IT 314 SOFTWARE ENGINEERING

LAB 8 : JUNIT TESTING

HEALTH CENTER MANAGEMENT

GROUP: 12

UNIT TESTING IN ANDROID STUDIO:

Here, we have taken the module **startups** to perform unit testing. The testing is being performed in Kotlin programming language as the project is being implemented in kotlin.

Functions that are to be tested:These are some functions which have been used to validate the user details.

1) isValidEmail

```
fun isValidMail(email: String): Boolean {
        val emailRegex = "^[a-zA-Z0-9_+&*-]+(?:\\." +
                "[a-zA-Z0-9_+&*-]+)*@" +
                "(?:[a-zA-Z0-9-]+\\.)+[a-z" +
                "A-Z]{2,7}$"

        val pat: Pattern = Pattern.compile(emailRegex)
        return if (email == null) false else
pat.matcher(email).matches()
    }
```

2) isValidMobile

```
fun isValidMobile(phone: String): Boolean {
        val p = Pattern.compile("^\\d{10}$")

        val m: Matcher = p.matcher(phone)

        return m.matches()
    }
```

3) convertTimeInMillisToString

```
 fun convertTimeInMillisToString(time: Long): String {
        val simpleDateFormat = SimpleDateFormat("hh:mm aa",
Locale.getDefault())
        val date = Date(time)
        return simpleDateFormat.format(date)
    }
```

4) convertDateInMillisToString

```
fun convertDateInMillisToString(time: Long): String {
        val simpleDateFormat = SimpleDateFormat("dd/MM/yyyy",
Locale.getDefault())
        val date = Date(time)
        return simpleDateFormat.format(date)
    }
```

Test Cases: All the test cases have been written in a single file as below

```
package com.mypackage.it314_health_center

import org.junit.Assert.*;
import org.junit.Before
import org.junit.Test
import
com.mypackage.it314_health_center.adapters.ValidationFunctions;
import com.mypackage.it314_health_center.startups.Login

class AdaptersTest {
   private lateinit var validationFunctions: ValidationFunctions;

   @Before
   fun setUp() {
      // initializing the variables
      validationFunctions= ValidationFunctions()
   }

   @Test
   fun test_valid_email()
   {
         val email1="gangarajbopparam@gmail.com"
         val email2="thisIsAnEmailId"
         val output1= Login.Companion.isValidMail(email1)
         val output2=Login.Companion.isValidMail(email2)
        assertEquals(true,output1)
        assertEquals(false,output2)
   }
```

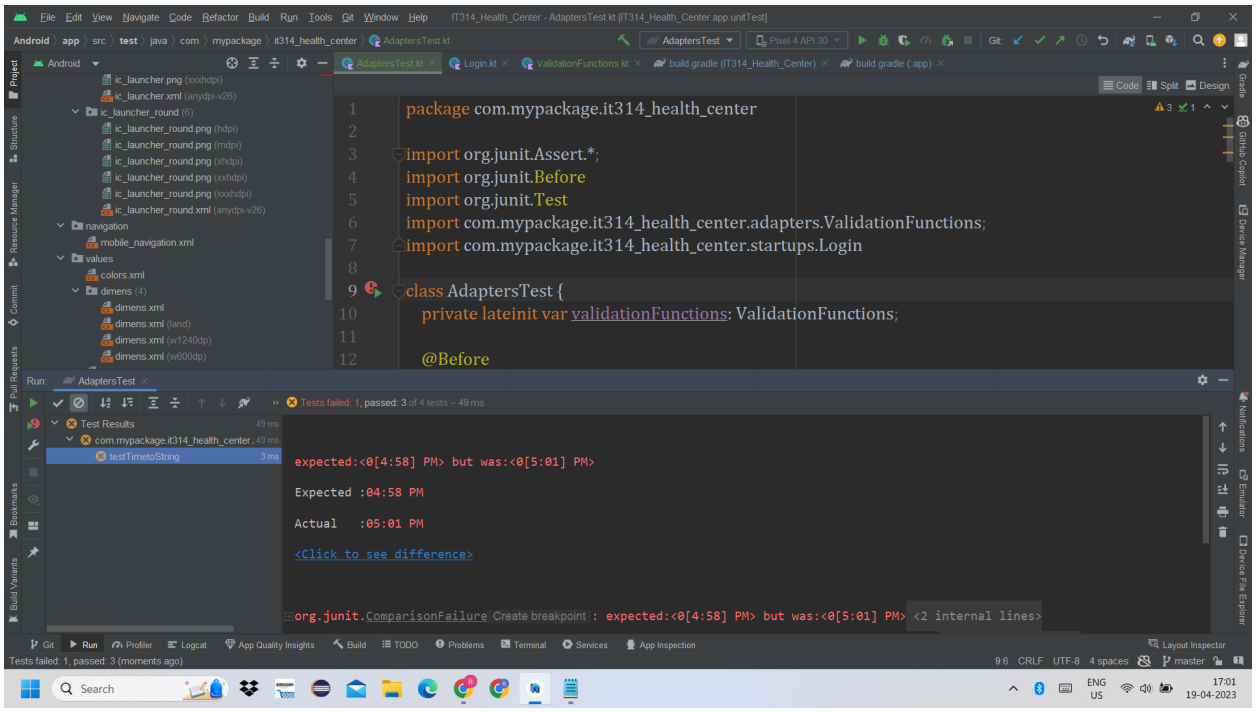
```
@Test
fun test_valid_mobile()
{
   val mobile1="6301231332"
   val mobile2="2342"
   val mobile3="23423rwe242"
   val output1= Login.Companion.isValidMobile(mobile1)
   val output2=Login.Companion.isValidMobile(mobile2)
   val output3=Login.Companion.isValidMobile(mobile3)
   assertEquals(true,output1)
   assertEquals(false,output2)
   assertEquals(false,output3)
}
@Test
fun testTimetoString()
{
       val t=System.currentTimeMillis()
       val expected ="04:58 PM" // current time
      val actual =validationFunctions.convertTimeInMillisToString(t)
       assertEquals(expected,actual)
}

@Test
fun testDateToString()
{
   val t=System.currentTimeMillis()
   val expected ="19/04/2023"
   val actual =validationFunctions.convertDateInMillisToString(t)
   assertEquals(expected,actual)
```

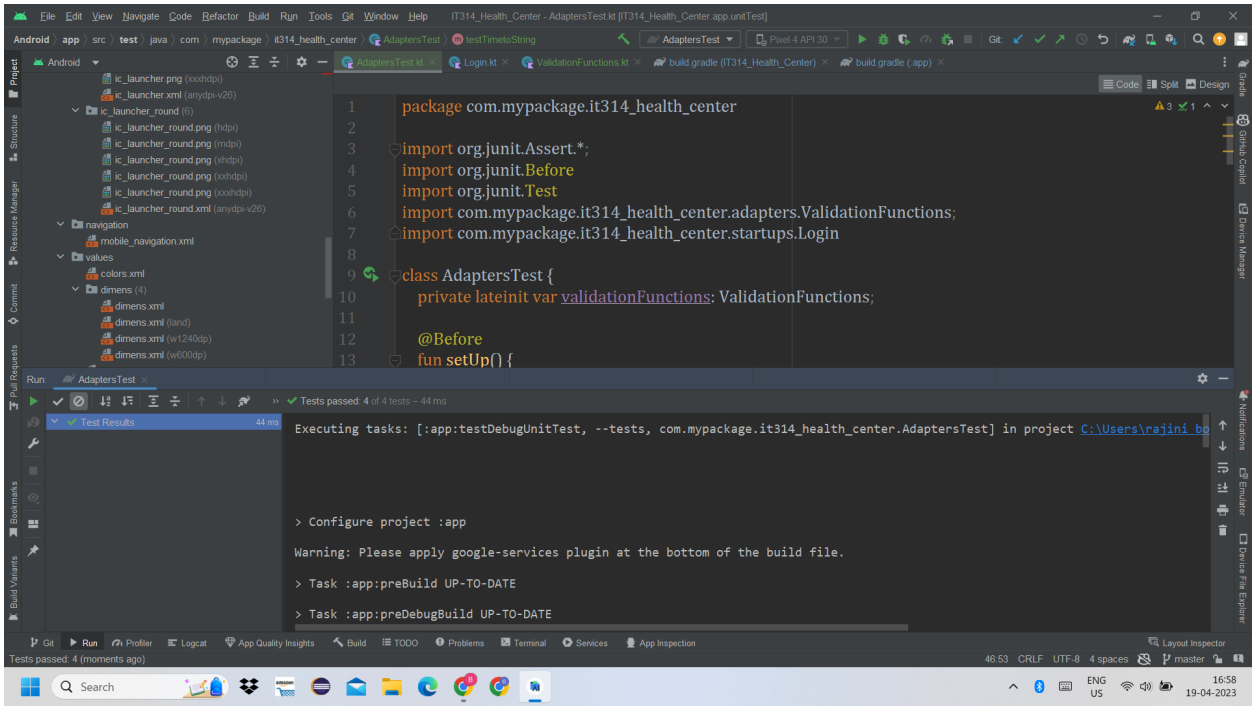
```
    }
}
```

Executing the test cases

Then we ran the above 4 test cases implemented:



For the first time, one of the test cases failed. We then checked source of the failure and noticed that there was a logical mistake.After correcting it, we ran the test cases again.



Now, it can be seen that all the test cases have been passed successfully.

Fixing the error:First we found the source of the error and changed the logical error in that function. Then all the test cases are passed.

The above test cases have been integrated with github.