Nama : Asih Putri Retno Kedhaton

NIM : 21120122130068

Mata Kuliah : Rekayasa Perangkat Lunak Berbasis Komponen

Kelas : C

Running Example SOLID, Dependency Inversion Principle (DIP) pada java mengenai anggota keluarga. Bukti Screenshot:

• Tanpa DIP:

```
keluarga_NO_DIP - Apache NetBeans IDE 12.5
         <u>Edit View Navigate Source Refactor Run Debug Profile Team Tools Window Help</u>
      The second secon
                                                                           Start Page x 🚳 Main java x 🗹 Person java x 🗹 Research java x 🗗 Relationship java x 🗗 Relationships java x
                                                                                     Source History 🕼 🖫 - 📮 - 🔍 🔁 👺 🖺 📮 🔗 😓 🖭 🖭   🕕 📲 📑
                      Source Packages
                           - 🔠 <default package>
                                      ∆ Main.java
                             keluarga.models
                                                                                                        package keluarga_no_dip;
                                     Relationship.java
                                                                                                       import keluarga_no_dip.models.Person;
                                                                                                       import keluarga_no_dip.relationships.Relationships;
                                     Relationships.java
                                                                                                       import keluarga_no_dip.research.Research;
                       🌳 🔠 keluarga.research
                                                                                                        public class Main {
                P- Test Packages
                                                                                                           public static void main(String[] args) {
                                                                                      11
                                                                                       12
                                                                                                                             Person parent = new Person("Eko");
                €- 📑 Libraries
                        → 및 JDK 16 (Default)
                        keluarga_NO_DIP
                                                                                                                             Relationships relationships = new Relationships();
                                      keluarga_no_dip
Main.java
                                                                                                                               relationships.addParentAndChild(parent, childl);
                                                                                      18
19
%
21
                                                                                                                               relationships.addParentAndChild(parent, child2);
                                                                                                                               new Research(relationships); // Research langsung menggunakan Relationships
                                      keluarga_no_dip.resea
                         Eko has a child named Putra
           ید
```

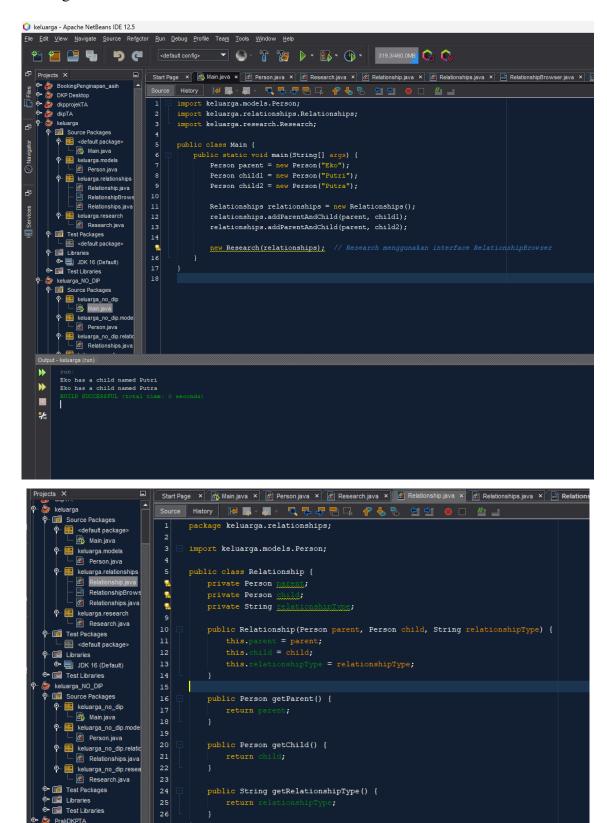
```
₽ Projects ×
                                   Start Page 🗴 🚳 Main. java 🗴 🙆 Person. java 🗴 🙆 Research. java 🗴 🙆 Relationship. java 🗴 🙆 Relationships. java
                                  Source History 🖟 📮 🔻 💆 🔁 🖫 🖫 🗸 😤 🖺 🖺 🖺 🖺
     b keluarga
                                          package keluarga_no_dip.relationships;
           <default package>
              Main.java
                                        import java.util.ArrayList;
import java.util.List;
           keluarga.models
              Person.java
                                        import keluarga_no_dip.models.Person;
🧷 Navigator
          keluarga.relationships
             Relationship.java
RelationshipBrows
Relationships.java
                                         public class Relationships {
                                    ₩.
                                              private List<Relationship> relations = new ArrayList<>();
         📳 keluarga.research
8
              Research.java
     P- m Test Packages
                                          private class Relationship {
           <default package>
                                    92
92
                                             private Person parent;
private Person child;
       Libraries

→ □ JDK 16 (Default)

     ⊕ 📠 Test Libraries
     keluarga_NO_DIP
                                                  public Relationship(Person parent, Person child) {
                                                      this.parent = parent;
this.child = child;
       - ma Source Packages
         Main.java
           keluarga_no_dip.mode
              Person.java
                                                   public Person getParent() {
         P- ■ keluarga_no_dip.relatio
□ ☑ Relationships.java
         P keluarga_no_dip.resea
              Research.java
     • mar Test Packages
                                                   public Person getChild() {
     • 🖼 Libraries
     PrakDKPTA
     b Struktur Data
        TA_RPL
      P 📠 Source Packages
                                               public void addParentAndChild(Person parent, Person child) {
         🗣 🔠 Ul_asset_login
                                                    relations.add(new Relationship(parent, child));
         φ- iii base
              ddd.java
              forgotpass.java
              login.java
              pencarian.java
                                              public List<Person> findAllChildrenOf(String name) {
                                              List<Person> children = new ArrayList<>();
              register.java
              report.java
reservasi.java
riwayat.java
                                    9<sub>A</sub>
                                                   for (Relationship relation : relations) {
                                                        if (relation.getParent().name.equals(name)) {
         ⊕- III ta_rpl
                                                             children.add(relation.getChild());
      • III Test Packages
        Libraries
      • Test Libraries
                                                    return children;
```

Tanpa menggunakan DIP, hanya ada 1 implementasi relationship pada modul. Bila implementasi Relationships diubah, seluruh modul yang bergantung pada kelas ini, seperti Research, harus diubah juga. Tidak ada pemisahan abstraksi dan implementasi pada modul sehingga mengurangi fleksibilitas.

• Dengan DIP



```
Projects X
                        Start Page × Main.java × M Person.java × M Research.java × M Relationship.ja
    keluarga
                                                           Source Packages
                                   package keluarga.relationships;
      <default package>
         Main.java
                                   import keluarga.models.Person;
                             4
                                   import java.util.List;
            Relationship.java
                             0
                                   public interface RelationshipBrowser {
                             0
                                       List<Person> findAllChildrenOf(String name);
           Relationships.java
         keluarga.research
                             9
         Research.java
```

Kode DIP ditambahkan abstraksi diwujudkan dengan dengan interface mendefinisikan RelationshipBrowser bahwa harus yang hanya ada metode findAllChildrenOf(String name) yang mengembalikan list anak-anak dari orang tua yang namanya diberikan. Namun, interface ini tidak peduli bagaimana detail implementasinya akan dilakukan. class relationship internal juga diberikan implementasi yg berbeda dari Relationships utama untuk memecah kebergantungan tiap implementasinya. Apapun yang diubah pada relationships tidak akan memengaruhi research karena sudah mendapatkan kelas abstrak.