LitterBug Report

Litter has a negative impact on the environment and contributes heavily to water and soil pollution. As litter degrades, it releases unnatural chemicals and microparticles into the environment. While awareness about litter and the environment has become more prevalent and it seems that more people care about the impact humans are having on the environment, I've noticed that the amount of litter in my community hasn't changed and that people are still unwilling to pick up litter. I spend a lot of time going on walks in my town and neighboring towns. I've noticed that no matter where I'm walking, in a neighborhood, on a busy street, and even in the woods, there is always litter that needs to be picked up.

A lot of people care about the environment and the health and well being of their communities, but are unmotivated to or have never picked up litter. I wanted to create an app that had a positive impact on the environment and that motivated people to help the environment through trash pickup.The LitterBug app is for people who want to make their communities more beautiful and safe. My goal was to create an app that allows users to be able to track their litter pickup on a map and see their impact. I also wanted to facilitate the users' cleanup by allowing them to see litter pinned by other users that needs to be picked up. This feature would also allow people that can't pick up litter to still have a positive impact on the environment and to help others more easily find litter that needs to be picked up.

I began my app creation process by designing three tabs for interfacing with the user: the Home tab, the Impact tab and the Litter Map tab. After completing the design layout of the tabs I created the bottom navigation bar that includes the three tabs. The home tab is where the user can see a summary of the amount of litter they've picked up. The user can view a map with the pinned locations of all the litter they've picked up on the Impact tab. The Litter Map tab displays a map with pins showing locations where litter needs to be picked up. On this tab the user can push a button to indicate that litter has been found at their current location; they can also select a pin and indicate that they have picked up litter. Once the litter is picked up it is added to the user's Impact map.

The Impact and Litter Map tabs are the most important and crucial components of my app. To create these so that users can map litter and see their mapped litter I used the Maps SDK for Android. To begin creating my maps I first had to obtain a Google Maps API key from the Google API console. Once I obtained my API key I could then create my map. To do this I used my Bottom navigation fragments that I created to act as containers for my two maps. I then got a handle to the map fragments by calling FragmentManager.findFragmentById(). Then I used getMapAsync() to register for the map callbacks. Next I Implemented the OnMapReadyCallback interface and overrode the onMapReady() method, to set up the map when the GoogleMap object is available.

The next step in the creation of my app is to get the user's current location when they want to pin a piece of litter so that it can automatically pin the litter without the user having to input their location themselves. To get the users current location I first needed to request permission to use their location in order to preserve privacy. To do this I used RequestPermission, included with AndroidX , that allows the system to manage the permission request code for me. When user wants to pin a location where litter has been found they will press a floating action button located on the Google Map fragment. When the user presses the button to tag the litter location, LitterBug will store the user's current location as long as the user gives location permission. I used FusedLocationProviderClient and LocationRequest Java classes to get the user's last known location.

In order to maintain user data, the litter location data and the user's litter pickup history I needed to incorporate some kind of backend database functionality. To do this I used FireBase as a Mobile Backend as a Service (MBaaS). Firestore Database contains collections of data which include documents. Within each document are data fields. For my LitterBug app I included a collection called litterLocations for my Litter Map tab which holds documents for each of the locations where litter is located. Each document contains the latitude and longitude of the litter location. Similarly, there is a pickupLocations collection for each Litterbug user that stores the latitude and longitude for each location where the user picked up litter.

The next step in my app creation process was to create a sign in page. With some research I found that firebase allows you to authenticate users with secure logins that track and identify every user who logs in and out of my app.