), + untuk string, dan *escape character \*).
  4. Wajib menuliskan:
    - Pseudocode program
    - Flowchart program (ditulis tangan, difoto/scan, masukkan ke laporan)
    - Kode program Java (file .java)
    - Screenshot output program (1 kasus gagal validasi dan 1 kasus berhasil).
  5. Simpan file Java dengan nama: tugasAlproPekan7\_NIM.java (untuk class utama) dan Akun.java (untuk class data).
  6. Buat laporan lengkap dalam format Word (.docx), kemudian **konversi ke PDF** dan unggah ke *repository* GitHub.
  7. Logika validasi (aturan password/email) harus dijelaskan di laporan (ditulis di bagian pseudocode atau penjelasan singkat program).
-

## 1. Pseudocode

### PROGRAM AkunJava\_2511533006

{Program untuk registrasi akun baru dengan validasi data}

#### Deklarasi

```
scanner : Scanner  
username, password, email : string  
pinAngka : integer  
validPin : boolean  
akun : Object Akun
```

#### Pseudocode

BUAT scanner = new Scanner(System.in)

TAMPILKAN "--- REGISTRASI AKUN BARU ---"

TAMPILKAN "Masukkan Username: "

INPUT username = scanner.nextLine()

TAMPILKAN "Masukkan Password: "

INPUT password = scanner.nextLine()

TAMPILKAN "Masukkan Email: "

INPUT email = scanner.nextLine()

SET pinAngka = 0

SET validPin = false

WHILE validPin == false

TRY

TAMPILKAN "Masukkan PIN (6 digit): "

INPUT pinAngka = scanner.nextInt()

```
scanner.nextLine() // clear buffer

IF String.valueOf(pinAngka).length() == 6
    SET validPin = true
ELSE
    TAMPILKAN "Error: PIN harus 6 digit! Silakan coba lagi."
    TAMPILKAN ""
ENDIF

CATCH InputMismatchException
    TAMPILKAN "Error: PIN harus berupa angka! Silakan coba lagi."
    TAMPILKAN ""
    scanner.nextLine() // clear invalid input
END TRY

END WHILE

BUAT akun = new Akun(username, password, email, pinAngka)

IF akun.isPasswordValid() AND akun.isEmailValid()
    TAMPILKAN "--- REGISTRASI BERHASIL ---"
    TAMPILKAN "Akun untuk '" + username + "' telah berhasil dibuat."

    TAMPILKAN "--- Detail Akun ---"
    TAMPILKAN "Username (Lowercase): " + username.toLowerCase()
    TAMPILKAN "Email (Uppercase): " + email.toUpperCase()
    TAMPILKAN "ID Pengguna (Gabungan): " + username + pinAngka

    TAMPILKAN "--- Uji Tipe Data (PIN Anda: " + pinAngka + ") ---"
    TAMPILKAN "PIN (int) + 10 = " + (pinAngka + 10)
    TAMPILKAN "PIN (String) + \"10\" = " + pinAngka + "10"

ELSE
    TAMPILKAN "--- REGISTRASI GAGAL ---"
```

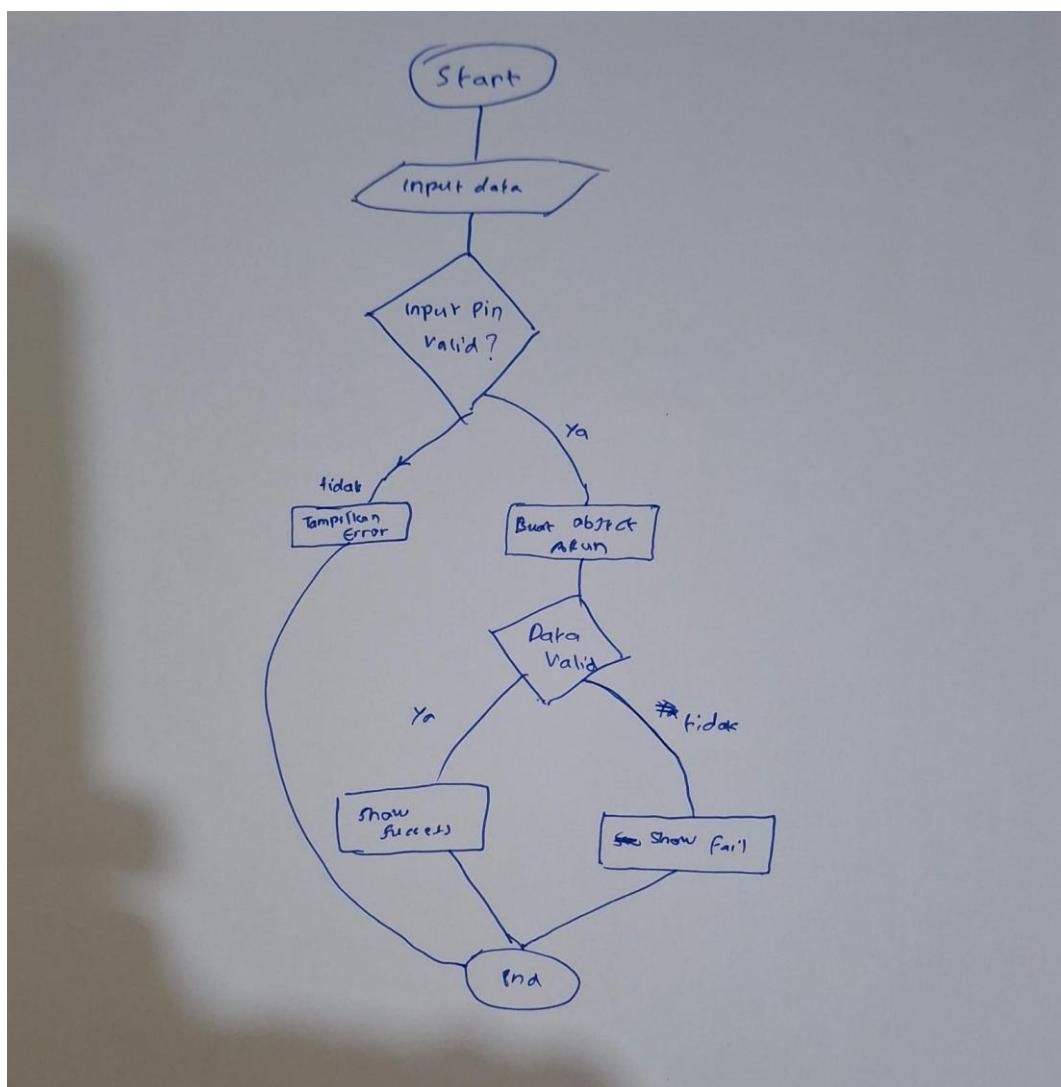
```
IF NOT akun.isPasswordValid()
    TAMPILKAN "Password Anda tidak valid (minimal 8 karakter)."
ENDIF

IF NOT akun.isEmailValid()
    TAMPILKAN "Email Anda \" + email + "\"" tidak valid (harus
mengandung '@' dan '.')."
ENDIF

TAMPILKAN "Silakan coba lagi."
ENDIF

scanner.close()
END PROGRAM
```

## 2. Flowchart



### 3. Kode Program

```
package pekan7_2511533006;

import java.util.Scanner;

import java.util.InputMismatchException;

class Akun {

    private String username;
    private String password;
    private String email;
    private int pinAngka;

    public Akun(String username, String password, String email, int pinAngka) {
        this.username = username;
        this.password = password;
        this.email = email;
        this.pinAngka = pinAngka;
    }

    public String getUsername() { return username; }
    public String getPassword() { return password; }
    public String getEmail() { return email; }
    public int getPinAngka() { return pinAngka; }

    public boolean isPasswordValid() {
        return password != null && password.length() >= 8;
    }

    public boolean isEmailValid() {
        return email != null && email.contains("@") && email.contains(".");
    }
}
```

```
public class AkunJava_2511533006 {  
  
    public static void main(String[] args) {  
        Scanner scanner = new Scanner(System.in);  
  
        System.out.println("--- REGISTRASI AKUN BARU ---");  
  
        // Input data Pengguna Akun  
        System.out.print("Masukkan Username: ");  
        String username = scanner.nextLine();  
  
        System.out.print("Masukkan Password: ");  
        String password = scanner.nextLine();  
  
        System.out.print("Masukkan Email: ");  
        String email = scanner.nextLine();  
  
        // Input PIN dengan output Error  
        int pinAngka = 0;  
        boolean validPin = false;  
  
        while (!validPin) {  
            try {  
                System.out.print("Masukkan PIN (6 digit): ");  
                pinAngka = scanner.nextInt();  
                scanner.nextLine(); // membersihkan newline character  
  
                // Validasi panjang PIN  
                if (String.valueOf(pinAngka).length() == 6) {  
                    validPin = true;  
                } else {  
                    System.out.println("Error: PIN harus 6 digit! Silakan coba lagi.\n");  
                }  
            } catch (Exception e) {  
                System.out.println("Input tidak valid. Silakan masukkan angka 6 digit.\n");  
            }  
        }  
    }  
}
```

```
        }

    } catch (InputMismatchException e) {
        System.out.println("Error: PIN harus berupa angka! Silakan coba
lagi.\n");
        scanner.nextLine(); // membersihkan input yang invalid
    }
}

Akun akun = new Akun(username, password, email, pinAngka);

if (akun.isPasswordValid() && akun.isEmailValid()) {
    System.out.println("\n--- REGISTRASI BERHASIL ---");
    System.out.println("Akun untuk '" + akun.getUsername() + "' telah
berhasil dibuat.");

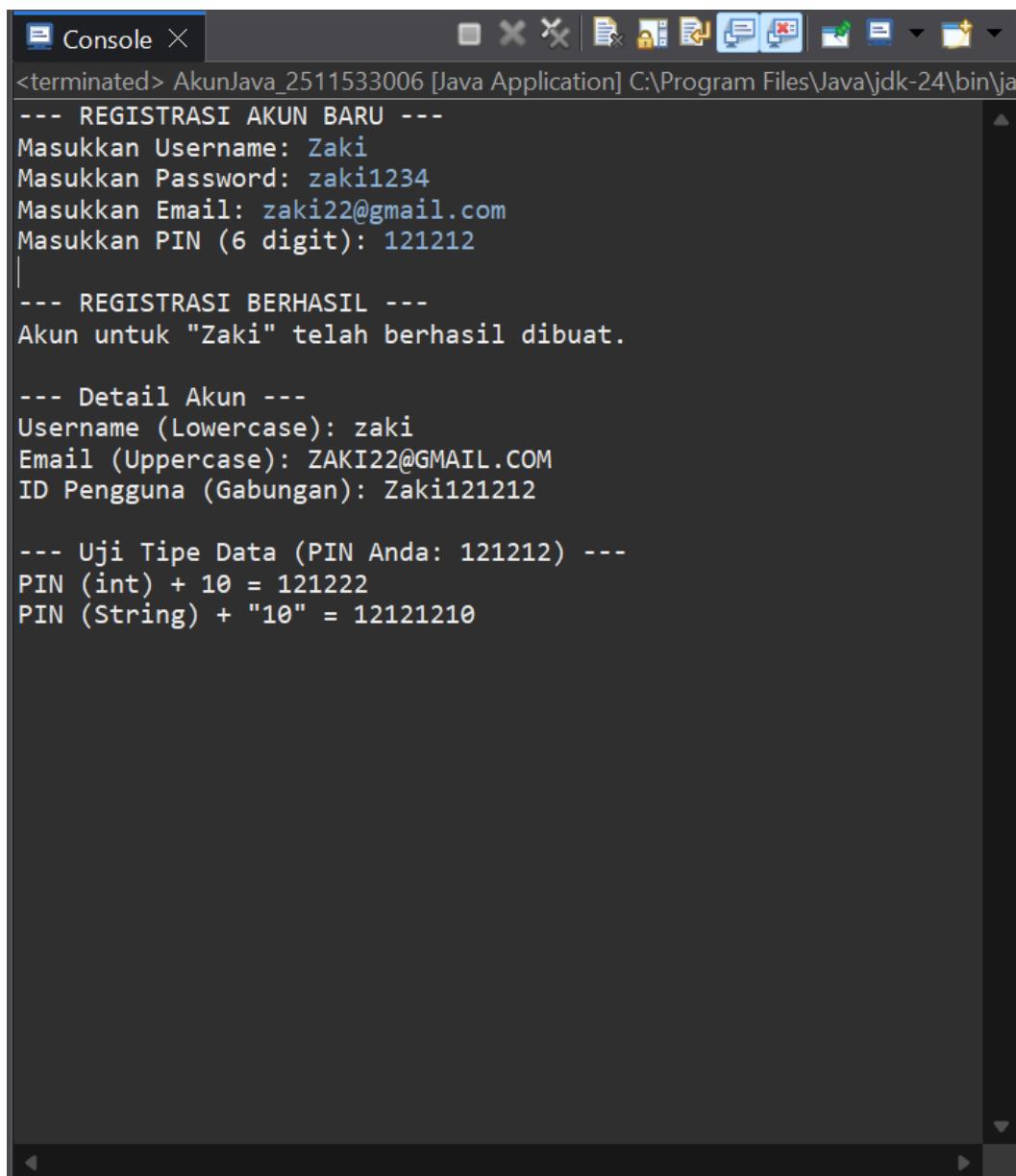
    System.out.println("\n--- Detail Akun ---");
    System.out.println("Username (Lowercase): " +
akun.getUsername().toLowerCase());
    System.out.println("Email (Uppercase): " +
akun.getEmail().toUpperCase());
    System.out.println("ID Pengguna (Gabungan): " + akun.getUsername() +
akun.getPinAngka());

    System.out.println("\n--- Uji Tipe Data (PIN Anda: " +
akun.getPinAngka() + ") ---");
    System.out.println("PIN (int) + 10 = " + (akun.getPinAngka() + 10));
    System.out.println("PIN (String) + \"10\" = " + akun.getPinAngka() +
"10");
} else {
    // Registrasi gagal
    System.out.println("\n--- REGISTRASI GAGAL ---");
}
```

```
if (!akun.isPasswordValid()) {  
    System.out.println("Password Anda tidak valid (minimal 8 karakter).");  
}  
  
if (!akun.isEmailValid()) {  
    System.out.println("Email Anda \" + akun.getEmail() + "\" tidak valid  
(harus mengandung '@' dan '!').");  
}  
  
System.out.println("Silakan coba lagi.");  
}  
  
scanner.close();  
}  
}
```

#### 4. Hasil Output Program

##### 1. Program dengan Output Berhasil



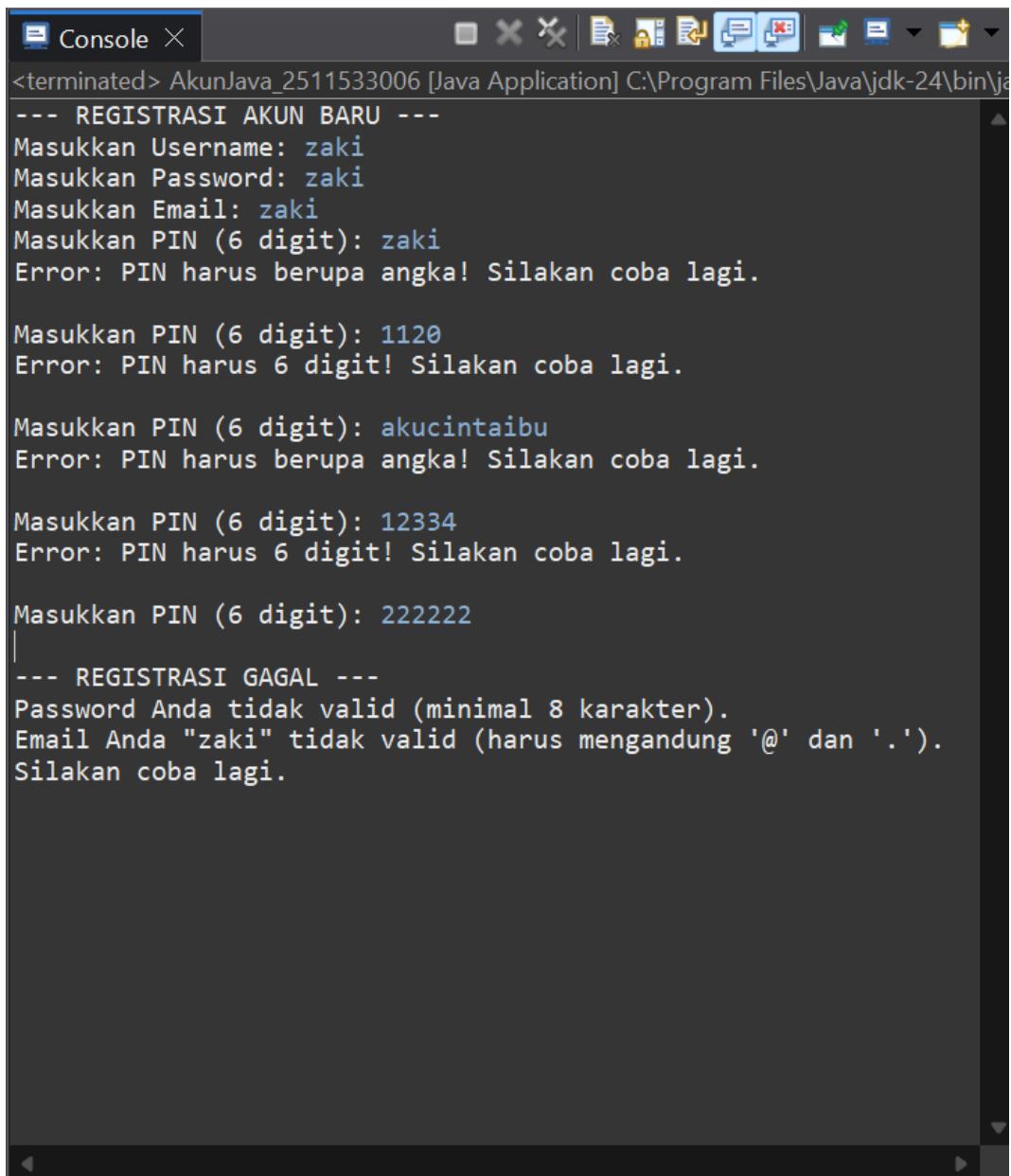
The screenshot shows a Java application window titled "Console". The application has registered a new account named "Zaki". The output text is as follows:

```
<terminated> AkunJava_2511533006 [Java Application] C:\Program Files\Java\jdk-24\bin\java
--- REGISTRASI AKUN BARU ---
Masukkan Username: Zaki
Masukkan Password: zaki1234
Masukkan Email: zaki22@gmail.com
Masukkan PIN (6 digit): 121212
|
--- REGISTRASI BERHASIL ---
Akun untuk "Zaki" telah berhasil dibuat.

--- Detail Akun ---
Username (Lowercase): zaki
Email (Uppercase): ZAKI22@GMAIL.COM
ID Pengguna (Gabungan): Zaki121212

--- Uji Tipe Data (PIN Anda: 121212) ---
PIN (int) + 10 = 121222
PIN (String) + "10" = 12121210
```

## 2. Program dengan Output Gagal



The screenshot shows a Java application window titled "Console". The title bar also displays the application name "AkunJava\_2511533006 [Java Application]" and the path "C:\Program Files\Java\jdk-24\bin\ja". The main area of the window is a terminal-like console where a registration process is being run. The console output is as follows:

```
<terminated> AkunJava_2511533006 [Java Application] C:\Program Files\Java\jdk-24\bin\ja
--- REGISTRASI AKUN BARU ---
Masukkan Username: zaki
Masukkan Password: zaki
Masukkan Email: zaki
Masukkan PIN (6 digit): zaki
Error: PIN harus berupa angka! Silakan coba lagi.

Masukkan PIN (6 digit): 1120
Error: PIN harus 6 digit! Silakan coba lagi.

Masukkan PIN (6 digit): akucintaibu
Error: PIN harus berupa angka! Silakan coba lagi.

Masukkan PIN (6 digit): 12334
Error: PIN harus 6 digit! Silakan coba lagi.

Masukkan PIN (6 digit): 222222
|
--- REGISTRASI GAGAL ---
Password Anda tidak valid (minimal 8 karakter).
Email Anda "zaki" tidak valid (harus mengandung '@' dan '.').
Silakan coba lagi.
```

## 5. Penjelasan Singkat Program

### Deskripsi Program

Program ini merupakan sistem registrasi akun baru yang memungkinkan pengguna untuk mendaftarkan diri dengan memasukkan username, password, email, dan PIN. Program melakukan validasi terhadap data yang dimasukkan sebelum akun dapat berhasil dibuat.

### Struktur Program

Program terdiri dari 2 class utama:

1. **Class Akun** - Sebagai blueprint untuk menyimpan data akun dan melakukan validasi
2. **Class AkunJava\_2511533006** - Sebagai class utama yang menangani input/output dan logika program

### Logika Validasi yang Diimplementasikan

1. Validasi Password

```
public boolean isPasswordValid() {  
    return password != null && password.length() >= 8;  
}
```

#### Kriteria Validasi:

- Password tidak boleh null (kosong)
- Panjang password minimal 8 karakter
- Menggunakan method length() dari String untuk menghitung jumlah karakter

2. Validasi Email

```
public boolean isValidEmail() {  
    return email != null && email.contains("@") && email.contains(".");  
}
```

#### Kriteria Validasi:

- Email tidak boleh null (kosong)
- Email harus mengandung karakter @
- Email harus mengandung karakter .

- Menggunakan method contains() dari String untuk pengecekan pattern

### 3. Validasi PIN

```
if (String.valueOf(pinAngka).length() == 6) {
    validPin = true;
```

#### Kriteria Validasi:

- PIN harus berupa angka (ditangani dengan try-catch)
- PIN harus tepat 6 digit
- Menggunakan konversi int ke String untuk menghitung digit

#### Alur Program

1. **Input Data** - Pengguna memasukkan username, password, email, dan PIN
2. **Validasi PIN** - Program memastikan PIN berupa angka 6 digit dengan loop dan error handling
3. **Validasi Akun** - Program memvalidasi password dan email menggunakan helper method
4. **Output Hasil** - Menampilkan hasil registrasi (berhasil/gagal) beserta detail akun

#### Fitur Keamanan

- **Null Safety Check** - Mencegah NullPointerException dengan pengecekan null
- **Error Handling** - Menangani input tidak valid dengan try-catch
- **Input Validation Loop** - Memastikan input PIN valid sebelum melanjutkan

#### Manipulasi String yang Digunakan

1. **.toLowerCase()** - Mengubah username menjadi huruf kecil
2. **.toUpperCase()** - Mengubah email menjadi huruf besar
3. **.length()** - Menghitung panjang password
4. **.contains()** - Memeriksa pattern dalam email
5. **Concatenation** - Menggabungkan string untuk ID Pengguna
6. **Escape Character** - Menggunakan \" untuk menampilkan tanda kutip

Program ini memastikan data yang dimasukkan pengguna memenuhi standar keamanan dasar sebelum akun dapat dibuat, sehingga mengurangi kemungkinan data tidak valid tersimpan dalam sistem.