

Unit Test

LocationViewModel

1. Ketika berhasil memuat data

- Memastikan Result.isSuccess bernilai true
- Memastikan Result.isFailure bernilai false
- Memastikan actualResponse tidak null
- Memastikan actualResponse sama dengan dummyStoriesResponse
- Memastikan method storyRepository.getAllStoriesWithLocation telah dipanggil

2. Ketika gagal memuat data

- Memastikan Result.isSuccess bernilai false
- Memastikan Result.isFailure bernilai true
- Memastikan data exception tidak null
- Memastikan method storyRepository.getAllStoriesWithLocation telah dipanggil

LoginViewModel

1. Ketika berhasil login

- Memastikan Result.isSuccess bernilai true
- Memastikan Result.isFailure bernilai false
- Memastikan data actualResponse tidak null
- Memastikan bahwa data actualResponse sama dengan expectedResponse
- Memastikan bahwa method authRepository.userLogin telah dipanggil

2. Ketika gagal login

- Memastikan bahwa Result.isSuccess bernilai false
- Memastikan bahwa Result.isFailure bernilai true
- Memastikan bahwa exception yang dikirimkan tidak null
- Memastikan bahwa method authRepository.userLogin telah dipanggil

3. Ketika menyimpan token authentication

- Memastikan bahwa method authRepository.saveAuthToken telah dipanggil

CreateViewModel

1. Ketika Berhasil Mendapatkan Token

- Memastikan bahwa actualToken tidak null
- Memastikan bahwa actualToken sama dengan dummyToken
- Memastikan bahwa method authRepository.getAuthToken() telah dipanggil
- Memastikan bahwa tidak ada interaksi dengan storyRepository

2. Ketika Berhasil Mendapatkan Token Tetapi Null

- Memastikan bahwa actualToken bernilai null
- Memastikan bahwa method authRepository.getAuthToken() telah dipanggil
- Memastikan bahwa tidak ada interaksi dengan storyRepository

3. Ketika Berhasil Mengupload File

- Memastikan bahwa Result.isSuccess bernilai true
- Memastikan bahwa Result.isFailure bernilai false
- Memastikan bahwa actualResponse tidak null
- Memastikan bahwa actualResponse sama dengan dummyUploadResponse

4. Ketika Gagal Mengupload File

- Memastikan bahwa Result.isSuccess bernilai false
- Memastikan bahwa Result.isFailure bernilai true
- Memastikan bahwa actualResponse yang berisi exception tidak null

HomeViewModel

1. Ketika Berhasil Mendapatkan Story

- Memastikan bahwa method homeViewModel.getAllStories telah dipanggil
- Memastikan bahwa nilai data story tidak null
- Memastikan ukuran data asli dengan data dummy sama

RegisterViewModel

1. Ketika Register Berhasil

- Memastikan bahwa Result.isSuccess bernilai true
- Memastikan bahwa Result.isFailure bernilai false
- Memastikan bahwa actualResponse tidak null
- Memastikan bahwa actualResponse sama dengan dummyRegisterResponse

- Memastikan bahwa method `authRepository.userRegister` telah dipanggil

2. Ketika Register Gagal

- Memastikan bahwa `Result.isSuccess` bernilai false
- Memastikan bahwa `Result.isFailure` bernilai true
- Memastikan bahwa nilai exception tidak null

SettingViewModel

1. Ketika Berhasil Menyimpan Token

- Memastikan bahwa method `authRepository.saveAuthToken` telah dipanggil

SplashViewModel

1. Ketika Berhasil Mendapatkan Token

- Memastikan bahwa `actualToken` tidak null
- Memastikan bahwa `actualToken` sama dengan `dummyToken`
- Memastikan bahwa method `authRepository.getAuthToken` telah dipanggil

2. Ketika Berhasil Mendapatkan Token Kosong

- Memastikan bahwa `actualToken` bernilai null
- Memastikan bahwa method `authRepository.getAuthToken` telah dipanggil

AuthRepositoryTest

1. Ketika User Berhasil Login

- Memastikan bahwa `Result.isSuccess` bernilai true
- Memastikan bahwa `Result.isFailure` bernilai false
- Memastikan bahwa `actualResponse` tidak null
- Memastikan bahwa `actualResponse` sama dengan `expectedResponse`
- Memastikan tidak ada exception pada `Result`

2. Ketika Gagal Login

- Memastikan bahwa `Result.isSuccess` bernilai false
- Memastikan bahwa `Result.isFailure` bernilai true
- Memastikan bahwa exception tidak null

3. Ketika Berhasil Register

- Memastikan bahwa Result.isSuccess bernilai true
- Memastikan bahwa Result.isFailure bernilai false
- Memastikan bahwa actualResponse tidak null
- Memastikan bahwa actualResponse sama dengan expectedResponse
- Memastikan tidak ada exception pada Result

4. Ketika Gagal Register

- Memastikan bahwa Result.isSuccess bernilai false
- Memastikan bahwa Result.isFailure bernilai true
- Memastikan bahwa exception tidak null

5. Ketika Berhasil Menyimpan Token

- Memastikan bahwa method preferencesDataSource.saveAuthToken telah dipanggil

6. Ketika Berhasil Mendapatkan Token

- Memastikan bahwa actualToken tidak null
- Memastikan bahwa actualToken sama dengan expectedToken
- Memastikan bahwa method preferencesDataSource.getAuthToken telah dipanggil

StoryRepositoryTest

1. Berhasil Mendapatkan Data Story

- Memastikan bahwa nilai data story tidak null
- Memastikan ukuran data asli dengan data dummy sama

2. Berhasil Mendapatkan Data Story Dengan Lokasi

- Memastikan bahwa Result.isSuccess bernilai true
- Memastikan bahwa Result.isFailure bernilai false
- Memastikan bahwa actualResponse tidak null
- Memastikan bahwa actualResponse sama dengan dummyStoriesResponse

3. Gagal Mendapatkan Data Story Dengan Lokasi

- Memastikan bahwa Result.isSuccess bernilai false
- Memastikan bahwa Result.isFailure bernilai true
- Memastikan bahwa nilai exception tidak null

4. Berhasil Mengunggah File

- Memastikan bahwa Result.isSuccess bernilai true
- Memastikan bahwa Result.isFailure bernilai false
- Memastikan bahwa actualResponse tidak null
- Memastikan bahwa actualResponse sama dengan expectedResponse
- Memastikan bahwa method apiService.uploadImage() telah dipanggil

5. Gagal Mengunggah File

- Memastikan bahwa Result.isSuccess bernilai false
- Memastikan bahwa Result.isFailure bernilai true
- Memastikan bahwa nilai exception tidak null
- Memastikan bahwa method apiService.uploadImage() telah dipanggil

@MediumTest

HomeFragmentTest

1. Berhasil Membuka Halaman Home Fragment

- Memastikan bahwa toolbar tampil di layar
- Memastikan bahwa recyclerview tampil di layar
- Memastikan bahwa terdapat teks “Dimas” pada layar

2. Berhasil Membuka Halaman Home Fragment dengan Data Kosong

- Memastikan pesan error berupa ImageView tampil di layar
- Memastikan pesan error berupa TextView tampil di layar

3. Berhasil Membuka Halaman Home Fragment Ketika Terjadi Error

- Memastikan pesan error berupa ImageView tampil di layar
- Memastikan pesan error berupa TextView tampil di layar

LocationFragmentTest

1. Berhasil Membuka Halaman Location Fragment

- Memastikan bahwa toolbar tampil di layar
- Memastikan bahwa mapview tampil di layar

@LargeTest

MainActivityTest

Diasumsikan bahwa pengguna telah login

1. Melihat Detail Cerita

- Memastikan MainActivity telah terbuka
- Memastikan RecyclerView telah tampil pada layar
- Melakukan aksi klik pada item pertama di RecyclerView
- Memastikan bahwa DetailStoryActivity telah terbuka
- Memastikan bahwa terdapat sebuah extra yang dibawa dari MainActivity
- Memastikan bahwa toolbar telah tampil
- Memastikan nama pengirim telah tampil
- Memastikan tanggal cerita telah tampil
- Memastikan gambar cerita telah tampil
- Melakukan aksi scroll ke bawah
- Memastikan deskripsi cerita telah tampil
- Kembali ke MainActivity