# Laporan Tugas Pertemuan ke-11

## Identitas

* NPM : 223040031
* Nama : Novan Ramdan
* Kelas : A
* URL Repository Github : <https://github.com/NovanR/pp2_223040031_A/tree/main/Pertemuan%2011/jfc-mvc>

## Penjelasan Kode

|  |
| --- |
| **User.java** |
|  |
| **Penjelasan** |
| Kode di atas adalah sebuah *class* Java bernama User dengan tiga atribut privat (id, name, dan email). Kode ini juga mencakup *getter* dan *setter* untuk setiap atribut. |

|  |
| --- |
| **UserMapper.java** |
|  |
| **Penjelasan** |
| 1. **@Select** digunakan untuk mengambil semua data dari tabel users dan mengembalikannya sebagai list objek User. 2. **@Insert** digunakan untuk menyisipkan data ke tabel users dengan parameter name dan email. |

|  |
| --- |
| **MyBatisUtil.java** |
|  |
| **Penjelasan** |
| Kode di atas adalah **Utility Class** yang digunakan untuk mendapatkan objek SqlSession dalam **MyBatis**:   1. **SqlSessionFactory**: Bertanggung jawab membuat sesi database (SqlSession). 2. **getSqlSession():**   Metode ini mengembalikan sesi SQL yang baru (openSession). |

|  |
| --- |
| **UserView.java** |
|  |
| **Penjelasan** |
| * Turunan dari JFrame untuk membuat GUI "User Management". * Menggunakan **Swing** untuk elemen-elemen GUI seperti JTextField, JButton, dan JList. * **userList**: Daftar pengguna yang ditampilkan menggunakan JList. * getNameInput() dan getEmailInput(): Mengambil teks dari input pengguna. * setUserList(): Menyegarkan data pada daftar pengguna. * addAddUserListener() dan addRefreshListener(): Menambahkan event listener ke tombol. |

|  |
| --- |
| **UserController.java** |
|  |
| **Penjelasan** |
| * Menghubungkan UserView (tampilan) dengan UserMapper (database melalui MyBatis). * Ketika tombol **Add** ditekan, input **name** dan **email** diambil. * Jika input tidak kosong, data akan dimasukkan ke database menggunakan mapper.insertUser(user). * Notifikasi sukses atau kesalahan ditampilkan dengan JOptionPane. * Ketika tombol **Refresh** ditekan, semua data pengguna diambil dari database menggunakan mapper.getAllUsers(). * Data diubah ke format String[] dan dikirim ke view.setUserList() untuk menampilkan daftar pengguna. |

## 

|  |
| --- |
| **Main.java** |
|  |
| **Penjelasan** |
| * Menggunakan **MyBatisUtil.getSqlSession()** untuk membuat sesi SQL. * Membuat **UserView** sebagai tampilan GUI. * Menghubungkan **UserView** dan **UserMapper** dengan **UserController**. * view.setVisible(true); memastikan tampilan GUI terlihat di layar. |
| **Output** |
|  |