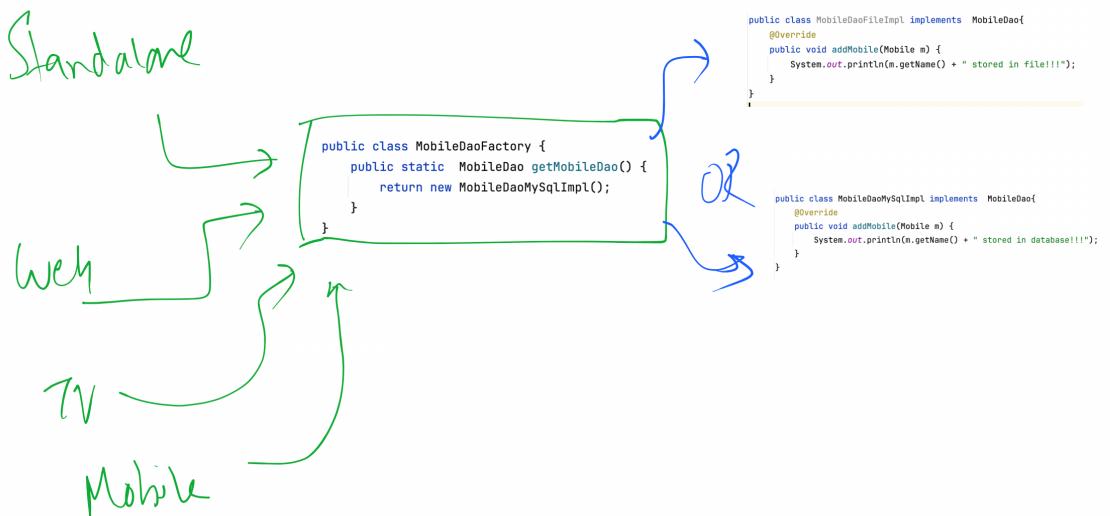
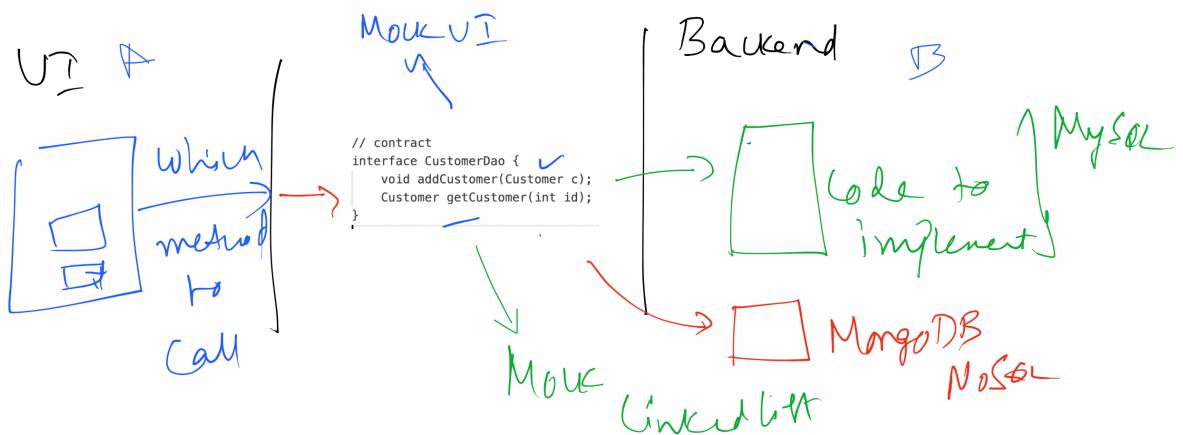
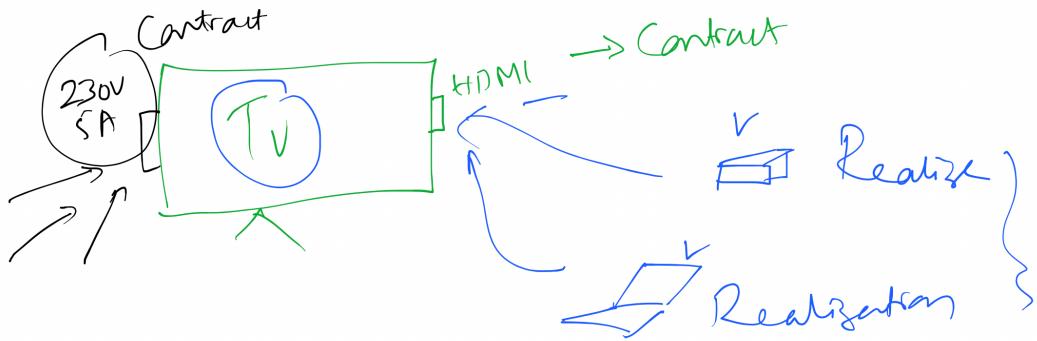
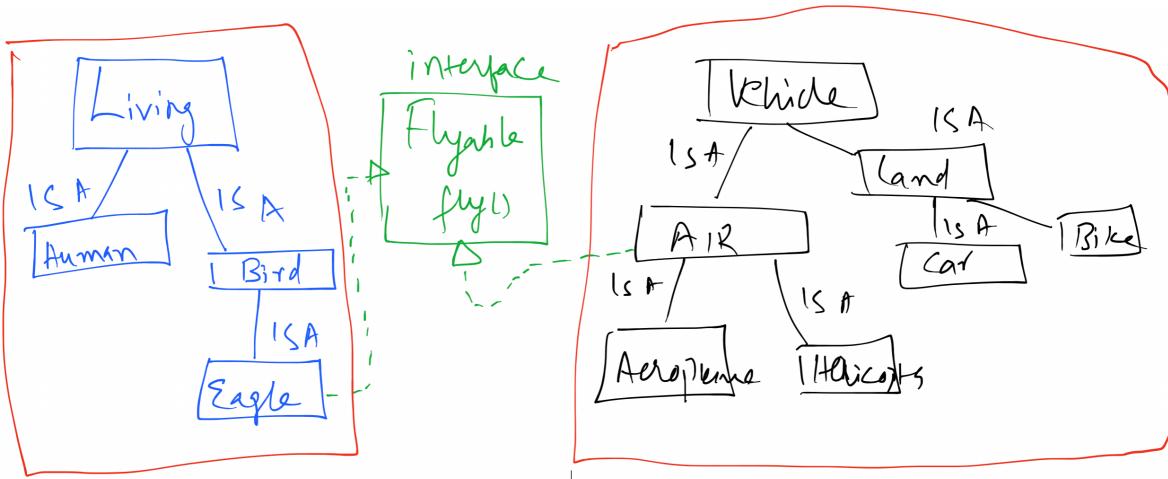


Realization relationship





```
interface Living {
    // ...
}
```

```
class Human {
    // ...
}
```

```
class Bird {
    // ...
}
```

```
class Eagle {
    // ...
}
```

interface Flyable

fly()

interface Klude

IS A

Land

IS A

Bike

AIR

IS A

Aerojenome

IS A

Helicots

```
interface Swimable {
    swim();
}
```

```
class Actor implements Danceable {
    // state and behaviour
    dance() {
        ...
    }
}
```

```
// all actors are capable to dance
class Hero extends Actor implements Swimable, Fightable {
    // state and behaviour
    fight() {
        ...
    }
    swim() {
        ...
    }
}
```

```
// hero object is created, referenced by Fight interface
// just like Product p = new Mobile();
Fightable f = new Hero();
f.fight(); // valid
f.swim(); // Not valid; Stunt Master can direct only fight sequence of Hero
```

```
Swimable s = (Swimable) f;
s.swim(); // makes hero swim --> Swim Instructor
```

```
Danceble d = (Danceable) s;
d.dance(); // Choreographer is directing dance sequence of Hero
```