Electrical circuit simulator

20194436 Nguyen Duy Hung

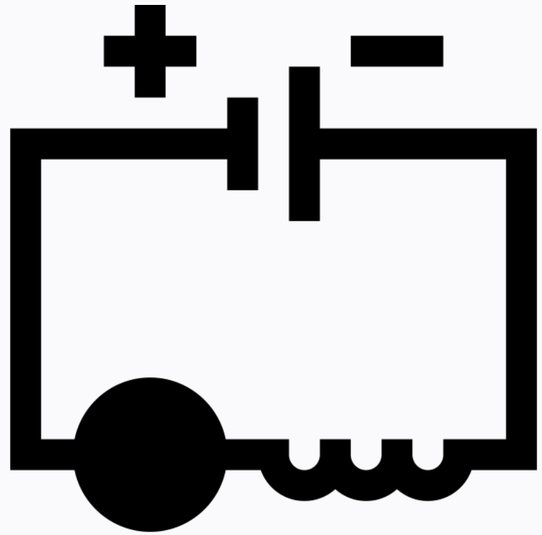
- Design GUI

20194448 Nguyen Hoang Nhat Quang

- Set initialization

20194449 Le Hai Son

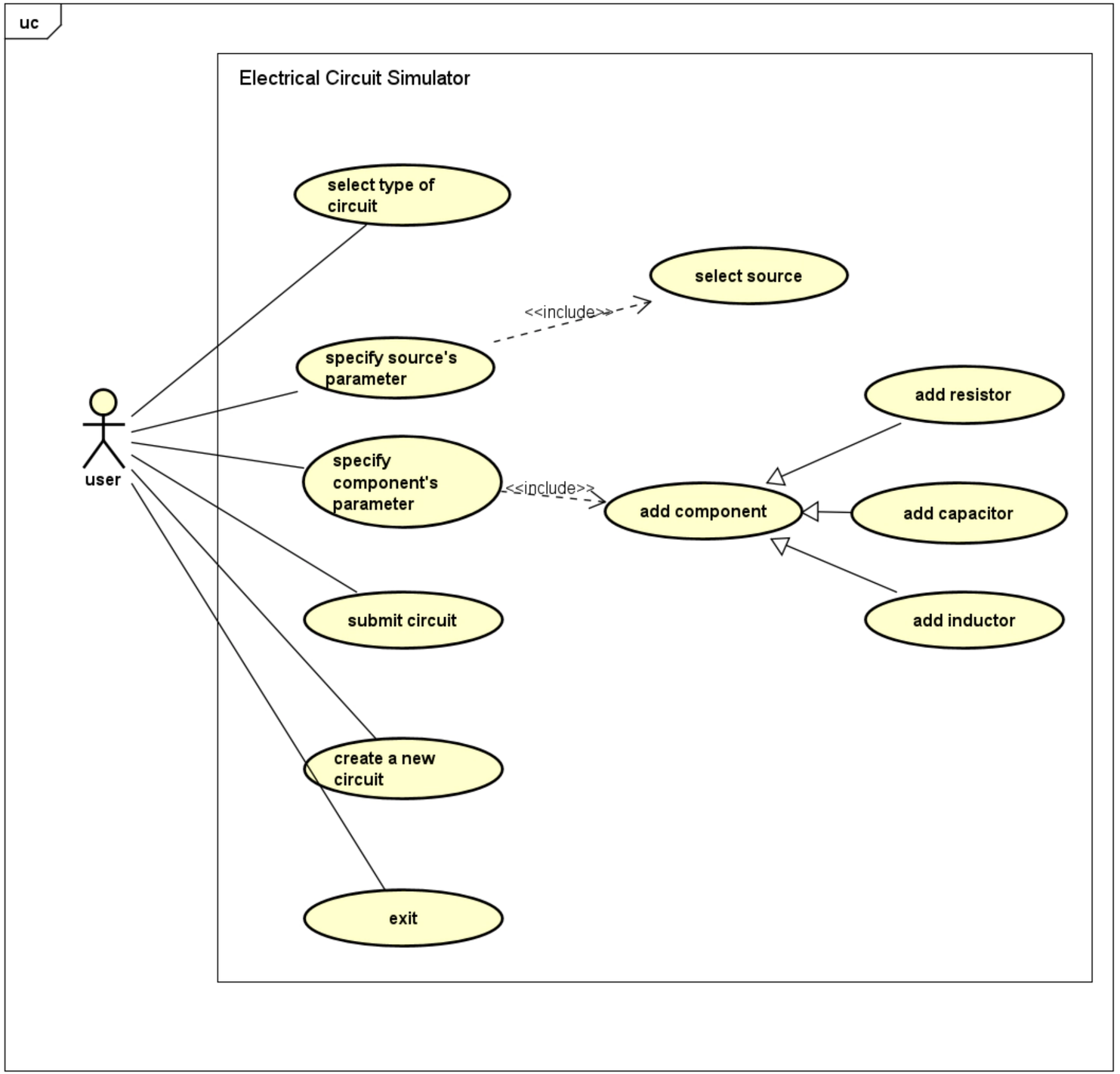
- Report and design diagram



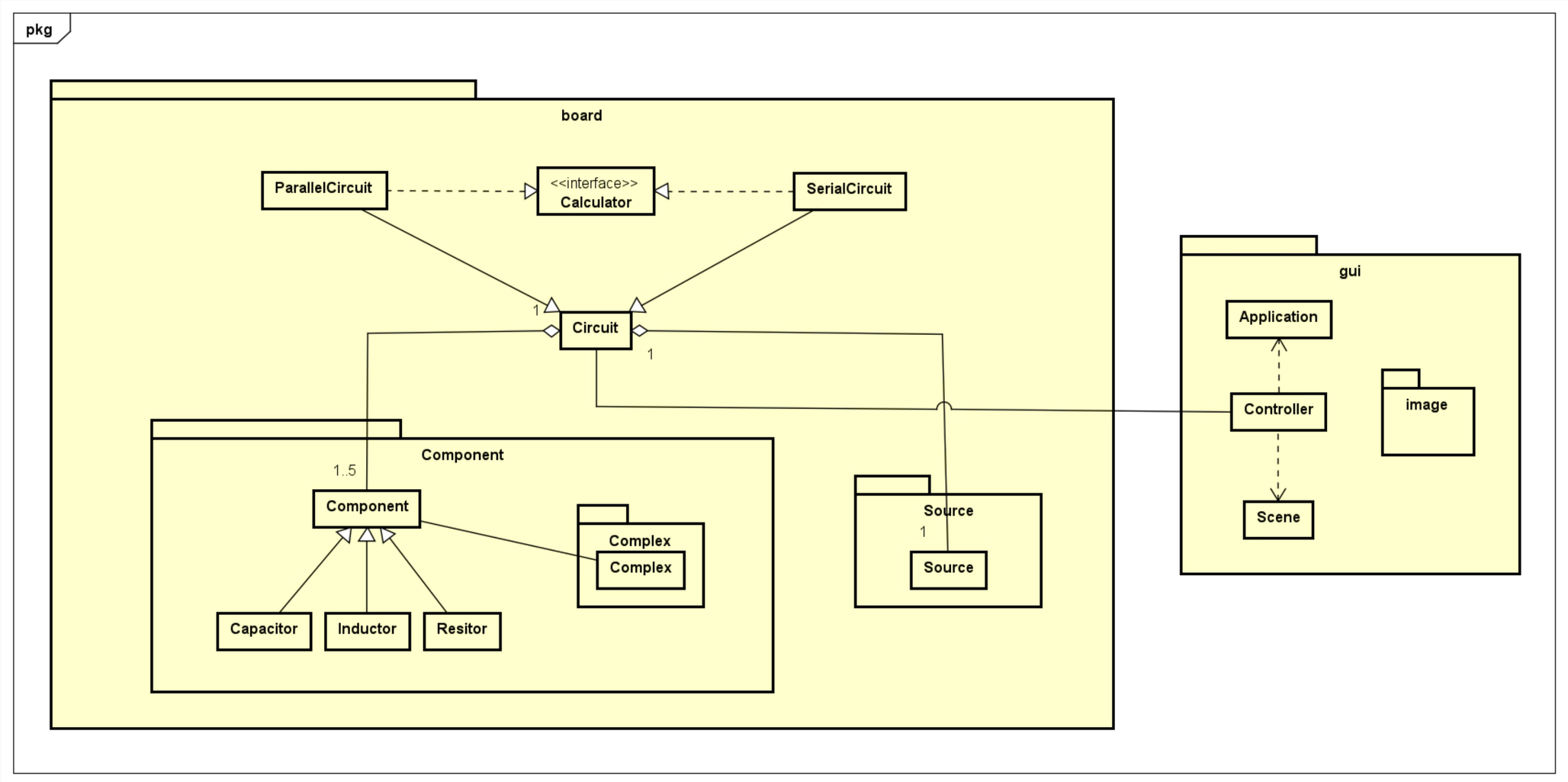
Problem statement

- Build an electrical circuit and perform circuit analysis on it
- Pick a type of circuit (parallel circuit and serial circuit)
- Select a type of source (AC and DC) and input parameter (V, Hz)
- Add component (resistor, inductor, capacitor) and input parameter (Ω, L, C)
- Press submit to view analysis sheet and circuit diagram
- Create a new circuit
- Exit

Use case diagram

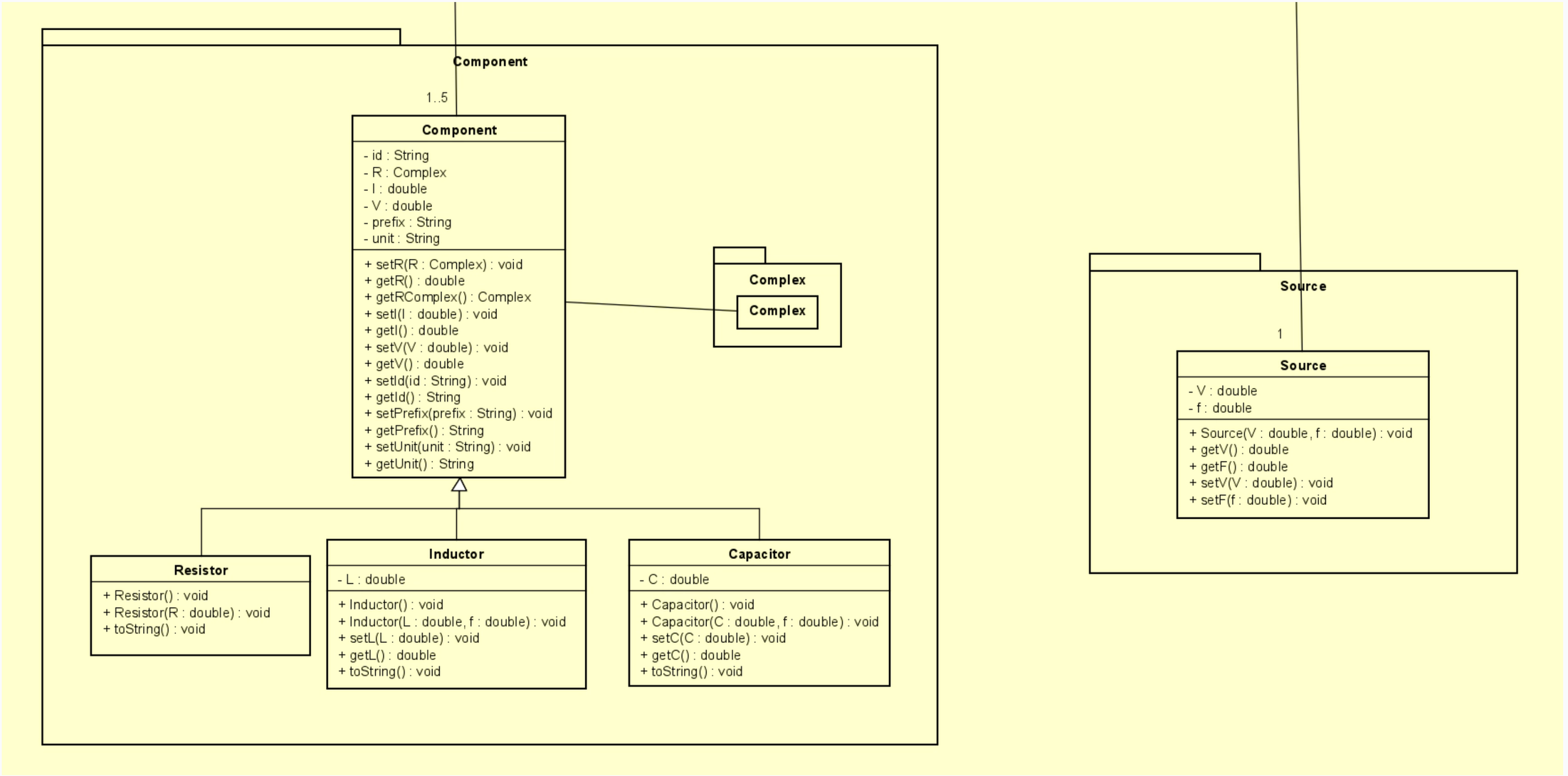


General class diagram



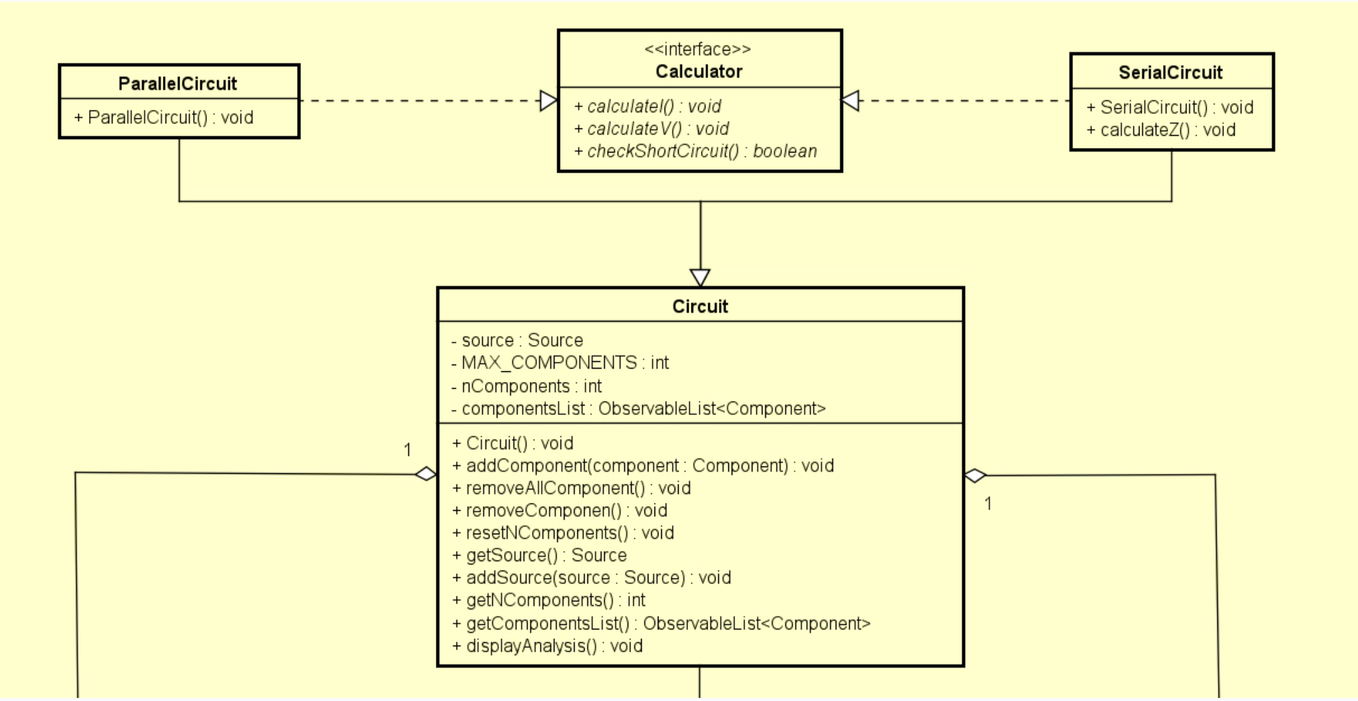
Class diagrams for packages

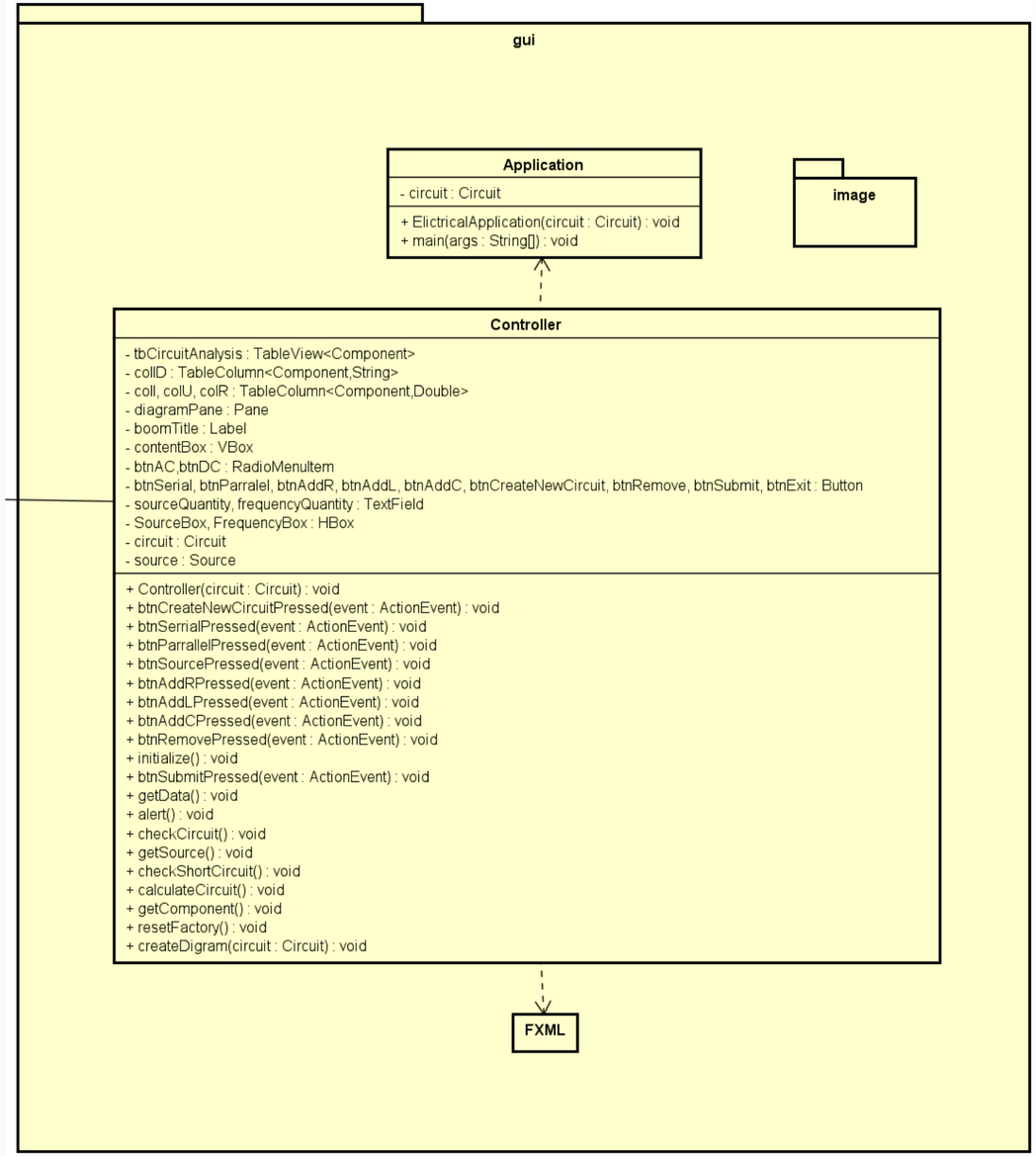
component and source



Class diagrams for packages

board and gui

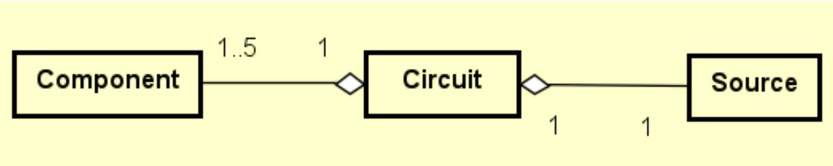




OOP Techniques

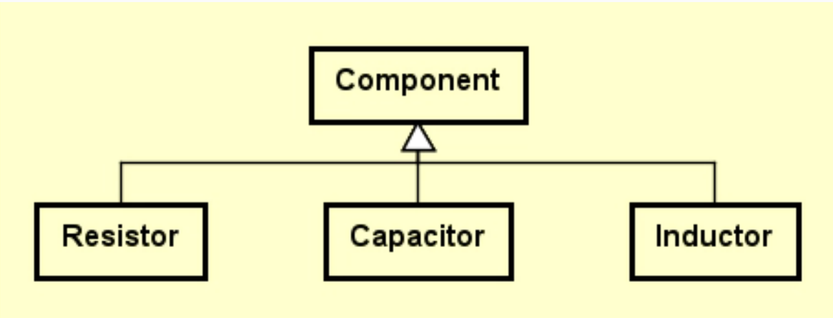
Aggregation

- A source *is a part of* a circuit board
- A component *is a part of* a circuit board

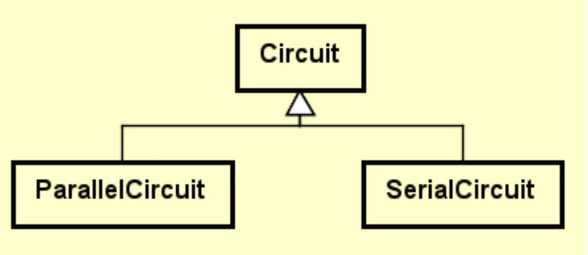


Inheritance

- A resistor *is a* component
- A capacitor *is a* component
- An inductor *is a* component



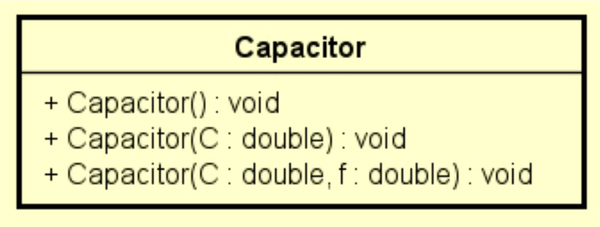
- A parallel circuit *is a* circuit
- A serial circuit *is a* circuit



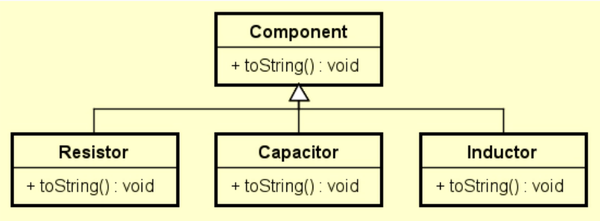
OOP Techniques

Polymorphism

- Static binding



- Dynamic binding




```
public void displayAnalysis() {
    for (Component component: componentsList) {
        System.out.println(component);
    }
}
```

Generics

```
private ObservableList<Component> componentsList = FXCollections.observableArrayList();
```

```
colID.setCellValueFactory(new
                PropertyValueFactory<Component, String>("id"));
colR.setCellValueFactory(new
                PropertyValueFactory<Component, Double>("R"));
colU.setCellValueFactory(new
                PropertyValueFactory<Component, Double>("V"));
colI.setCellValueFactory(new
                PropertyValueFactory<Component, Double>("I"));
```

Demo video

bitly.com/electrical-circuit-simulator-demo-video