

Tugas Praktikum VIII

Tugas Praktikum VII

(Kelompok 8)

- M. Naufal Muafa (1301180091)
- Ingrid Resmi Benita (1301184107)
- Saskia Putri Ananda (1301184157)
- Anis Novitasari (1301184424)

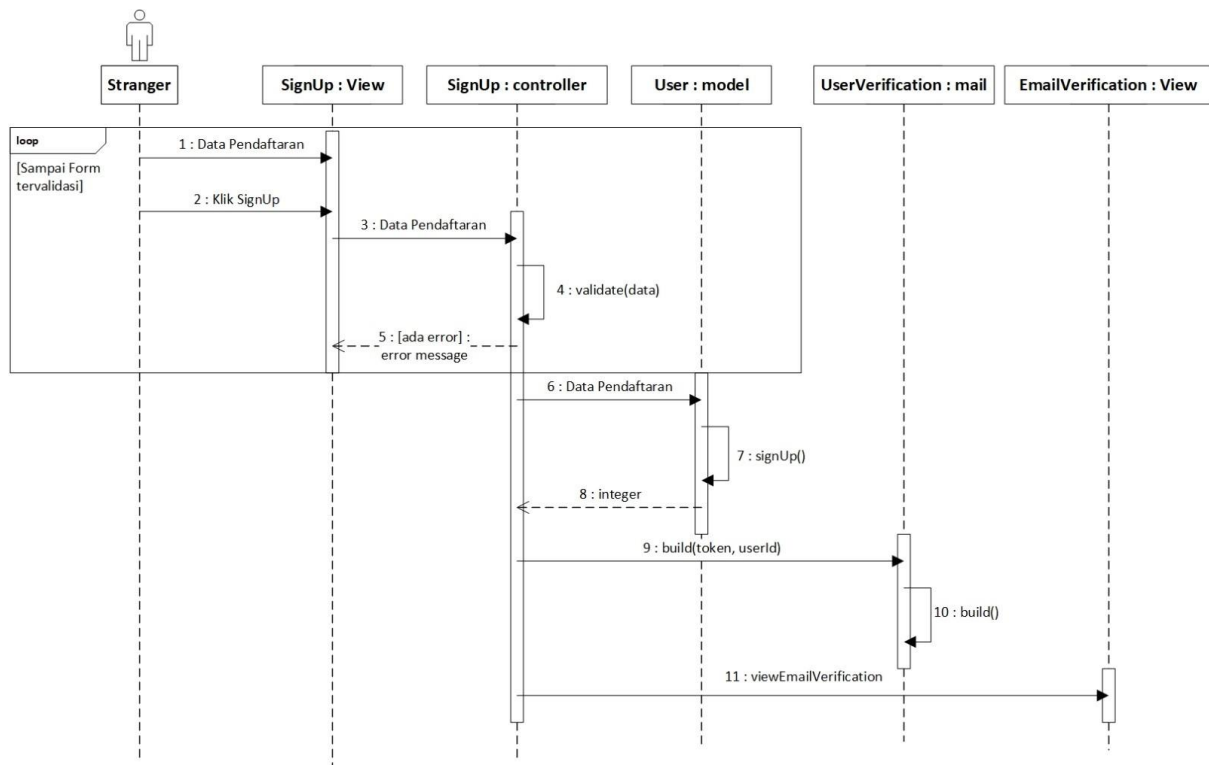
Pada file ini terdapat jawaban Tugas Pendahuluan

TUGAS PENDAHULUAN - Membuat *SEQUENCE DIAGRAM*

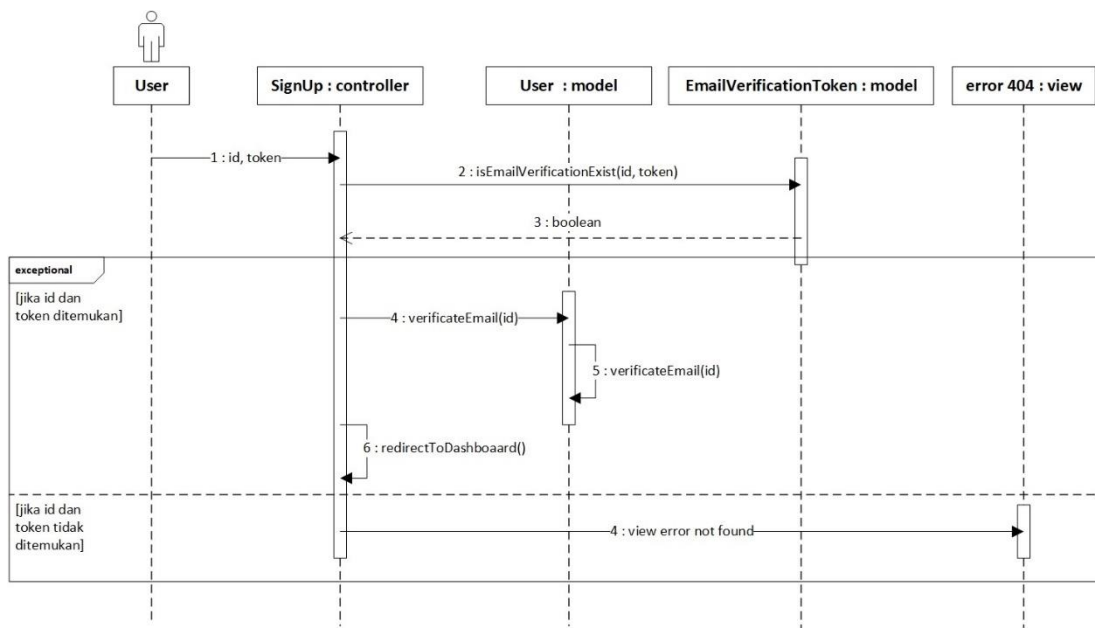
1. Periksa kembali apakah di DPPL Kelompok Anda terdapat *Sequence Diagram* untuk setiap *USE CASE* yang ada di *USE CASE DIAGRAM*. (Misal ada 5 *use case*, maka perlu dibuat 5 *Sequence Diagram*).
2. Jika belum ada, buatlah *SEQUENCE DIAGRAM* di DPPL Anda.
3. Jika sudah ada, periksa kembali kelengkapan dan keakuratannya. Laporkan dokumen DPPL dengan *Sequence Diagram* yang sudah lengkap pada Tugas Pendahuluan.

Pada DPPL kelompok kami sudah terdapat *Sequence Diagram* untuk setiap *Use Case* yang ada. Jumlah dari *Sequence Diagram* yang berada pada DPPL kelompok kami adalah 31. Link untuk mendownload SKPL dan DPPL ada di <https://github.com/Kelompok8RPLIF4203/modul4>. Berikut *Sequence Diagram* yang ada pada DPPL kelompok kami :

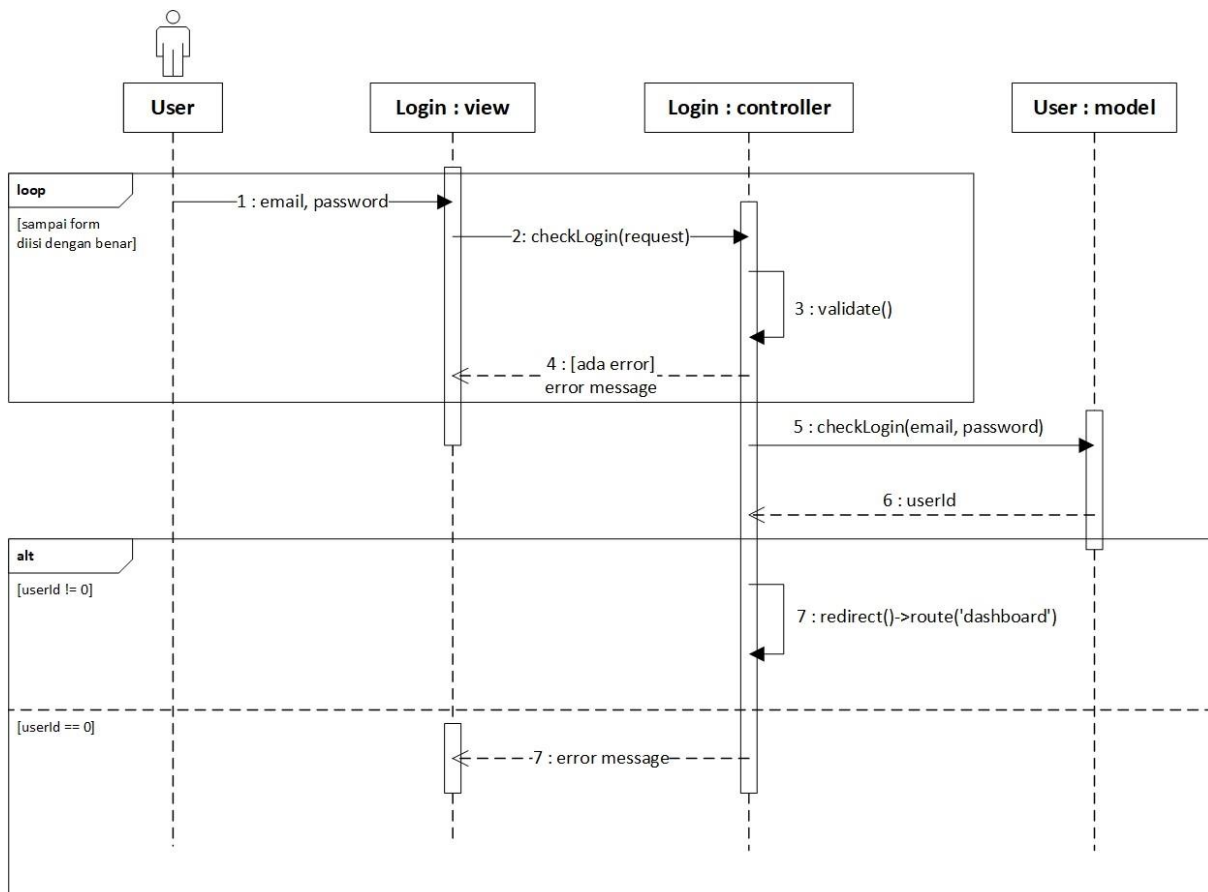
Use Case #1 (Daftar Akun)



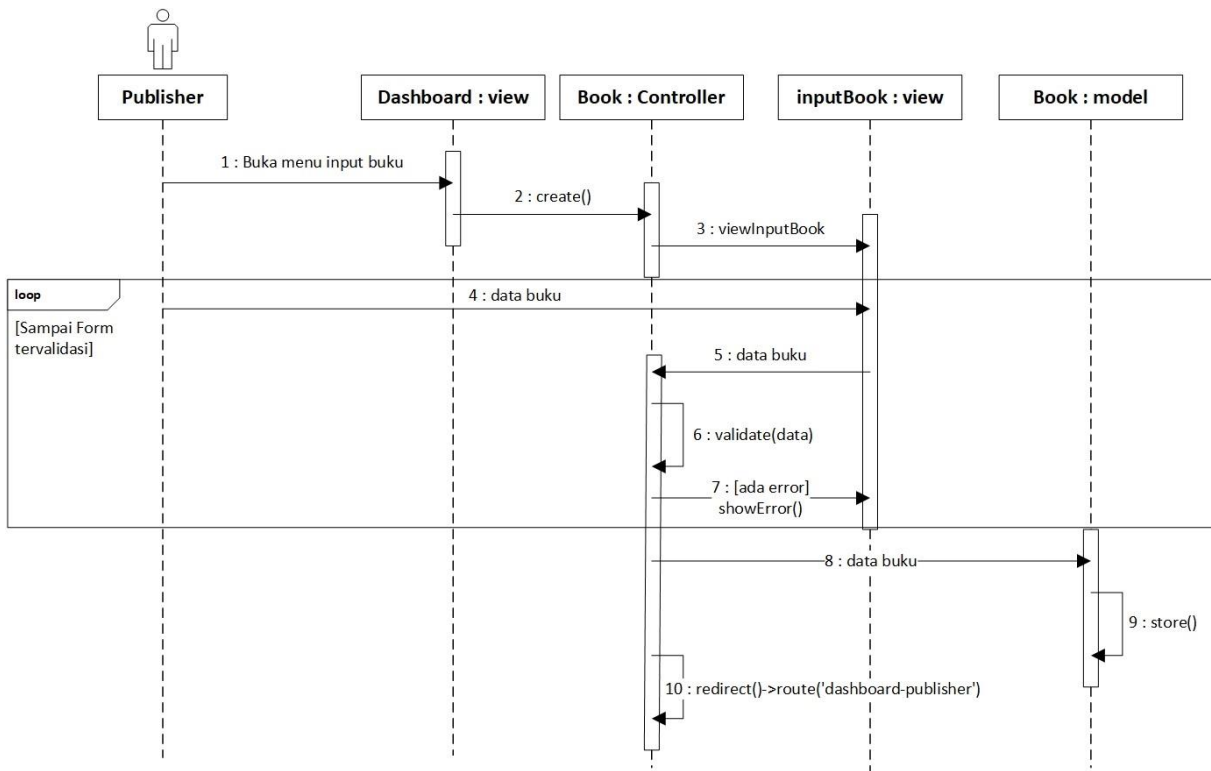
Use Case #2 (Verifikasi e-mail)



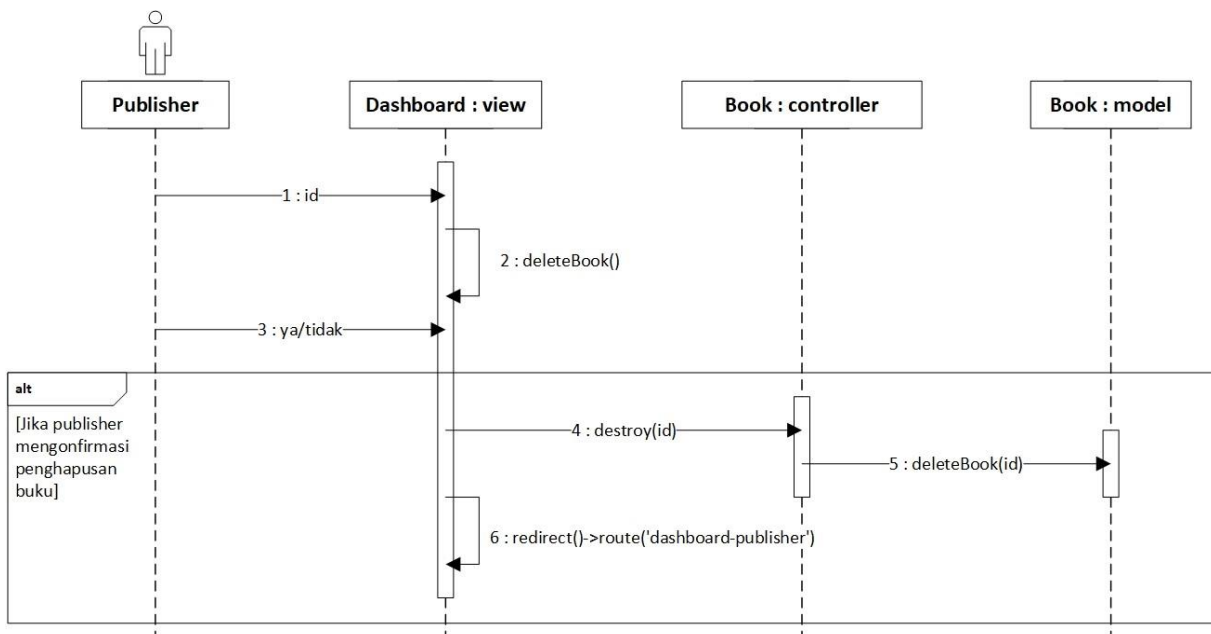
Use Case #3 (Login)



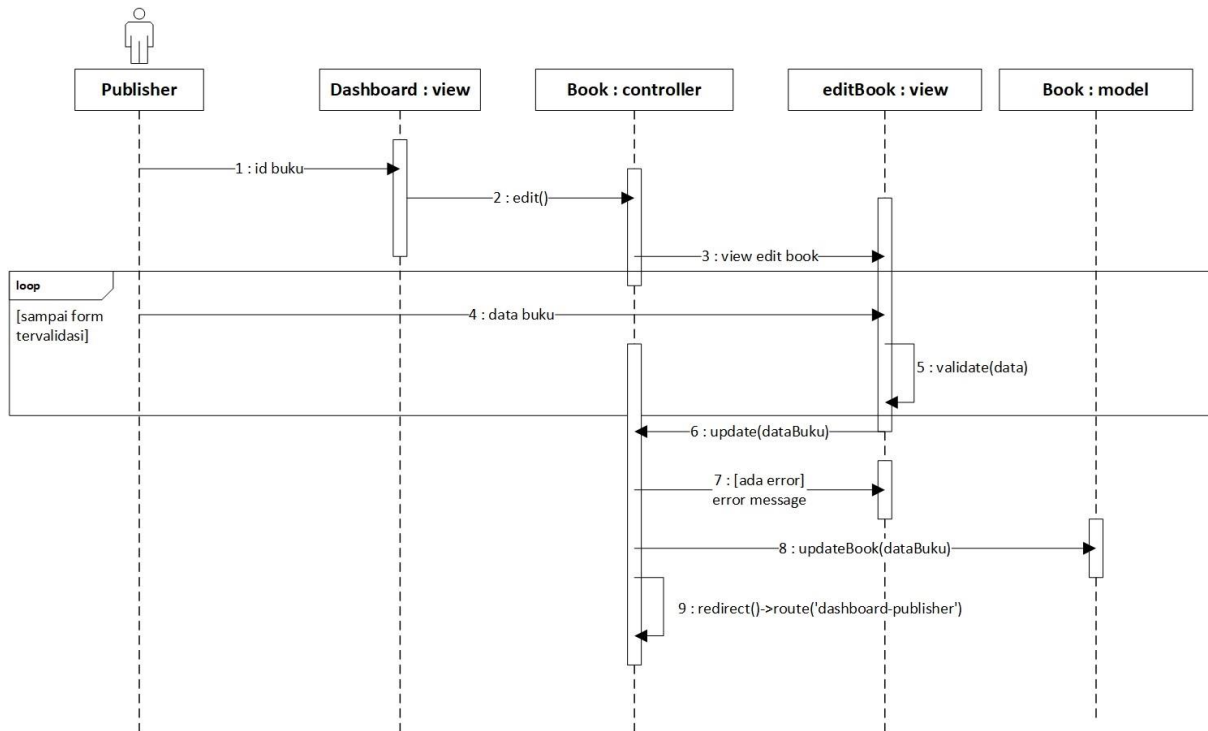
Use Case #4 (Input Buku)



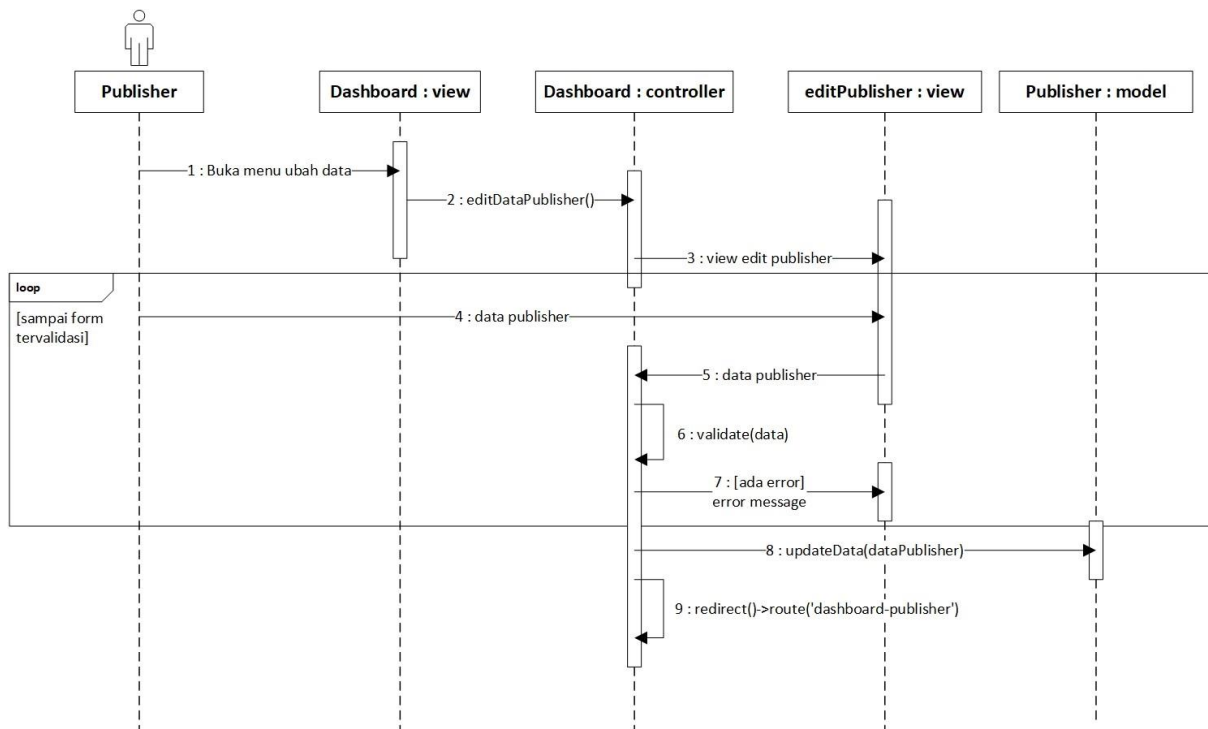
Use Case #5 (Hapus Buku)



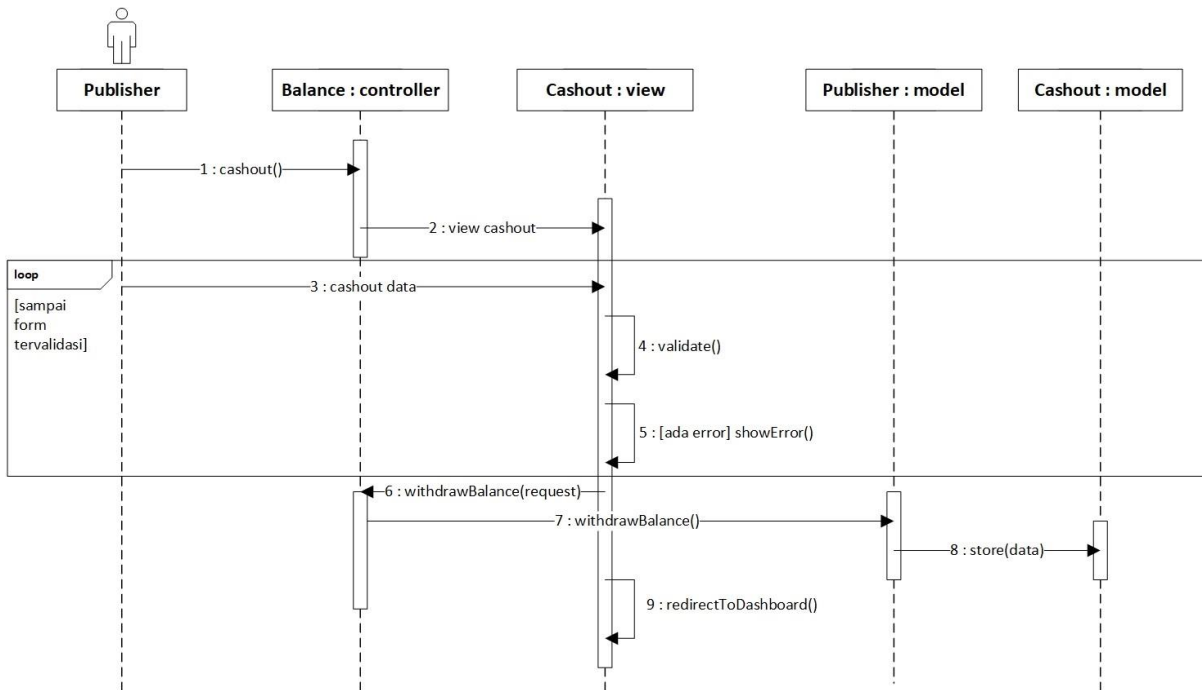
Use Case #6 (Ubah data diri)



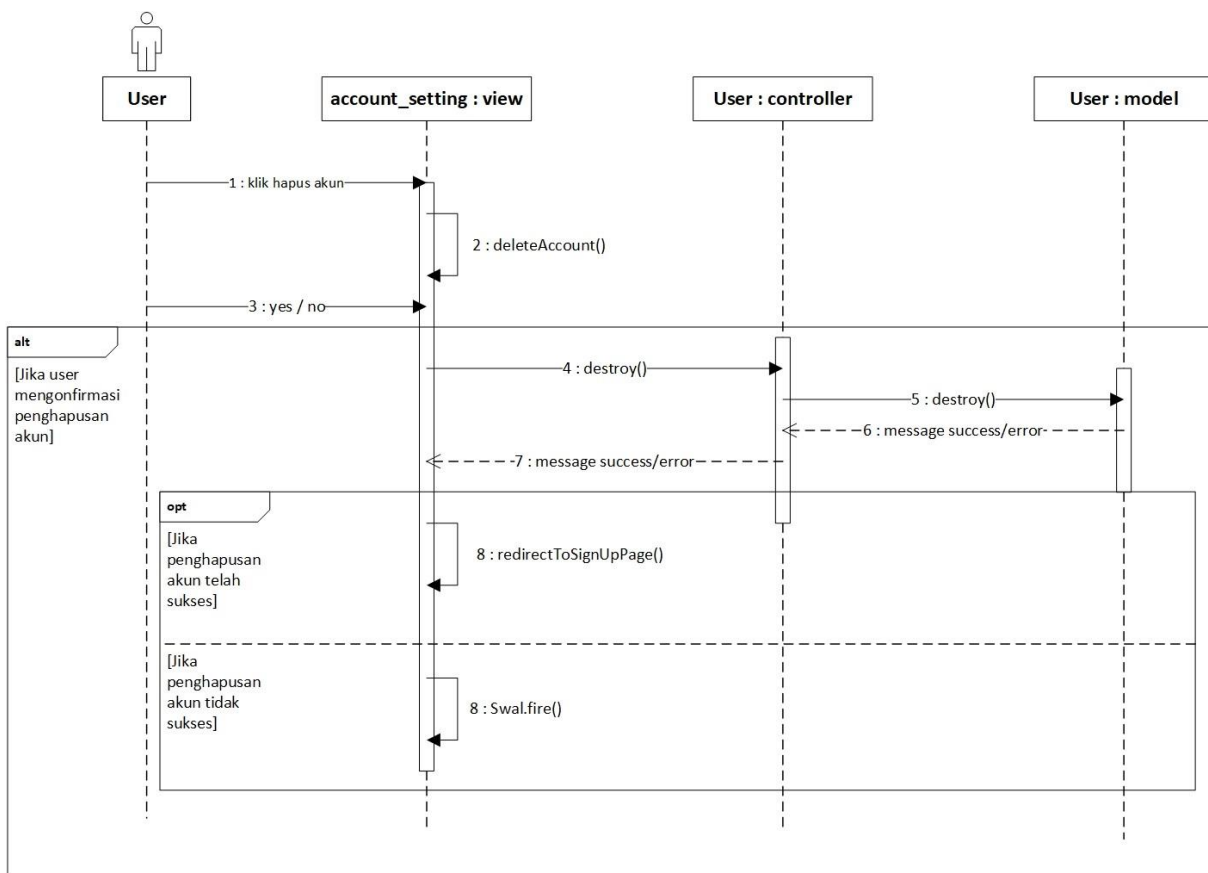
Use Case #7 (Data publisher)



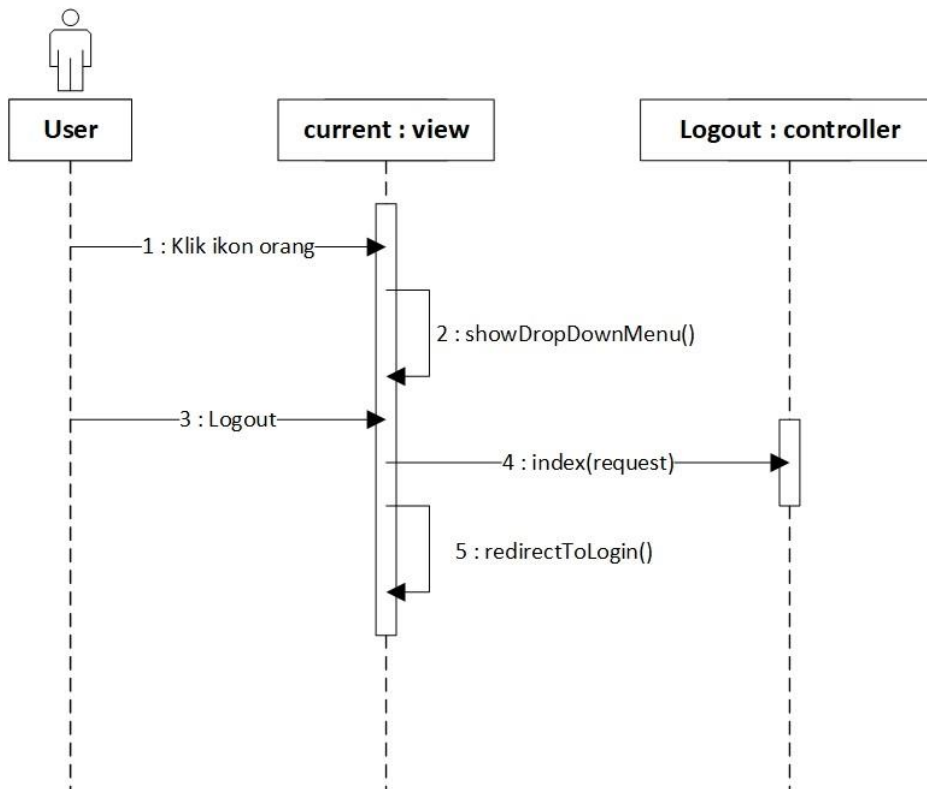
Use Case #8 (Cairkan saldo)



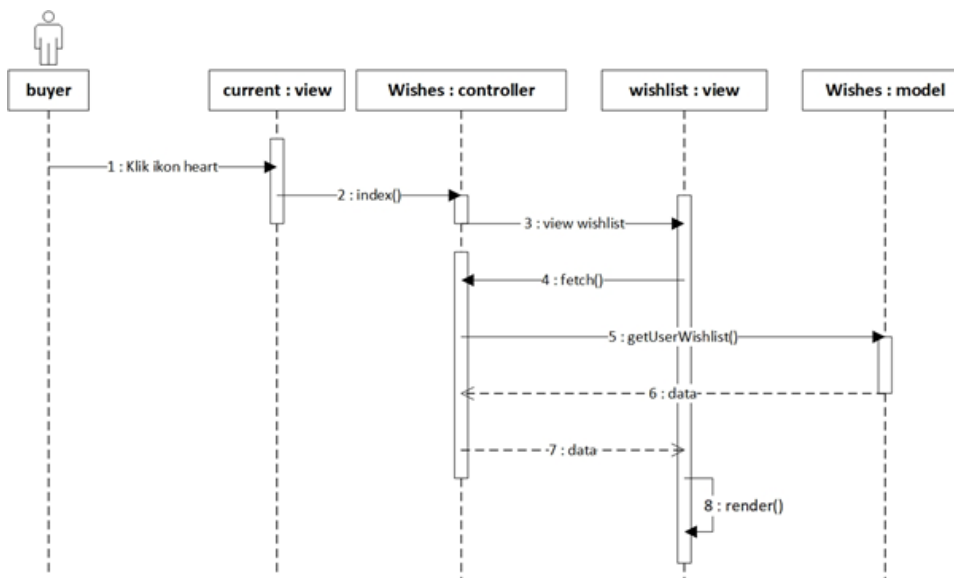
Use Case #9 (Hapus akun)



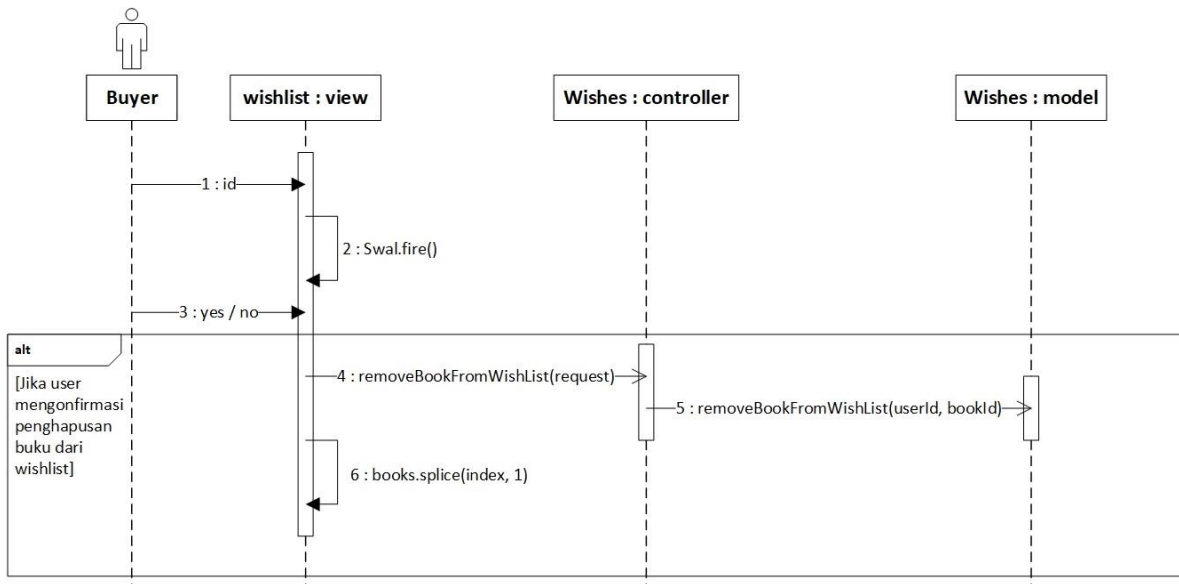
Use Case #10 (Logout)



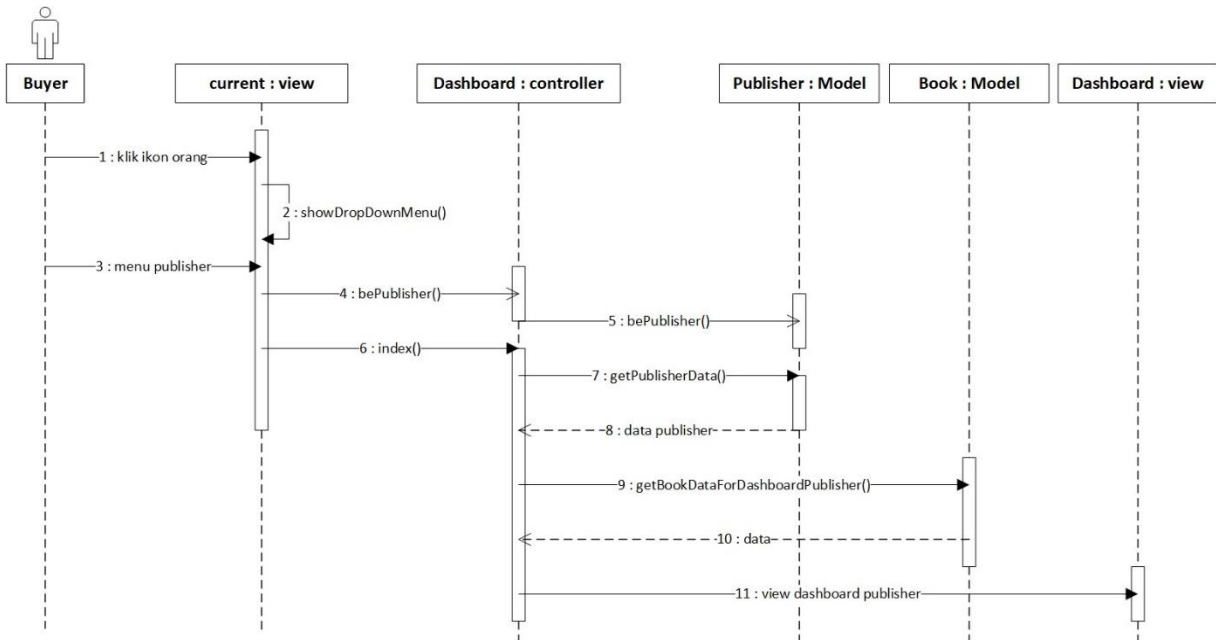
Use Case #11 (Lihat wishlist)



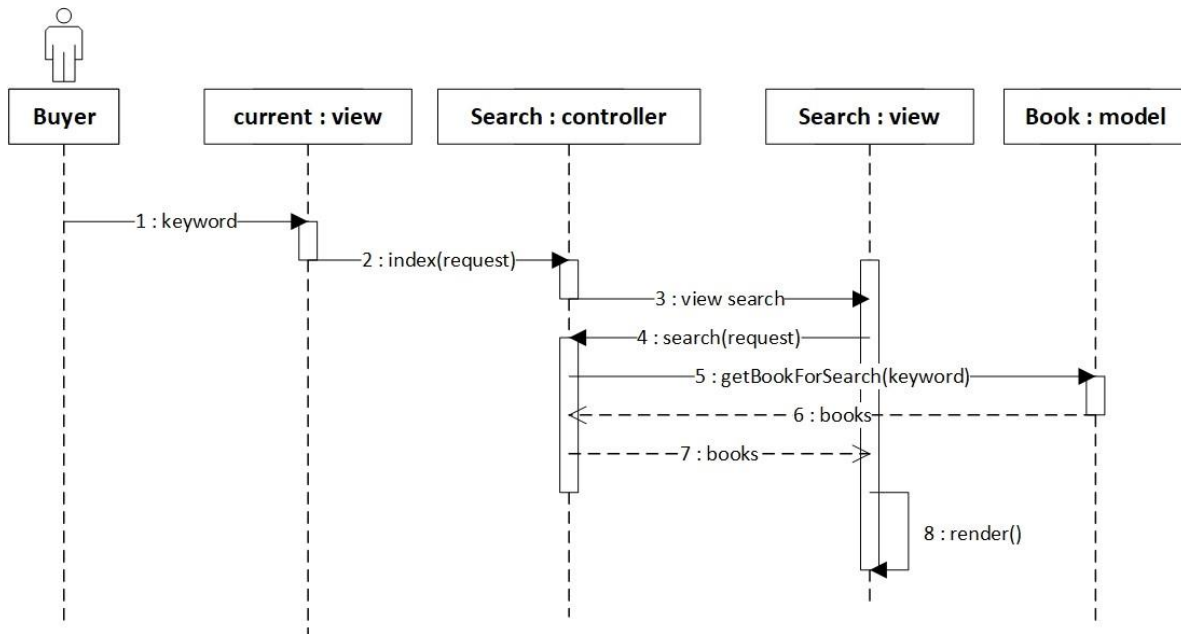
Use Case #12 (Hapus buku dari wishlist)



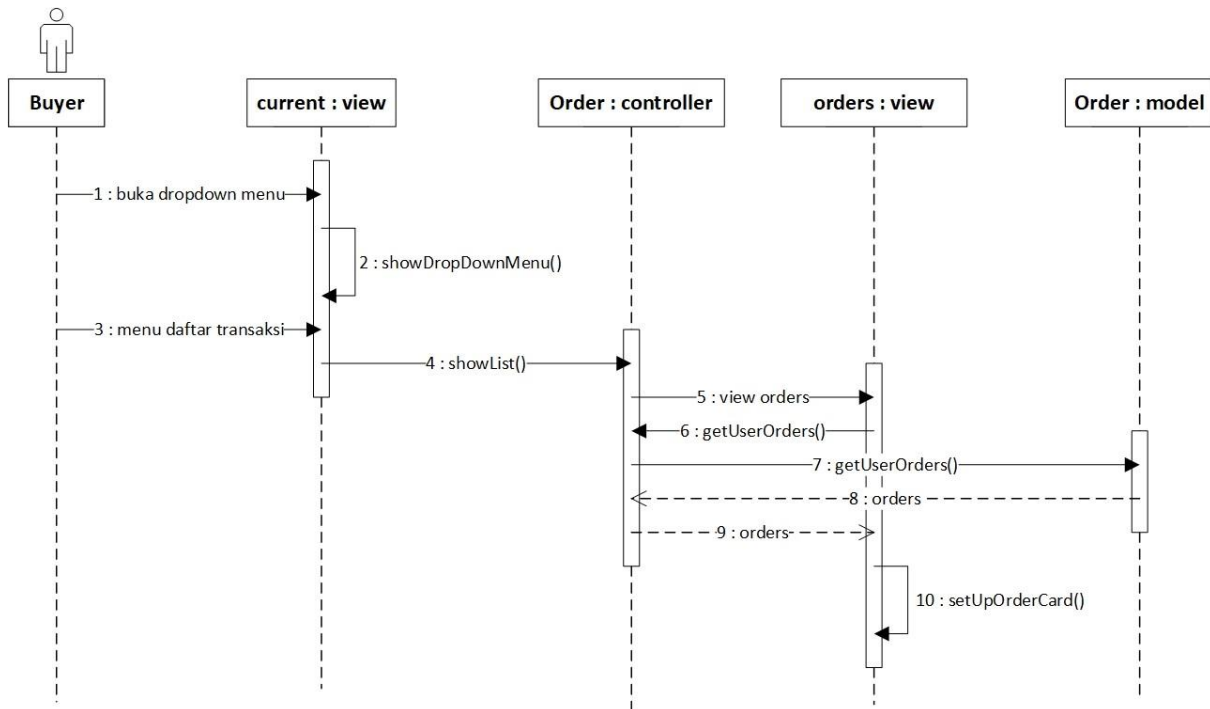
Use Case #13 (Buka menu publisher)



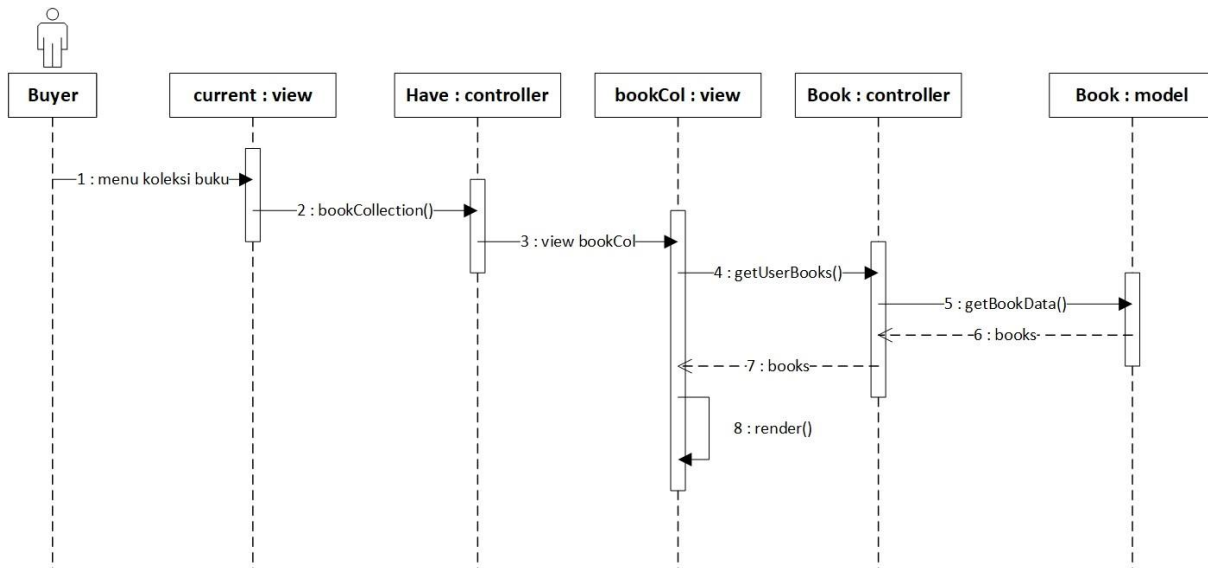
Use Case #14 (cari buku)



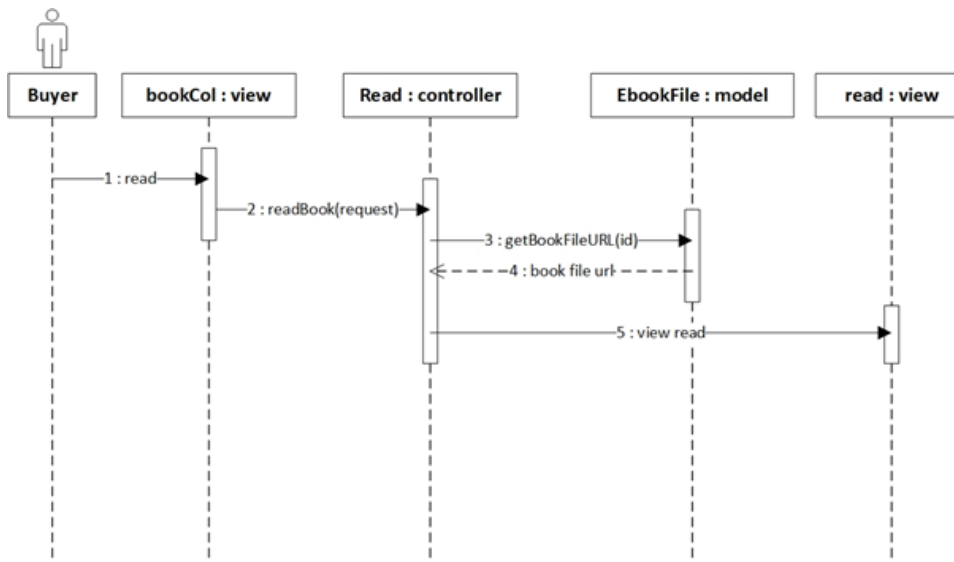
Use Case #15 (Daftar transaksi)



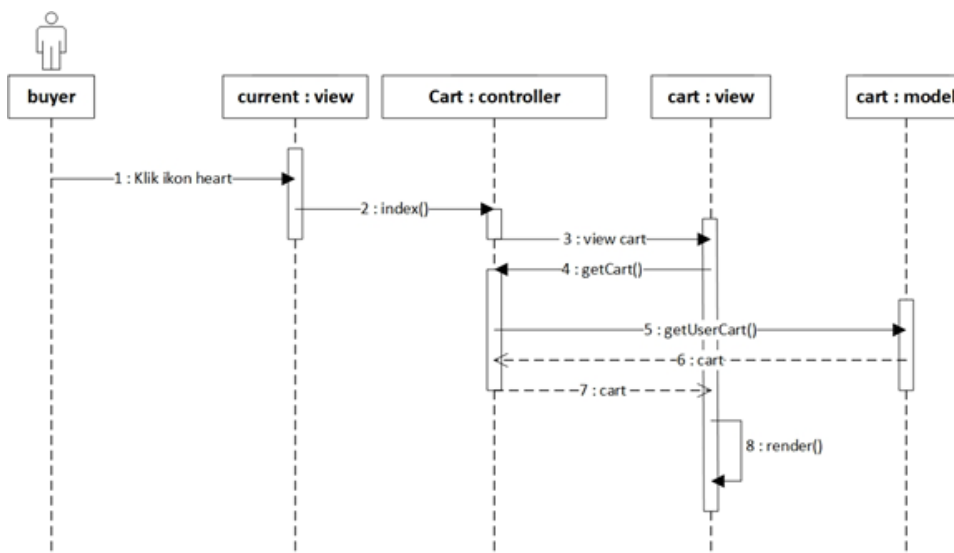
Use Case #16 (Lihat koleksi buku)



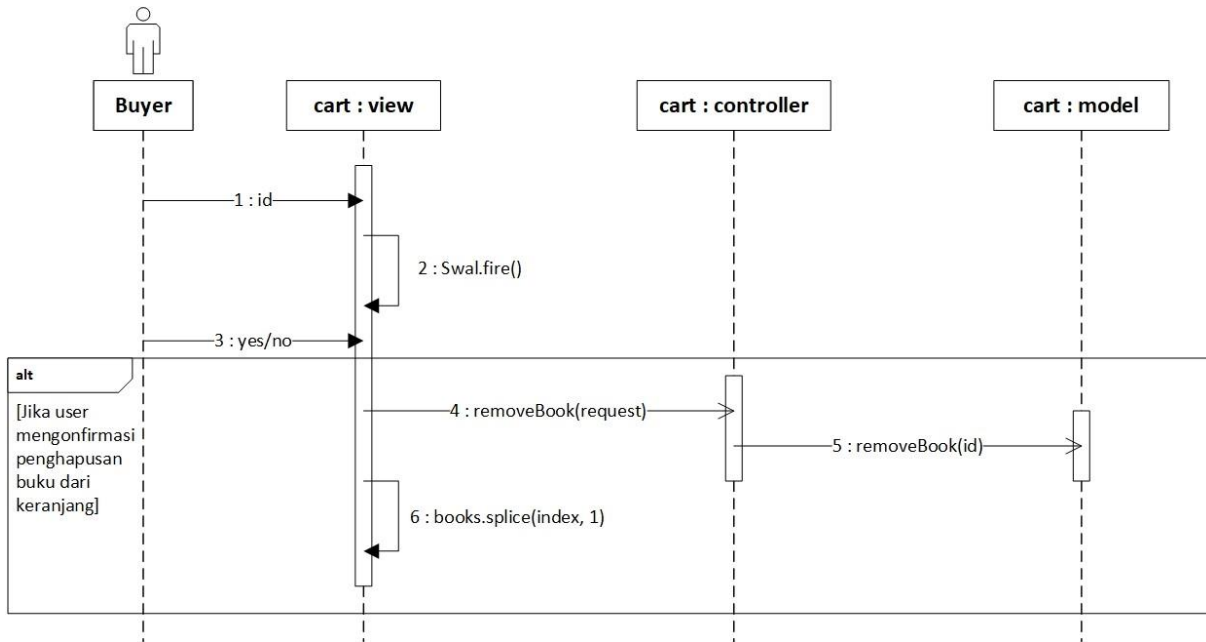
Use Case #17 (Baca buku)



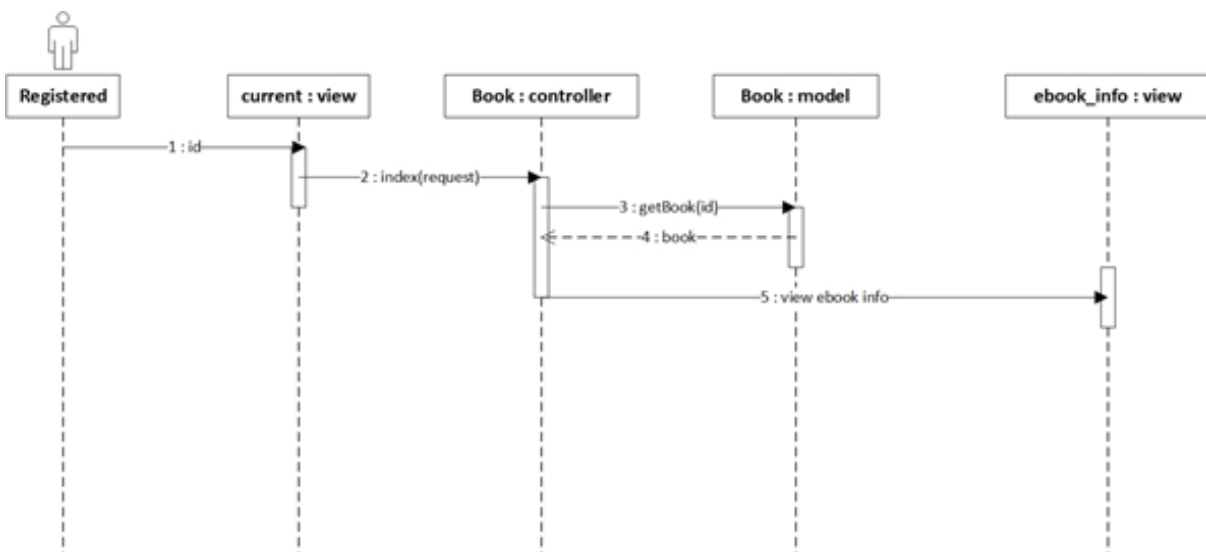
Use Case #18 (Lihat keranjang belanja)



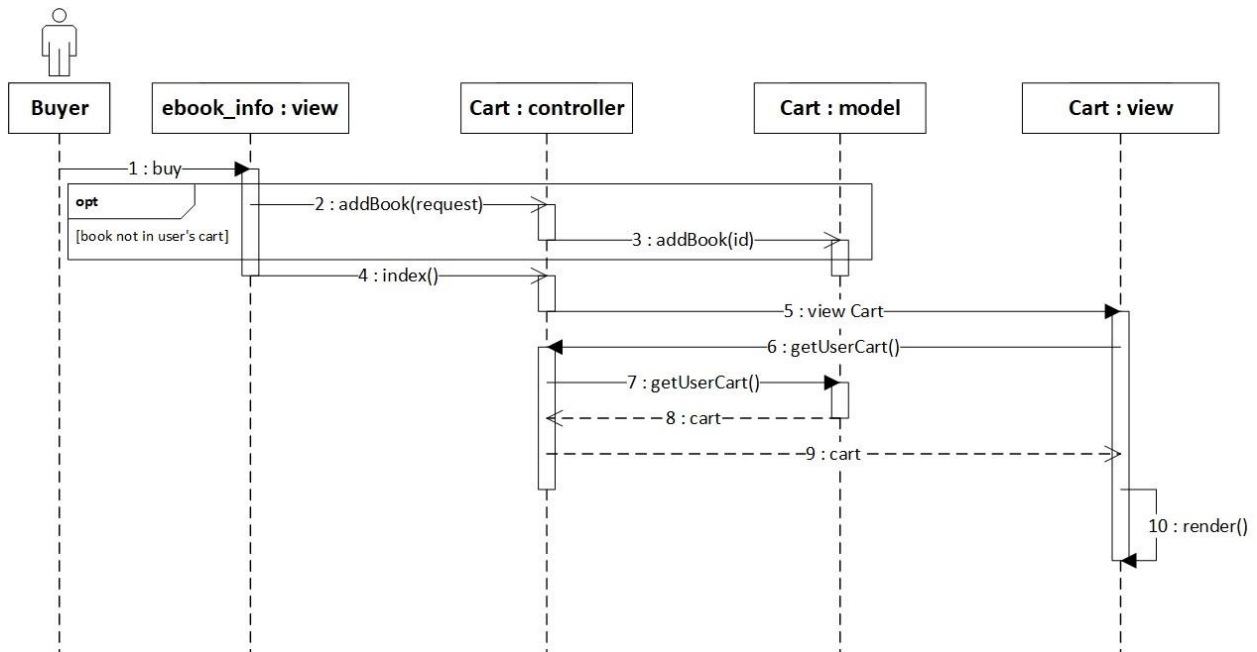
Use Case #19 (Hapus buku dari keranjang)



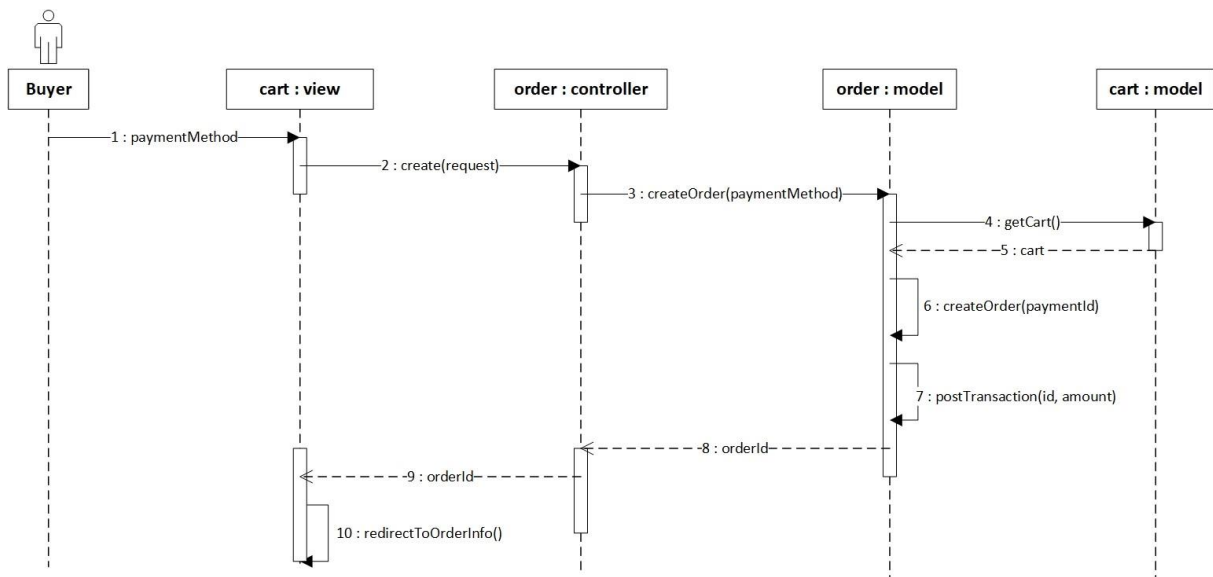
Use Case #20 (Lihat Info Buku)



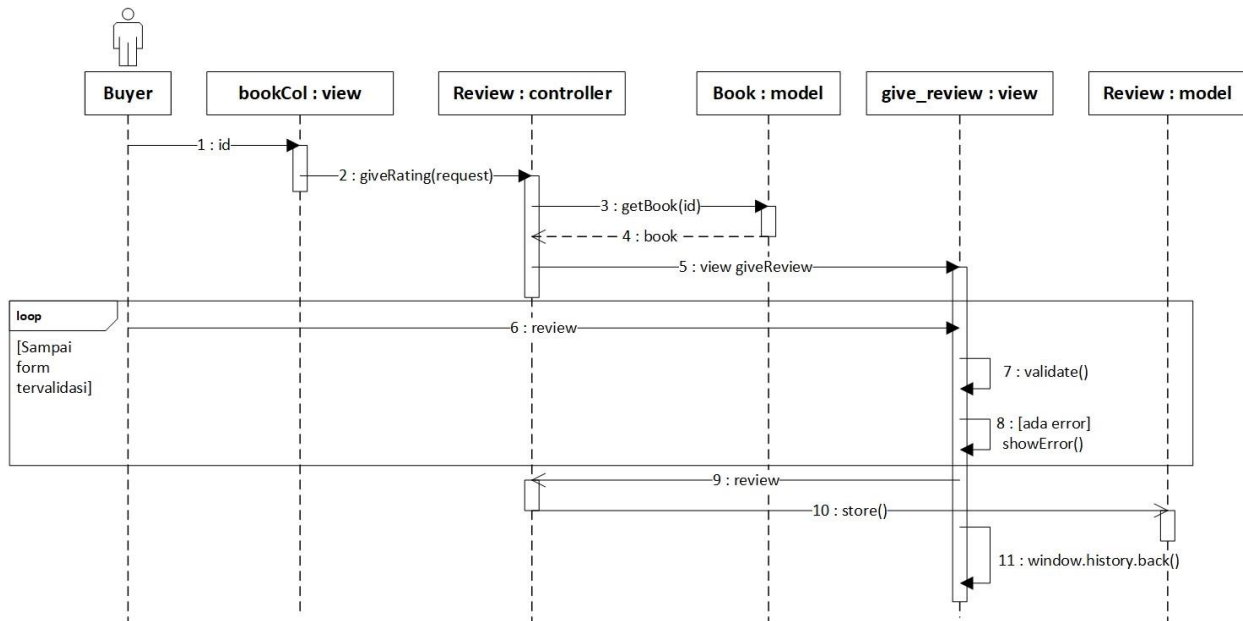
Use Case #21 (Beli Buku)



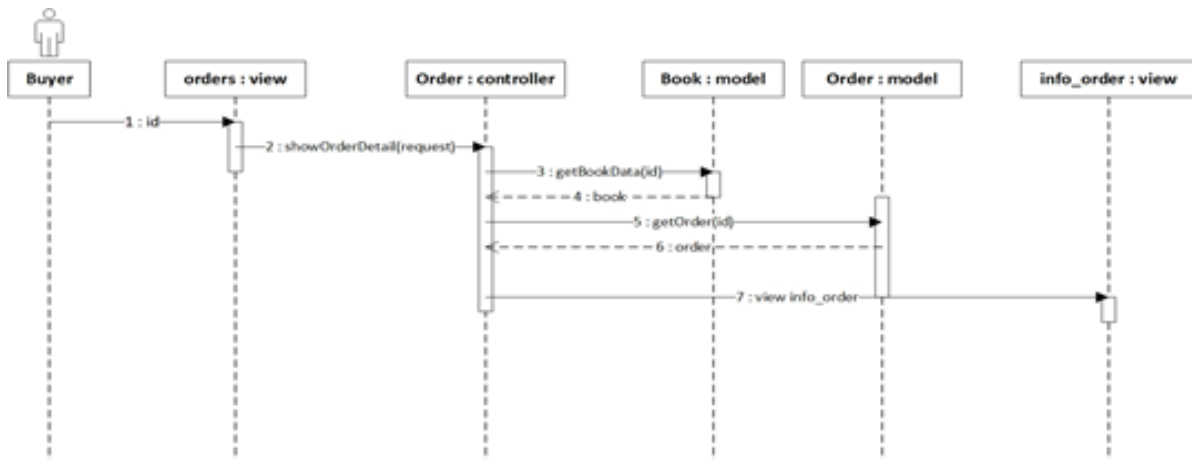
Use Case #22 (Checkout Pesanan)



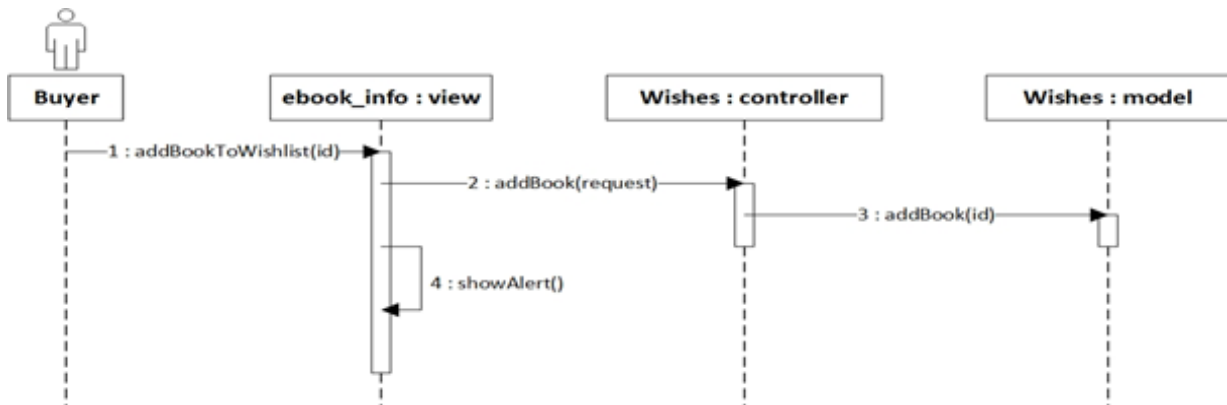
Use Case #23 (Memberikan Ulasan)



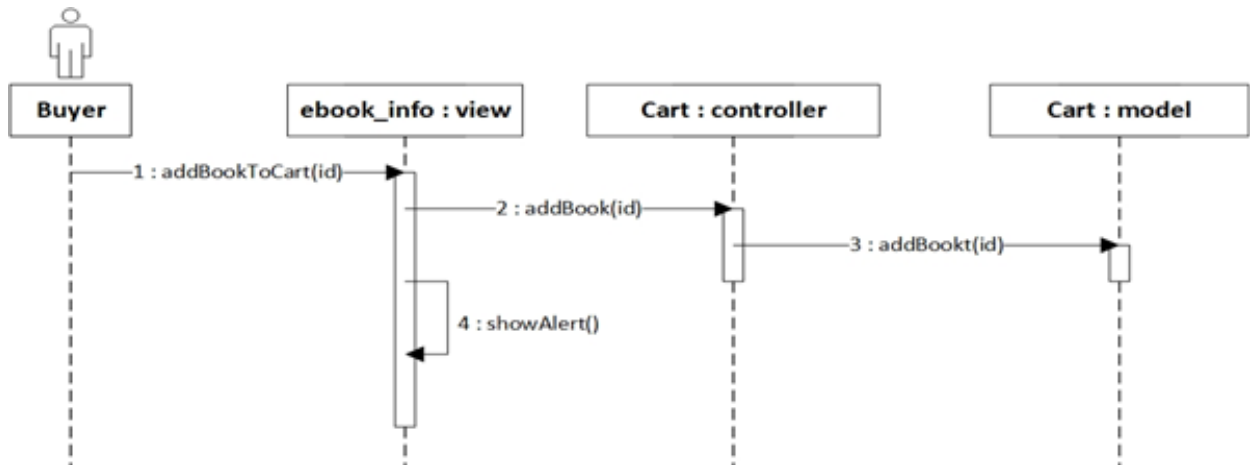
Use Case #24 (Detail Transaksi)



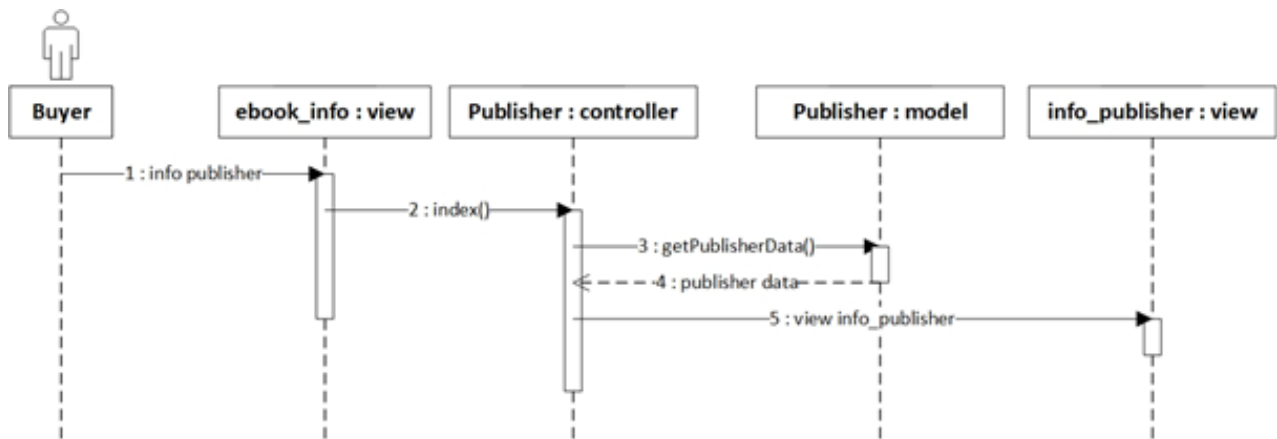
Use Case #25 (Tambah Buku Ke Wishlist)



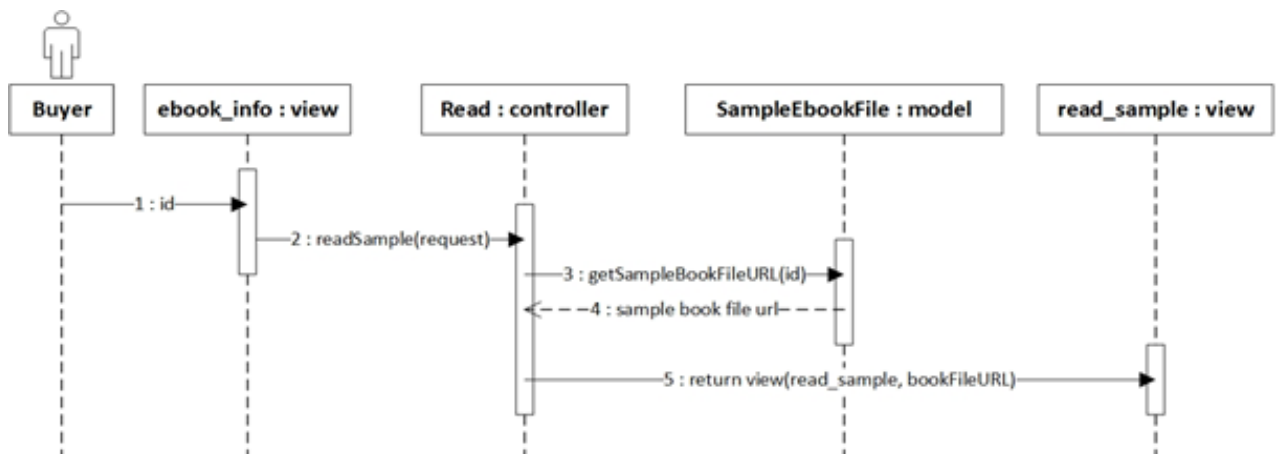
Use Case #26 (Tambah Buku Ke Keranjang)



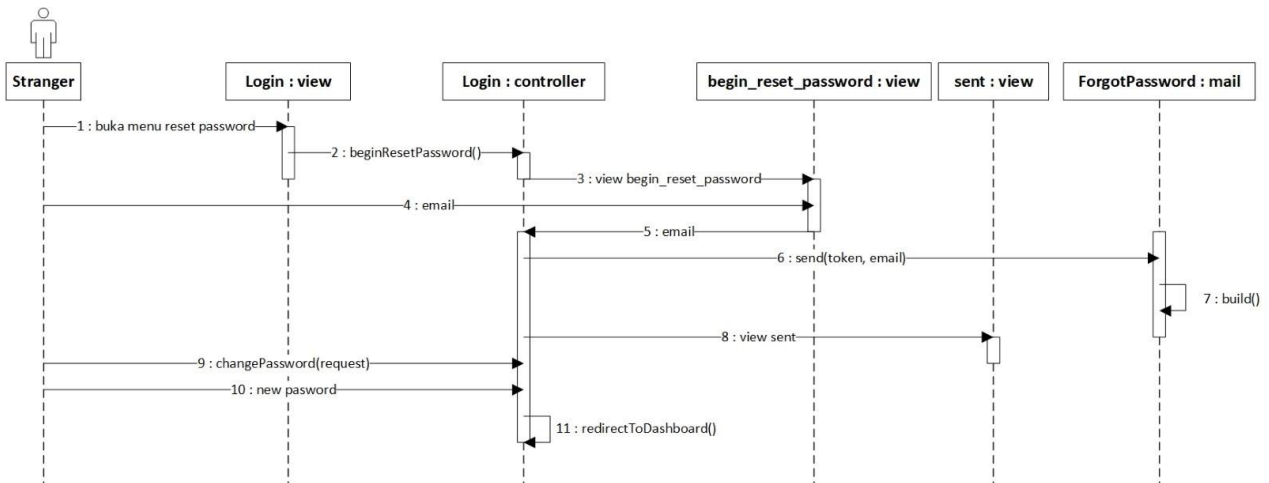
Use Case #27 (Lihat Info Publisher)



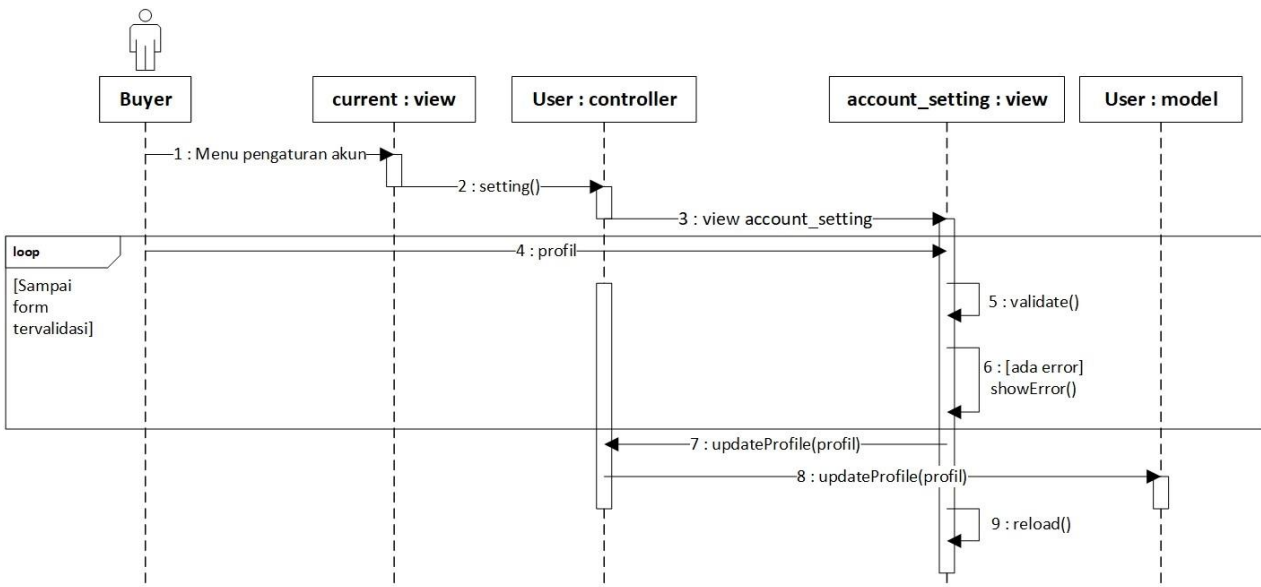
Use Case #28 (Baca Sample Buku)



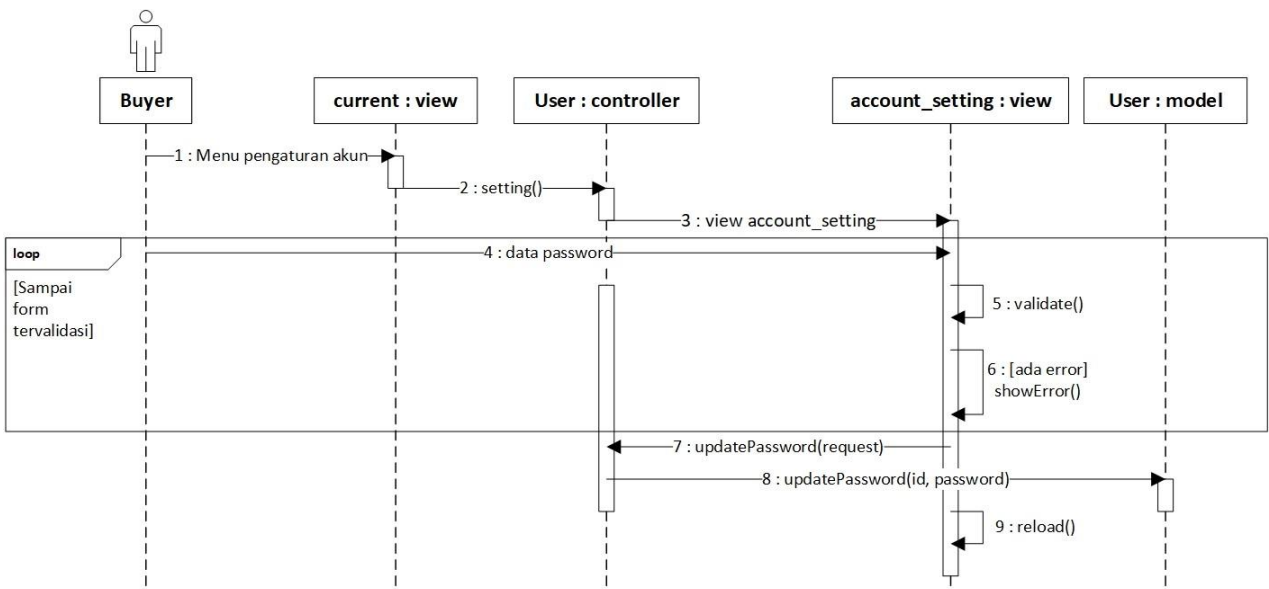
Use Case #29 (Reset Password)



Use Case #30 (Ubah Profil)



Use Case #31 (Ubah Password)



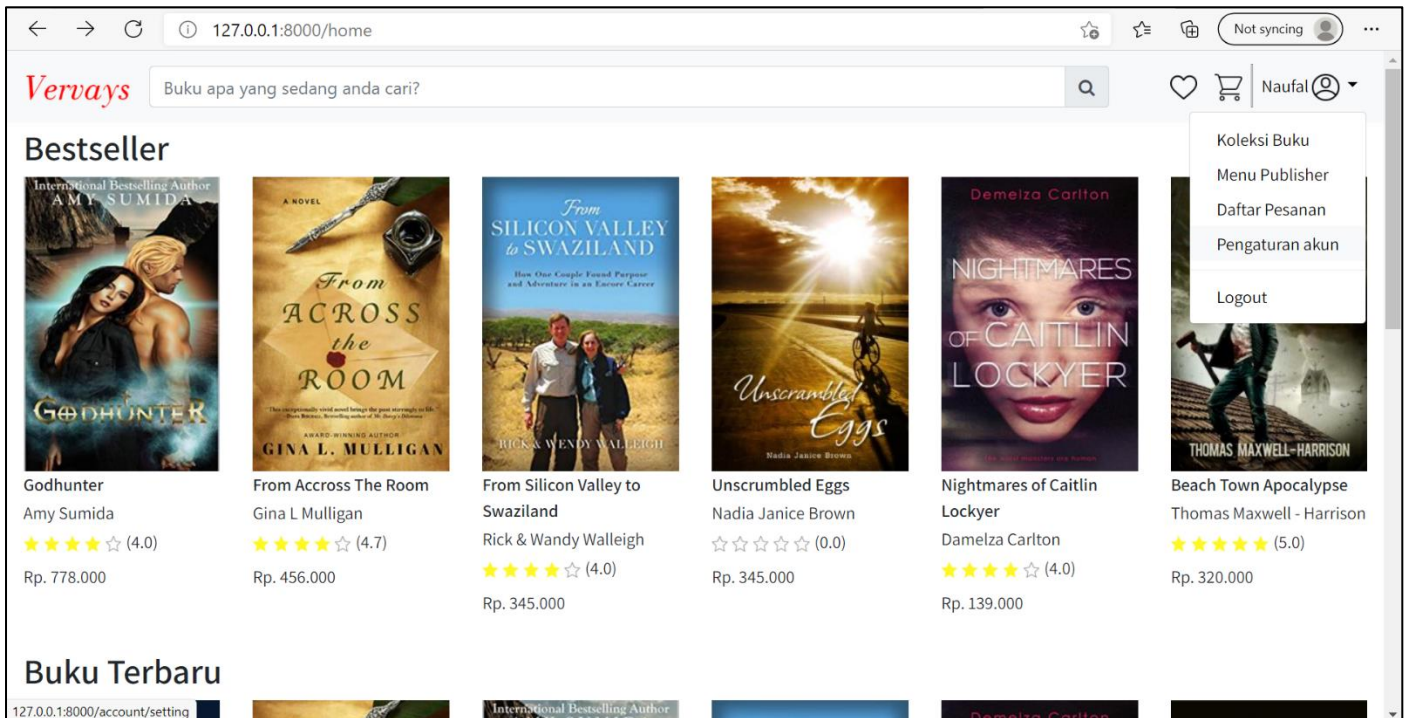
TUGAS LAB

Sequence Diagram pada umumnya menggambarkan interaksi *Actor* dengan sistem dalam satu rangkaian masukan dan keluaran. Masukan dari *Actor* via IMK, akan direspon oleh aplikasi melalui pemanggilan *method-method* dari *class-class* yang saling berinteraksi, yang jika diperlukan sampai perlu mengakses tabel basis data yang relevan dengan SQL yang di-"*embed*" di bahasa pemrograman, lalu memberikan hasil pengaksesannya ke *Actor* via IMK lagi.

Implementasikan setidaknya satu *Sequence Diagram* yang dapat bekerja sesuai deskripsi di atas.

Use Case #31 (Ubah Password)

1. Pada ikon profil, user memilih menu pengaturan akun



2. View akan memanggil method `setting()` yang ada di kelas `UserController()`

```
class UserController extends Controller
{
    public function setting()
    {
        $data = [
            "firstName" => User::getFirstName(session('id')),
            "user" => User::getUserDataForAccountSettingPage(session('id')),
            "currentDate" => date("Y-m-d"),
        ];
        // dd($data["user"]);
        return view('pages.buyer.account_setting', $data);
    }
}
```

3. Method `setting()` memanggil method `getFirstName()` agar nama awal pengguna bisa ditampilkan nantinya

```
public static function getFirstName($id)
{
    $result = DB::table('users')
        ->where('id', $id)
        ->select('firstName')
        ->get();
    return $result[0]->firstName;
}
```

4. Method `setting()` memanggil method `getUserDataForAccountSettingPage()` agar data pengguna bisa digunakan

```
public static function getUserDataForAccountSettingPage($userId)
{
    $user = DB::table('users')
        ->where('id', $userId)
        ->select('id', 'email', 'firstName', 'lastName', 'birthDay', 'phoneNumber', 'gender',
        ->first();
    $user->name = User::getFirstName($userId)." ".(User::getLastName($userId) ?? "");
    $parsedDate = Carbon::parse($user->created_at);
    $user->created_at_month = $parsedDate->month;
    $user->created_at_year = $parsedDate->year;
    return $user;
}
```

5. Method `setting()` memanggil method `date()` agar program dapat mengetahui tanggal sekarang
6. Method `setting()` kemudian menampilkan halaman pengaturan akun

The screenshot shows the 'Ubah Profil' (Change Profile) page of a user named Naufal Muafa. The page has a sidebar with navigation links: 'Ubah Profil' (selected), 'Ubah Password', and 'Hapus Akun'. The main content area contains several form fields: 'Nama depan' (First Name) with the value 'Naufal', 'Nama belakang' (Last Name) with the value 'Muafa', 'Tanggal Lahir (opsional)' (Optional Birth Date) with the value '06/02/2000', 'No HP (opsional)' (Optional Phone Number) with the value '08xxxxxxxx', and 'Jenis Kelamin (opsional)' (Optional Gender) with radio buttons for 'Laki-laki' (selected) and 'Perempuan'. A red 'Ubah Profil' button is located at the bottom right of the form.

7. User memilih menu ubah password

8. User memasukkan password lama dan password baru dan klik ubah password

9. Data password lama dan password baru dikirimkan oleh 'View Account setting' ke method `updatePassword(Request $request)` yang ada di `UserController`

```
public function updatePassword(Request $request)
{
    return User::updatePassword(session('id'), $request->post('password'));
}
```

10. Method `updatePassword()` yang ada di kelas `UserController` memanggil method `updatePassword()` yang ada di kelas `User`.

```
public static function updatePassword($userId, $newPassword)
{
    $value = DB::table('users')
        ->where('id', $userId)
        ->where('isDeleted', '0')
        ->first();
    if ($value != null) {
        DB::table('users')
            ->where('id', $userId)
            ->where('isDeleted', '0')
            ->update([
                "password" => Hash::make($newPassword),
            ]);
        PasswordResetToken::deleteByUserId($userId); //Menghapus token reset password jika ada
        return true; // Tanda jika berhasil update password
    }
    return false; // Tanda jika gagal update password
}
```

11. Pada method tersebut, sistem mengubah password user pada database.

12. 'View Account Setting' kemudian melakukan reload halaman

```
location.reload();
```

Selamat bekerja