

Unit Test

AuthRepositoryTest

1. Ketika Login Sukses
 - a. Memastikan fungsi userLogin() pada AuthRepository mengembalikan nilai Result.Success dan membawa nilai response yang sesuai
 - b. Memastikan fungsi userLogin() terpanggil dari AuthRepository
 - c. Memastikan bahwa actualLogin tidak null
 - d. Memastikan bahwa Result.Success bernilai true
 - e. Memastikan data dummyLoginResponse dan data actualLogin sama.
2. Ketika Login Loading
 - a. Memastikan fungsi userLogin() pada AuthRepository mengembalikan nilai Result.Loading
 - b. Memastikan fungsi userLogin() terpanggil dari AuthRepository
 - c. Memastikan bahwa actualLogin tidak null
 - d. Memastikan bahwa Result.Loading bernilai true
3. Ketika Login Error
 - a. Memastikan fungsi userLogin() pada AuthRepository mengembalikan nilai Result.Error dan membawa nilai String yang sesuai
 - b. Memastikan fungsi userLogin() terpanggil dari AuthRepository
 - c. Memastikan bahwa actualLogin tidak null
 - d. Memastikan bahwa Result.Error bernilai true
4. Ketika SignUp Sukses
 - a. Memastikan fungsi userSignUp() pada AuthRepository mengembalikan nilai Result.Success dan membawa nilai response yang sesuai
 - b. Memastikan fungsi userSignUp() terpanggil dari AuthRepository
 - c. Memastikan bahwa actualSignUp tidak null
 - d. Memastikan bahwa Result.Success bernilai true
 - e. Memastikan data dummySignUpResponse dan data actualSignUp sama.
5. Ketika SignUp Loading
 - a. Memastikan fungsi userSignUp() pada AuthRepository mengembalikan nilai Result.Loading
 - b. Memastikan fungsi userSignUp() terpanggil dari AuthRepository
 - c. Memastikan bahwa actualSignUp tidak null
 - d. Memastikan bahwa Result.Loading bernilai true
6. Ketika SignUp Error
 - a. Memastikan fungsi userSignUp() pada AuthRepository mengembalikan nilai Result.Error dan membawa nilai String yang sesuai
 - b. Memastikan fungsi userSignUp() terpanggil dari AuthRepository

- c. Memastikan bahwa actualSignUp tidak null
 - d. Memastikan bahwa Result.Error bernilai true
- 7. Ketika Save Token Berhasil
 - a. Memastikan bahwa fungsi saveToken() telah terpanggil.
- 8. Ketika Get Token Berhasil
 - a. Memastikan data tidak null
 - b. Memastikan data dummyToken sama dengan actualToken
 - c. Memastikan bahwa fungsi getToken() telah terpanggil

StoryRepositoryTest

1. Ketika Get Story Berhasil
 - a. Memastikan data tidak null
 - b. Memastikan jumlah data sesuai dengan yang diharapkan
2. Ketika Get Story with Location Berhasil
 - a. Memastikan fungsi getStoriesWithLocation() pada StoryRepository mengembalikan nilai Result.Success dan membawa nilai response yang sesuai
 - b. Memastikan fungsi getStoriesWithLocation() terpanggil dari StoryRepository
 - c. Memastikan bahwa actualGetStories tidak null
 - d. Memastikan bahwa Result.Success bernilai true
 - e. Memastikan data dummyGetStoriesResponse dan data actualGetStories sama.
3. Ketika Get Story with Location Loading
 - a. Memastikan fungsi getStoriesWithLocation() pada StoryRepository mengembalikan nilai Result.Loading
 - b. Memastikan fungsi getStoriesWithLocation() terpanggil dari StoryRepository
 - c. Memastikan bahwa actualGetStories tidak null
 - d. Memastikan bahwa Result.Loading bernilai true
4. Ketika Get Story with Location Error
 - a. Memastikan fungsi getStoriesWithLocation() pada StoryRepository mengembalikan nilai Result.Error dan membawa nilai String yang sesuai
 - b. Memastikan fungsi getStoriesWithLocation() terpanggil dari StoryRepository
 - c. Memastikan bahwa actualGetStories tidak null
 - d. Memastikan bahwa Result.Error bernilai true
5. Ketika Upload Image Berhasil
 - a. Memastikan fungsi uploadImage() pada StoryRepository mengembalikan nilai Result.Success dan membawa nilai response yang sesuai
 - b. Memastikan fungsi uploadImage() terpanggil dari StoryRepository
 - c. Memastikan bahwa actualUploadImage tidak null
 - d. Memastikan bahwa Result.Success bernilai true

- e. Memastikan data `dummyUploadImageResponse` dan data `actualUploadImage` sama.
- 6. Ketika Upload Image Loading
 - a. Memastikan fungsi `uploadImage()` pada `StoryRepository` mengembalikan nilai `Result.Loading`
 - b. Memastikan fungsi `uploadImage()` terpanggil dari `StoryRepository`
 - c. Memastikan bahwa `actualUploadImage` tidak null
 - d. Memastikan bahwa `Result.Loading` bernilai true
- 7. Ketika Upload Image Error
 - a. Memastikan fungsi `uploadImage()` pada `StoryRepository` mengembalikan nilai `Result.Error` dan membawa nilai `String` yang sesuai
 - b. Memastikan fungsi `uploadImage()` terpanggil dari `StoryRepository`
 - c. Memastikan bahwa `actualUploadImage` tidak null
 - d. Memastikan bahwa `Result.Error` bernilai true

AddStoryViewModelTest

- 1. Ketika Get Token Berhasil
 - a. Memastikan fungsi `getToken()` pada `addStoryViewModel` mengembalikan nilai sesuai `expectedGetToken`
 - b. Memastikan data tidak null
 - c. Memastikan data `dummyToken` sama dengan `actualToken`
 - d. Memastikan bahwa fungsi `getToken()` telah terpanggil
- 2. Ketika AddStory Berhasil
 - a. Memastikan fungsi `addStory()` pada `addStoryViewModel` mengembalikan nilai `Result.Success` dan membawa nilai response yang sesuai
 - b. Memastikan fungsi `addStory()` terpanggil dari `addStoryViewModel`
 - c. Memastikan bahwa `actualAddStory` tidak null
 - d. Memastikan bahwa `Result.Success` bernilai true
 - e. Memastikan data `dummyAddStoryResponse` dan data `actualAddStory` sama.
- 3. Ketika AddStory Loading
 - a. Memastikan fungsi `addStory()` pada `addStoryViewModel` mengembalikan nilai `Result.Loading`
 - b. Memastikan fungsi `addStory()` terpanggil dari `addStoryViewModel`
 - c. Memastikan bahwa `actualAddStory` tidak null
 - d. Memastikan bahwa `Result.Loading` bernilai true
- 4. Ketika AddStory Error
 - a. Memastikan fungsi `addStory()` pada `addStoryViewModel` mengembalikan nilai `Result.Error` dan membawa nilai `String` yang sesuai
 - b. Memastikan fungsi `addStory()` terpanggil dari `addStoryViewModel`

- c. Memastikan bahwa actualAddStory tidak nulli.
- d. Memastikan bahwa Result.Error bernilai true

ListStoryViewModelTest

- 1. Ketika Get Story Berhasil
 - a. Memastikan data tidak null
 - b. Memastikan jumlah data sesuai dengan yang diharapkan
- 2. Ketika resetUserData Berhasil
 - a. Memastikan fungsi resetUserData() berhasil dipanggil

LoginViewModelTest

- 1. Ketika Save Token Berhasil
 - a. Memastikan bahwa fungsi saveToken() telah terpanggil.
- 2. Ketika Login Sukses
 - a. Memastikan fungsi userLogin() pada LoginViewModel mengembalikan nilai Result.Success dan membawa nilai response yang sesuai
 - b. Memastikan fungsi userLogin() terpanggil dari LoginViewModel
 - c. Memastikan bahwa actualLogin tidak null
 - d. Memastikan bahwa Result.Success bernilai true
 - e. Memastikan data dummyLoginResponse dan data actualLogin sama.
- 3. Ketika Login Loading
 - a. Memastikan fungsi userLogin() pada LoginViewModel mengembalikan nilai Result.Loading
 - b. Memastikan fungsi userLogin() terpanggil dari LoginViewModel
 - c. Memastikan bahwa actualLogin tidak null
 - d. Memastikan bahwa Result.Loading bernilai true
- 4. Ketika Login Error
 - a. Memastikan fungsi userLogin() pada LoginViewModel mengembalikan nilai Result.Error dan membawa nilai String yang sesuai
 - b. Memastikan fungsi userLogin() terpanggil dari LoginViewModel
 - c. Memastikan bahwa actualLogin tidak null
 - d. Memastikan bahwa Result.Error bernilai true

MapsStoryViewModelTest

1. Ketika Get Story with Location Berhasil
 - a. Memastikan fungsi `getStoriesWithLocation()` pada `StoryRepository` mengembalikan nilai `Result.Success` dan membawa nilai response yang sesuai
 - b. Memastikan fungsi `getStoriesWithLocation()` terpanggil dari `StoryRepository`
 - c. Memastikan bahwa `actualGetStories` tidak null
 - d. Memastikan bahwa `Result.Success` bernilai true
 - e. Memastikan data `dummyGetStoriesResponse` dan data `actualGetStories` sama.
2. Ketika Get Story with Location Loading
 - a. Memastikan fungsi `getStoriesWithLocation()` pada `StoryRepository` mengembalikan nilai `Result.Loading`
 - b. Memastikan fungsi `getStoriesWithLocation()` terpanggil dari `StoryRepository`
 - c. Memastikan bahwa `actualGetStories` tidak null
 - d. Memastikan bahwa `Result.Loading` bernilai true
3. Ketika Get Story with Location Error
 - a. Memastikan fungsi `getStoriesWithLocation()` pada `StoryRepository` mengembalikan nilai `Result.Error` dan membawa nilai String yang sesuai
 - b. Memastikan fungsi `getStoriesWithLocation()` terpanggil dari `StoryRepository`
 - c. Memastikan bahwa `actualGetStories` tidak null
 - d. Memastikan bahwa `Result.Error` bernilai true

SignUpViewModelTest

1. Ketika SignUp Sukses
 - a. Memastikan fungsi `userSignUp()` pada `SignUpViewModel` mengembalikan nilai `Result.Success` dan membawa nilai response yang sesuai
 - b. Memastikan fungsi `userSignUp()` terpanggil dari `SignUpViewModel`
 - c. Memastikan bahwa `actualSignUp` tidak null
 - d. Memastikan bahwa `Result.Success` bernilai true
 - e. Memastikan data `dummySignUpResponse` dan data `actualSignUp` sama.
2. Ketika SignUp Loading
 - a. Memastikan fungsi `userSignUp()` pada `SignUpViewModel` mengembalikan nilai `Result.Loading`
 - b. Memastikan fungsi `userSignUp()` terpanggil dari `SignUpViewModel`
 - c. Memastikan bahwa `actualSignUp` tidak null
 - d. Memastikan bahwa `Result.Loading` bernilai true
3. Ketika SignUp Error
 - a. Memastikan fungsi `userSignUp()` pada `SignUpViewModel` mengembalikan nilai `Result.Error` dan membawa nilai String yang sesuai
 - b. Memastikan fungsi `userSignUp()` terpanggil dari `SignUpViewModel`
 - c. Memastikan bahwa `actualSignUp` tidak null

- d. Memastikan bahwa Result.Error bernilai true

SignUpViewModelTest

1. Ketika Get Token Berhasil
 - a. Memastikan fungsi getToken() pada SplashScreenViewModel mengembalikan nilai sesuai dengan expectedGetToken
 - b. Memastikan bahwa fungsi getToken() telah terpanggil
 - c. Memastikan data tidak null
 - d. Memastikan data dummyToken sama dengan actualToken

Integration Test

@MediumTest

ListStoryActivityMediumTest

1. Mejalankan ListStoryActivity
 - a. Memastikan topAppBar(AppBar) tampil dilayar.
 - b. Memastikan rv_story(RecyclerView) tampil dilayar.
 - c. Memastikan rv_story dapat melakukan scrolling

MapsActivityMediumTest

1. Mejalankan MapsActivity
 - a. Memastikan map(Google Map) tampil dilayar

@LargeTest

ListStoryActivityLargeTest

1. Mejalankan ListStoryActivity
 - a. Memastikan topAppBar(AppBar) tampil dilayar.
 - b. Memastikan fab_maps(FloatingActionButton) tampil di layar
 - c. Memastikan rv_story(RecyclerView) tampil dilayar.
 - d. Memastikan rv_story dapat melakukan scrolling
2. Menjalankan DetailStoryActivity
 - a. Memastikan saat salah satu item pada rv_story di klik, maka akan berpindah ke halaman DetailStoryActivity

- b. Memastikan topAppBar(AppBar) tampil dilayar.
- c. Memastikan profile_image(ImageView) tampil dilayar.
- d. Memastikan tv_detail_name(TextView) tampil dilayar.
- e. Memastikan tv_detail_photo(TextView) tampil dilayar.
- f. Memastikan tv_detail_description(TextView) tampil dilayar.

3. Menjalakan MapsActivity

- a. Memastikan fab_maps(FloatingActionButton) pada saat di klik berpindah ke MapsActivity
- b. Memastikan map(GoogleMaps) tampil dilayar

4. Menjalankan AddStoryActivity

- a. Memastikan addStory pada saat di klik berpindah ke AddStoryActivity
- b. Memastikan topAppBar(AppBar) tampil dilayar.
- c. Memastikan previewImageView(ImageView) tampil dilayar.
- d. Memastikan ed_add_description(EditText) tampil dilayar.
- e. Memastikan sw_location(SwitchCompat) tampil dilayar.
- f. Memastikan btn_gallery(Button) tampil dilayar.
- g. Memastikan btn_camera(Button) tampil dilayar.
- h. Memastikan button_add (Button) tampil dilayar.