Sean Stott and Katleho Makhooane

2017273315,  2020102527

GPS APP REPort

Contents

[Singleton Pattern vs. ViewModel Injection: 1](#_Toc165823849)

[GoogleMap as Composable: 1](#_Toc165823850)

[Geolocation: 1](#_Toc165823851)

[FusedLocationProviderClient: 1](#_Toc165823852)

The report outlines the specific implementations that differ from Shad Sluiter's approach:

# Singleton Pattern vs. ViewModel Injection:

Sluiter uses the Singleton pattern to contain a global list of store locations. This pattern ensures that only one instance of the list is created throughout the application's lifecycle, providing global access to the list.

In our implementation, we injected the LocationViewModel into our activity class and passed it to different composable functions. This approach follows the MVVM architecture recommended when building Apps with Jetpack Compose, where view-related data and logic are encapsulated within a ViewModel class.

# GoogleMap as Composable:

Sluiter uses a fragment to display the GoogleMap, whereas we use a composable to display GoogleMap.

Geolocation:The method used to get the address of the current location has been deprecated in SDK 33, and a new method that takes a GeolocationListener as an extra parameter has been included, so this request is non-blocking.

# FusedLocationProviderClient:

The method used to request continuous location updates are different. Sluiter used a function that took LocationCallback, whereas we used a method that took a LocationListener.