

## Screenshots From Unit Tests



10:18



Travel Gavel

# Sign In

Username

testuser

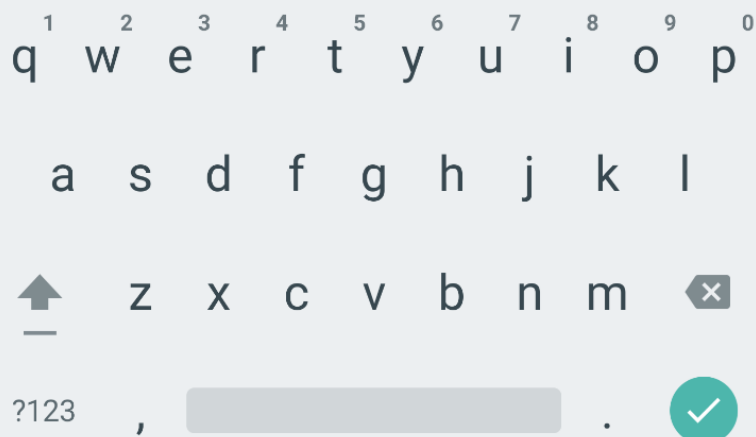
Please enter your password.

Password

|



SIGN IN





10:19



Travel Gavel

# Sign In

Username

testuser

Password

.....



**That doesn't match our records.**

SIGN IN



10:20

Travel Gavel

Create A New Account

Your Name

John Doe

New Username

johndoe

New Password

password

Both passwords don't match

Verify New Password

notthesamepassword

CREATE A NEW ACCOUNT

# Create A New Account

Your Name

John Doe

New Username

johndoe

New Password

password



Verify New Password

password



CREATE A NEW  
ACCOUNT



10:20



Travel Gavel

# Create A New Account

Please enter your name

Your Name

New Username

johndoe

New Password

password



Verify New Password

password



CREATE A NEW  
ACCOUNT



```
package com.example.travelgavelplaces;

import static org.junit.Assert.*;

import org.junit.Test;

public class MainActivityTest {

    @Test
    public void availability() {
        boolean activity = new MainActivity().isServicesOK();
        assertTrue(activity);
    }

    //TEST WHETHER GOOGLE PLAY SERVICES IS WORKING
    //PASSED TEST MEANS GOOGLE PLAY SERVICES IS OK| You, 2 minutes ag
    //FAILED TEST MEANS IT IS NOT WORKING AND SOMETHING IS WRONG
}
```

```

/*TESTS WHETHER LOCATION PERMISSION HAS BEEN GRANTED
 * IF IT HAS, DEVICE LOCATION IS GOTTEN AND LOCATION IS ACCESSED
 * ELSE, NOTHING HAPPENS*/   You, Moments ago • Uncommitted changes

if (mLocationPermissionGranted) {
    getDeviceLocation();
    if (ActivityCompat.checkSelfPermission( context: this, Manifest.permission.ACCESS_FINE_LOCATION)
        != PackageManager.PERMISSION_GRANTED && ActivityCompat.checkSelfPermission( context: this,
            Manifest.permission.ACCESS_COARSE_LOCATION) != PackageManager.PERMISSION_GRANTED) {
        return;
    }
    mMap.setMyLocationEnabled(true);
    mMap.getUiSettings().setMyLocationButtonEnabled(false);

    init();
}

```

```

/*TESTS IF THERE IS IOException
 * IF SO SEND MESSAGE TO LOG*/   You, 3 minutes ago • Uncommitted changes

try{
    list = geocoder.getFromLocationName(searchString,   maxResults: 1);
}catch (IOException e){
    Log.e(TAG,   msg: "geoLocate: IOException" + e.getMessage() );
}

```

```

/*TESTS IF CURRENT LOCATION IS FOUND
 * IF NOT, THROWS ERROR MESSAGE TO LOG
 * IF SO, THROWS SUCCESSFUL MESSAGE TO LOG AND MOVES CAMERA TO LATITUDE AND LONGITUDE
 * */
CasualGamer1221
@Override
public void onComplete(@NonNull Task task) {
    if (task.isSuccessful()){
        Log.d(TAG,   msg: "onComplete: found location!");
        Location currentLocation = (Location) task.getResult();

        moveCamera(new LatLng(currentLocation.getLatitude(), currentLocation.getLongitude()),
            DEFAULT_ZOOM,   CasualGamer1221, Yesterday • Initial implementation of places api, working search bar and map
            title: "My Location");
    }else {
        Log.d(TAG,   msg: "onComplete: current location is null");
        Toast.makeText( context: MapActivity.this,   text: "unable to get current location", Toast.LENGTH_SHORT).show();
    }
}
}

```



```
/*TESTS FOR USER PERMISSIONS
* CHECKS IF EITHER COARSE OR FINE LOCATION IS GIVEN
* IF SO, INITIALIZES MAP | You, 3 minutes ago • Uncommitted changes
* ELSE, REQUESTS THE PERMISSIONS FROM THE USER*/
if(ContextCompat.checkSelfPermission(this.getApplicationContext(),
    FINE_LOCATION) == PackageManager.PERMISSION_GRANTED){
    if(ContextCompat.checkSelfPermission(this.getApplicationContext(),
        COARSE_LOCATION) == PackageManager.PERMISSION_GRANTED){
        mLocationPermissionGranted = true;
        initMap();
    }else {
        ActivityCompat.requestPermissions( activity: this,
            permissions,
            LOCATION_PERMISSION_REQUEST_CODE);
    }
}else{
    ActivityCompat.requestPermissions( activity: this,
        permissions,
        LOCATION_PERMISSION_REQUEST_CODE);
}
```