

## Skenario Testing – DJetpack Movie

### Unit Test

#### MainViewModelTest

1. Memuat data **Movie**
  - Memastikan data **movies** tidak *null*.
  - Memastikan data yang dihasilkan sama dengan yang diharapkan.
2. Memuat data **TV Show**
  - Memastikan data **tvShows** tidak *null*.
  - Memastikan data yang dihasilkan sama dengan yang diharapkan.

#### FavoriteViewModelTest

1. Memuat data favorite **Movie**
  - Memastikan data **movies** tidak *null*.
  - Memastikan data yang dihasilkan sama dengan yang diharapkan.
2. Memuat data favorite **TV Show**
  - Memastikan data **tvShows** tidak *null*.
  - Memastikan data yang dihasilkan sama dengan yang diharapkan.

#### DetailViewModelTest

1. Menambahkan favorit **Movie**
  - Memastikan data **movie** menjadi Favorite
2. Menambahkan favorit **Tv Show**
  - Memastikan data **tv show** menjadi Favorite
3. Menghapus favorit **Movie**
  - Memastikan data **movie** sudah tidak menjadi Favorite
4. Menghapus favorit **Tv Show**
  - Memastikan data **tv show** sudah tidak menjadi Favorite
5. Cek apakah data yang ditampilkan adalah favorite
  - Memastikan movie/tv show adalah favorite

#### SearchViewModel

1. Memuat data favorite **Movie** sesuai dengan query
  - Memastikan data **movies** tidak *null*
  - Memastikan data yang dihasilkan sama dengan yang diharapkan.
2. Memuat data favorite **TV Show** sesuai dengan query
  - Memastikan data **tvShows** tidak *null*
  - Memastikan data yang dihasilkan sama dengan yang diharapkan.

**WAJIB - Hapus data dan cache aplikasi terlebih dahulu sebelum melakukan instrument test.**

**WAJIB – Ubah nilai menjadi 'true' untuk mengaktifkan idling resource Espresso**

Utils → EspressoIdlingResource.java

```
object EspressoIdlingResource {  
    //untuk instrument test ubah ke 'true'  
    private const val IDLING_RESOURCE = false  
}
```

## Instrument Test

MainActivityTest

1. Menampilkan data **Movies**
  - Memastikan **rv\_films** pada tab **MOVIES** dalam keadaan tampil.
  - Gulir **rv\_films** ke posisi data terakhir.
2. Menampilkan detail **Movie**
  - Melakukan klik pada data pertama **rv\_films** pada tab **MOVIES**.
  - Memastikan **TextView** untuk **tv\_title** tampil sesuai dengan yang diharapkan.
  - Memastikan **TextView** untuk **tv\_date\_release** tampil sesuai dengan yang diharapkan.
  - Memastikan **TextView** untuk **tv\_overview** tampil sesuai dengan yang diharapkan.
  - Memastikan **Menu** untuk **menu\_share** tampil sesuai dengan yang diharapkan.
3. Menampilkan data **TV Show**
  - Memastikan **rv\_films** pada tab **TV SHOW** dalam keadaan tampil.
  - Gulir **rv\_films** ke posisi data terakhir.
4. Menampilkan detail **TV Show**
  - Melakukan klik pada data pertama **rv\_films** pada tab **TV Show**.
  - Memastikan **TextView** untuk **tv\_title** tampil sesuai dengan yang diharapkan.
  - Memastikan **TextView** untuk **tv\_date\_release** tampil sesuai dengan yang diharapkan.
  - Memastikan **TextView** untuk **tv\_overview** tampil sesuai dengan yang diharapkan.
  - Memastikan **Menu** untuk **menu\_share** tampil sesuai dengan yang diharapkan.

FavoriteActivityTest

1. Daftar favorite **Movie** kosong
  - Memastikan layout empty list muncul Ketika daftar favorite kosong
2. Daftar favorite **Tv Show** kosong
  - Memastikan layout empty list muncul Ketika daftar favorite kosong
3. Menambahkan dan menghapus favorite **Movie**

- Menambahkan data **Movie** ke favorite
  - Cek apakah sudah masuk ke list favorite
  - Menghapus data **Movie** dari favorite
  - Cek apakah data sudah tidak ada di list favorite
4. Menambahkan dan menghapus favorite **Tv Show**
- Menambahkan data **Tv Show** ke favorite
  - Cek apakah sudah masuk ke list favorite
  - Menghapus data **Tv Show** dari favorite
  - Cek apakah data sudah tidak ada di list favorite

#### SearchActivityTest

1. Mencari data **Movie** – menemukan hasil
  - Memastikan pencarian data **Movie** menemukan hasil
2. Mencari data **Tv Show** – menemukan hasil
  - Memastikan pencarian data **Tv Show** menemukan hasil
3. Mencari data **Movie** – **tidak** menemukan hasil
  - Memastikan layout empty/error muncul Ketika pencarian data **Movie** tidak menemukan hasil
4. Mencari data **Tv Show** – **tidak** menemukan hasil
  - Memastikan layout empty/error muncul Ketika pencarian data **Tv Show** tidak menemukan hasil

\* Untuk melakukan Instrument Test dengan Idling Resource

1. Buka class 'DataViewModel.kt'
2. Ubah nilai 'forTest' menjadi 'true'
3. Kembalikan nilainya menjadi 'false' jika sudah tidak melakukan Instrument Testing

```
class DataViewModel(  
    private val repo: DataRepository,  
    private val networkHelper: NetworkHelper  
) : ViewModel() {  
  
    //untuk idling resource Instrument Test  
    private val forTest = false
```