



PRAKTIKUM PEMROGRAMAN BERORIENTASI OBJEK 2017/2018

CASE STUDY LANJUTAN

MODUL 9: Mobile GUI

Pada modul ini data yang dikumpulkan adalah:

- **1 commit** pada Github dengan batas waktu adalah **5 Mei 2018 23:55**
- **1 laporan/dokumentasi** pada SCELE dengan batas waktu **5 Mei 2018 23:55**

Pendahuluan

Pada JHotel, kosongkan method main anda kecuali baris `SpringApplication.run()`. Lalu pada main, tambahkan 1 hotel dan 3 kamar pada hotel tersebut, lalu jalankan program JHotel anda dengan `bootRun` selama mengerjakan CS ini.

Note: Buat laporan yang berisikan screenshot dari tahapan-tahapan yang ada.

Tugas 1: Buat activity untuk membuat Pesanan

- Buat empty activity baru, beri nama `BuatPesananActivity`
- Tambahkan 4 variable, (semua private) yakni `int currentUserId`, `banyakHari` & `idHotel`, `double tariff`, `String roomNumber`.
- Ubah layout menjadi seperti berikut

Room Number : TextView

Tariff : TextView

Durasi : _____

Total : TextView

HITUNG

PESAN



Dengan ketentuan nama/id component dan isi component sebagai berikut

```
Ab total_biaya (TextView) - "TextView"
Ab staticTotal (TextView) - "Total :"
Ab staticDurasi (TextView) - "Durasi :"
Ab staticTarrif (TextView) - "Tariff :"
Ab tariff (TextView) - "TextView"
Ab staticRoomNumber (TextView) - "Room Number :"
Ab room_number (TextView) - "TextView"
■ hitung (Button) - "Hitung"
■ pesan (Button) - "Pesan"
Ab durasi_hari (EditText) (Number)
```

- d. Inisiasi semua component pada layout, kecuali yang diawali dengan static.
- e. Pada onCreate, ubahlah atau tambahkanlah kode sehingga activity memenuhi ketentuan berikut:
 - Buat tombol Pesan menghilang/tidak terlihat.
 - Berikan nilai pada field room_number sesuai dengan variable roomNumber pada activity.
 - Berikan nilai pada field tariff sesuai dengan variable tariff pada activity.
 - Berikan nilai "0" pada field total_biaya
- f. Selanjutnya, buat OnClickListener untuk tombol hitung. Buatlah jika tombol hitung diclick, maka variable banyakHari bernilai sama dengan inputan pada field durasi_hari & field total_biaya bernilai $\text{tariff} \times \text{banyakHari}$. Selanjutnya hilangkan tombol hitung dan tampilkan tombol pesan.
- g. Buat class baru bernama BuatPesananRequest yang merupakan subclass dari StringRequest. Pada request ini, URL yang dituju adalah URL API yang bertujuan untuk membuat pesanan baru. Sesuaikan parameter yang dibutuhkan. Tampilkan feedback jika pesanan berhasil maupun gagal!

Tugas 2: Ubah LoginActivity&MenuActivity

- a. Untuk dapat membuat pesanan, diperlukan ID Customer, dimana ID Customer itu sendiri didapatkan pada halaman LoginActivity. Buatlah agar saat transisi antara LoginActivity dengan MenuActivity ID dari user ikut terbawa! (Hint: gunakan intent dan putExtra).



- b. Pada MenuActivity, tambahkan instance variable currentUserId. Nilai dari currentUserId didapatkan dari nilai ID yang telah di-'pass' oleh LoginActivity (Hint: ambil dari intent).
- c. Selanjutnya, pada Expandable List, tambahkan setOnChildClickListener. Pada method onChildClick dapatkan object Room yang sedang diclick. Anda dapat menggunakan potongan berikut

```
Room selected = childMapping.get(listHotel.get(groupPosition)).get(childPosition);
```

- d. Selanjutnya, buatlah sehingga jika child tersebut diclick, maka activity akan berpindah ke BuatPesananActivity. Tidak hanya itu, berikan juga (pass atau putExtra) variable yang dibutuhkan untuk membuat pesanan (ID Customer, Room Number, ID Hotel).
- e. Ubah BuatPesananActivity sehingga dapat menerima variable yang dipass pada nomor d. Assign ke variable yang ada pada activity tersebut.
- f. Dengan ini, semua variable/parameter yang dibutuhkan untuk membuat pesanan dapat dimiliki BuatPesananActivity. Cobalah membuat pesanan dan coba tampilkan pesanan yang sudah dibuat (dari Android) melalui Postman.

Tugas 3 : Menambahkan Pembatalan dan Penyelesaian Pemesanan

- a. Buatlah empty Activity dan namakan SelesaiPesananActivity.
- b. Tambahkan tombol berjudul pesanan pada MenuActivity. Buatlah jika tombol tersebut ditekan, maka activity akan dipindahkan ke SelesaiPesananActivity. Selain itu, pastikan Customer ID juga ikut dikirimkan ke SelesaiPesananActivity.
- c. Buatlah class baru bernama PesananFetchRequest. Gunakan URL API yang digunakan untuk mendapatkan Pesanan dari suatu Customer (pesananCustomer) gunakan parameter yg sesuai.
- d. Buatlah class baru bernama PesananSelesaiRequest. Gunakan URL API dan parameter yang sesuai.
- e. Buatlah class baru bernama PesananBatalRequest. Gunakan URL API dan parameter yang sesuai.
- f. Pada SelesaiPesananActivity, ambil Customer ID yang dipass dari MenuActivity.



-
- g. Buatlah layout yang mampu menampilkan karakteristik dari pesanan (ID Pesanan, biaya, Jumlah Hari, tanggal pesan). Selain itu tambah 2 buat tombol, satu untuk Batal dan satu untuk Selesai. Buat layout semenarik mungkin.
 - h. Pada onCreate di SelesaiPesananActivity, ubah semua komponen layout menjadi tak terlihat.
 - i. Buatlah method fetchPesanan() (dan pastikan method dipanggil pada onCreate), dimana didalamnya buatlah mekanisme (request, & queue) untuk mengambil data pesanan dari customer (menggunakan PesananFetchRequest yang dibuat pada nomor C).
 - j. Buatlah conditional Statement dimana jika fetch tidak menghasilkan JSON Apapun (tidak ada pesanan) maka user akan dilempar kembali ke MenuActivity. Sebaliknya, jika JSON dihasilkan, ubah semua komponen layout menjadi terlihat dan tampilkan data pesan sesuai field yang sudah anda buat.
 - k. Buatlah onClickListener untuk tombol batal dan selesai, dimana masing-masing akan menjalankan proses request yang sesuai dengan namanya (batal akan menjalankan request PesananBatalRequest dan selesai akan menjalankan PesananSelesaiRequest). Buatlah feedback yang sesuai jika proses penyelesaian atau pembatalan berhasil.
 - l. Lakukan pengujian pembatalan dan penyelesaian pesanan menggunakan aplikasi yang telah anda buat. Selanjutnya, lakukan pengecekan pesanan tersebut melalui Postman.