Topics in Software Engineering

Assignment 1



Department of Computer Science

COMSATS University Islamabad, Lahore Campus

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| **Project Group Members** | | | | | | |
| Sr.# | Reg. # | Student Name | CGPA | Email ID | Phone # | Signature |
| (i) | FA19-BSE-076 | Hira Arshad | 2.97 | [FA19-BSE-076@cuilahore.edu.pk](mailto:FA19-BSE-076@cuilahore.edu.pk) | 0313-4389419 |  |
| (ii) | FA19-BSE-078 | Usman Muhammad Amin | 2.55 | [FA19-BSE-078@cuilahore.edu.pk](mailto:FA19-BSE-078@cuilahore.edu.pk) | 0309-4160520 |  |
| (iii) | FA19-BSE-102 | Syed Sarmad Moeen | 2.45 | [FA19-BSE-102@cuilahore.edu.pk](mailto:FA19-BSE-102@cuilahore.edu.pk) | 0334-4048456 |  |

**Part A**

1. The have used java and know we have many other languages like react native which is best for front end and for backend database we have mongodb etc. For app we can use react native nowadays they make our work a lot easier. Moreover They have used HTML,CSS,Boostrap,PHP,MySQL,PHP API library Slim for Web Development, but there is some latest languages for front-end is React that is more powerful than HTML,CSS and Boostrap. Node and Express framework are better than PHP.
2. Code has bottom navigation but documentation do not have this they have full code of bottom navigation but have commented it so they can use it later but in documentation they haven’t written about bottom navigation and write code for it. Code have Separate java file for the GPS tracker but didn’t mention properly in Documentation as a separate module.
3. In our opinion as we studied documentation and report many times there is no proper version detail, revision date and compatibility option. If any random person who will read a report of their fyp how they come to know about the version revision date and other technical aspect which I think some of them they have added but most and important one are missing like we have discussed above

**Part B**

1. **Blocker Violation**
   1. **FieldNamingConvention:**

**Code**

EditText Password;

**Message:**

The {0} name ''{1}'' doesn''t match ''{2}''

**Suggestion**

public class Foo {

Integer instanceField; // This is in camel case, so it's ok

Integer INSTANCE\_FIELD; // This will be reported unless you change the regex

}

* 1. **VariableNamingConventions**

**Code**

RequestBody id\_res = RequestBody.create(MediaType.parse("text/plain"), "1");

**Message**

{0} variable {1} should begin with {2}

**Suggestion**

public class Foo {

public static final Integer MY\_NUM = 0;

public String myTest = '';

DataModule dmTest = new DataModule();

}

1. **Show Warning Violation**
   1. **LawOfDemeter**

**Code**

Toast.makeText(IntroLocationActivity.**this**, "" + response.body().getMessage(), Toast.LENGTH\_SHORT).show();

**Message**

Potential violation of Law of Demeter

**Suggestion**

public class Foo {

/\*\*

\* This example will result in two violations.

\*/

public void example(Bar b) {

// this method call is ok, as b is a parameter of "example"

C c = b.getC();

// this method call is a violation, as we are using c, which we got from B.

// We should ask b directly instead, e.g. "b.doItOnC();"

c.doIt();

// this is also a violation, just expressed differently as a method chain without temporary variables.

b.getC().doIt();

// a constructor call, not a method call.

D d = new D();

// this method call is ok, because we have create the new instance of D locally.

d.doSomethingElse();

}

}

* 1. **AvoidDuplicateLiterals**

**Code**

RequestBody pass = RequestBody.create(MediaType.parse("text/plain"), Name.getText().toString());

**Message**

The String literal {0} appears {1} times in this file; the first occurrence is on line {2}

**Suggestion**

private void bar() {

buz("Howdy");

buz("Howdy");

buz("Howdy");

buz("Howdy");

}

private void buz(String x) {}

1. **Warning Violation**

**DataflowAnomalyAnalysis**

**Code**

Intent i = **new** Intent(IntroLocationActivity.**this**, MainActivity.**class**);

**Error Message**

Found ''{0}''-anomaly for variable ''{1}'' (lines ''{2}''-''{3}'').

**Suggestion**

public void foo() {

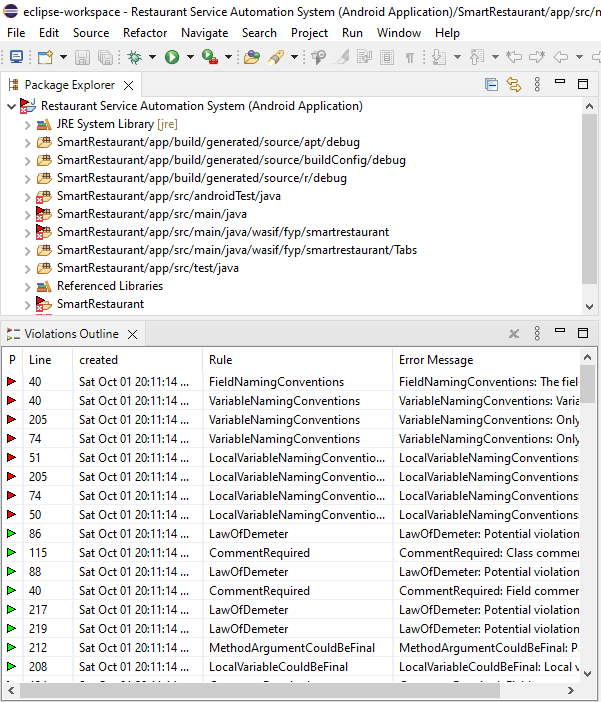
int buz = 5;

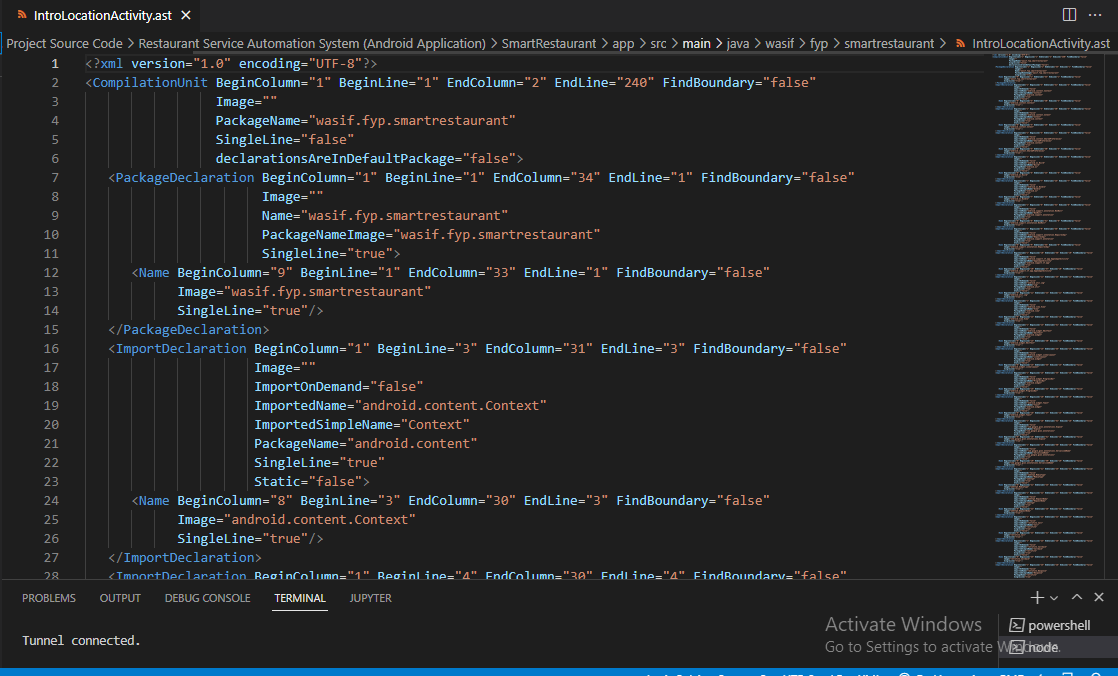
buz = 6; // redefinition of buz -> dd-anomaly

foo(buz);

buz = 2;

} // buz is undefined when leaving scope -> du-anomaly





**Part C**

* **Line having more than 100 character**

Snackbar.make(view, "Replace with your own action", Snackbar.LENGTH\_LONG).setAction("Action", null).show();

* **Child has incorrect indentation**

fragmentTransaction.commitAllowingStateLoss();

* **Local variable name must have id\_res must match**

RequestBody id\_res = RequestBody.create(MediaType.parse("text/plain"), "1");

