What to test?

Levels of tests

- Like levels of requirements
- Lower level → Easier
- Higher level → More confidence

Functions

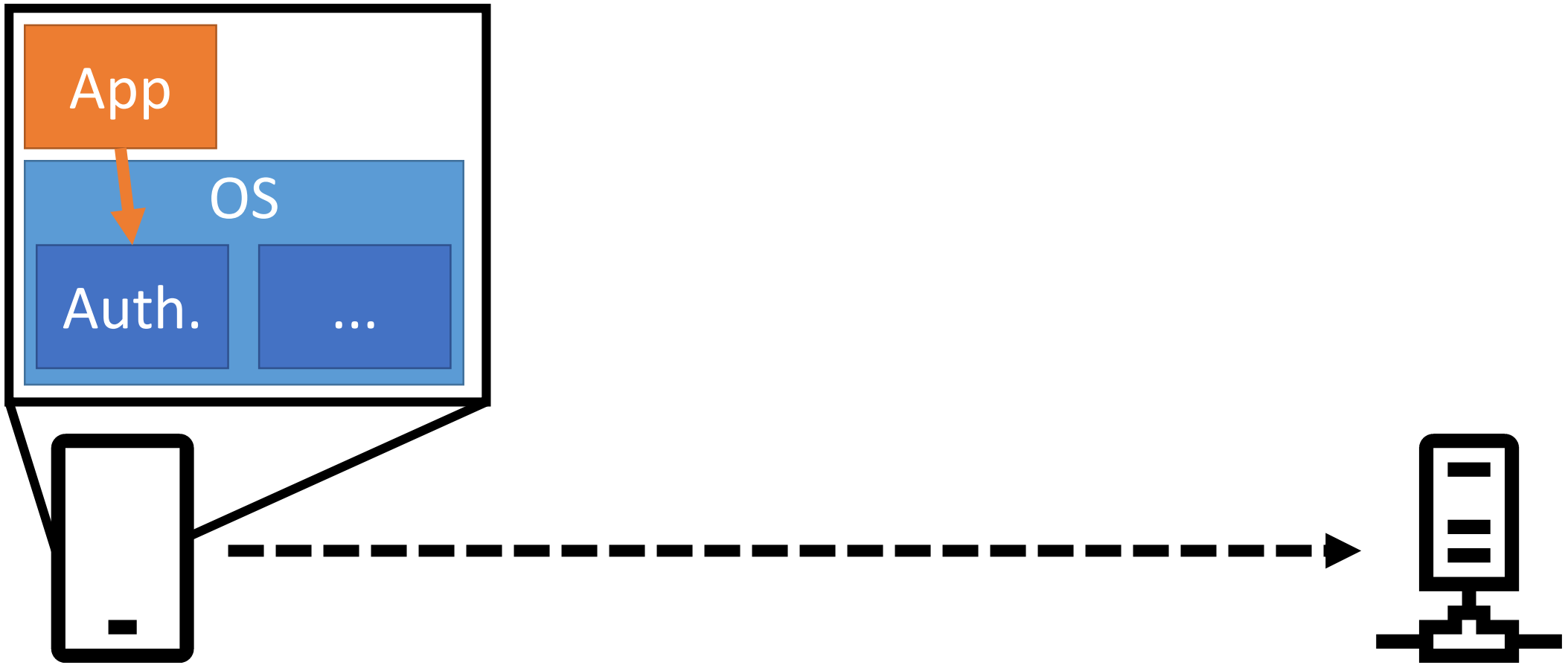
```
int sum(int... ints) {
    int sum = 0;
    for (int i : ints) {
        sum += i;
    }
    return sum;
}

@Test
void sumOfOneToThreeIsSix() {
    assertThat(sum(1, 2, 3), is(6));
}
```

Modules

```
/** Posts the given text
    as the current user
    to the forum. */
void postMessage(String text);
```

Context



Make dependencies explicit

```
class User { ... }
interface AuthService {
  User getCurrentUser();
}
interface HttpClient {
  String post(String url, String body);
}
void postMessage(String text,
                 AuthService auth,
                 HttpClient client);
```

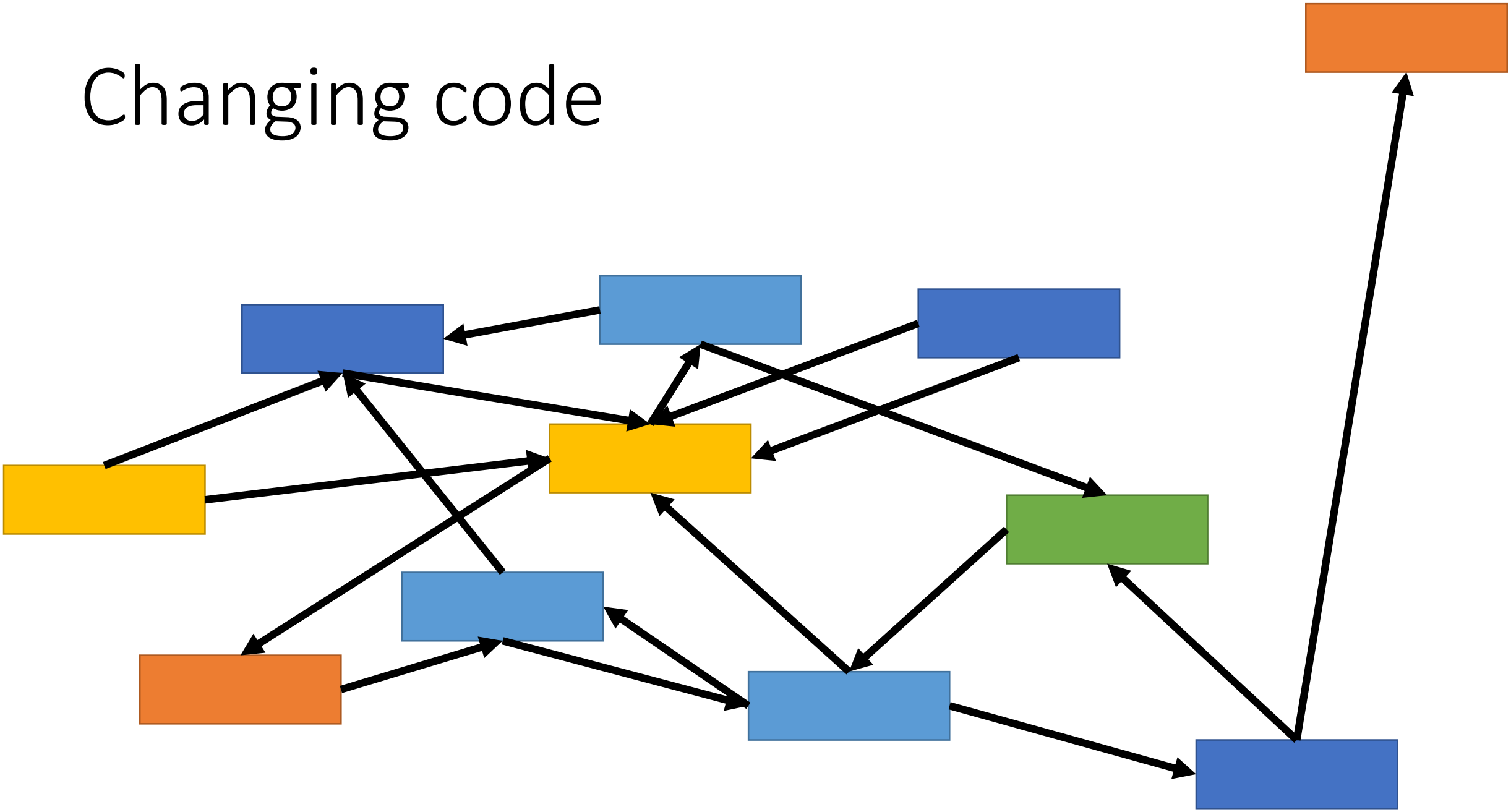
Inject fake dependencies

```
AuthService auth = new AuthService() {
    @Override
    public User getCurrentUser() {
        return null;
    }
};
HttpClient client = ...;
assertThrows(
  IllegalStateException.class,
  () -> postMessage("Hi", auth, client));
```

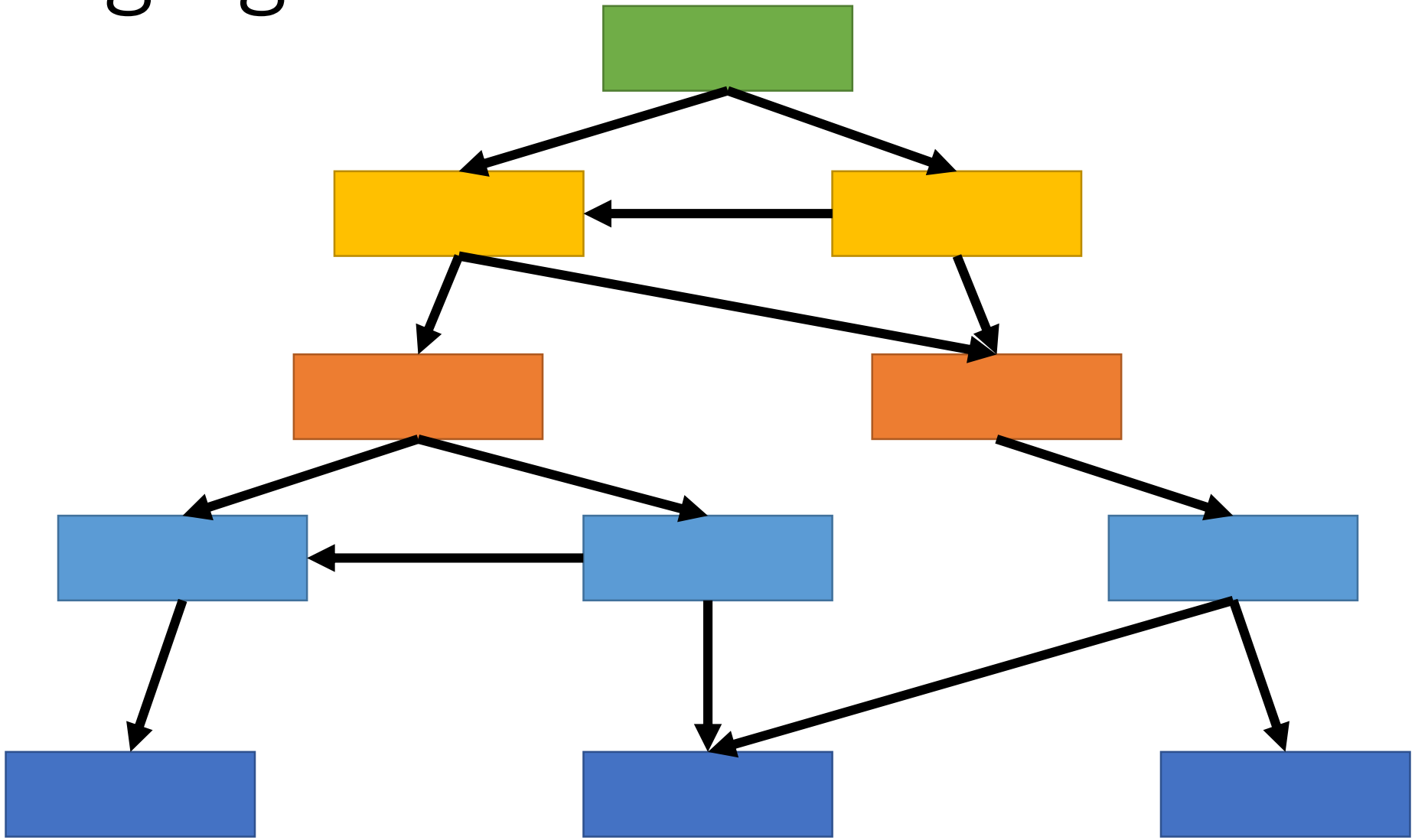
Test dependency calls

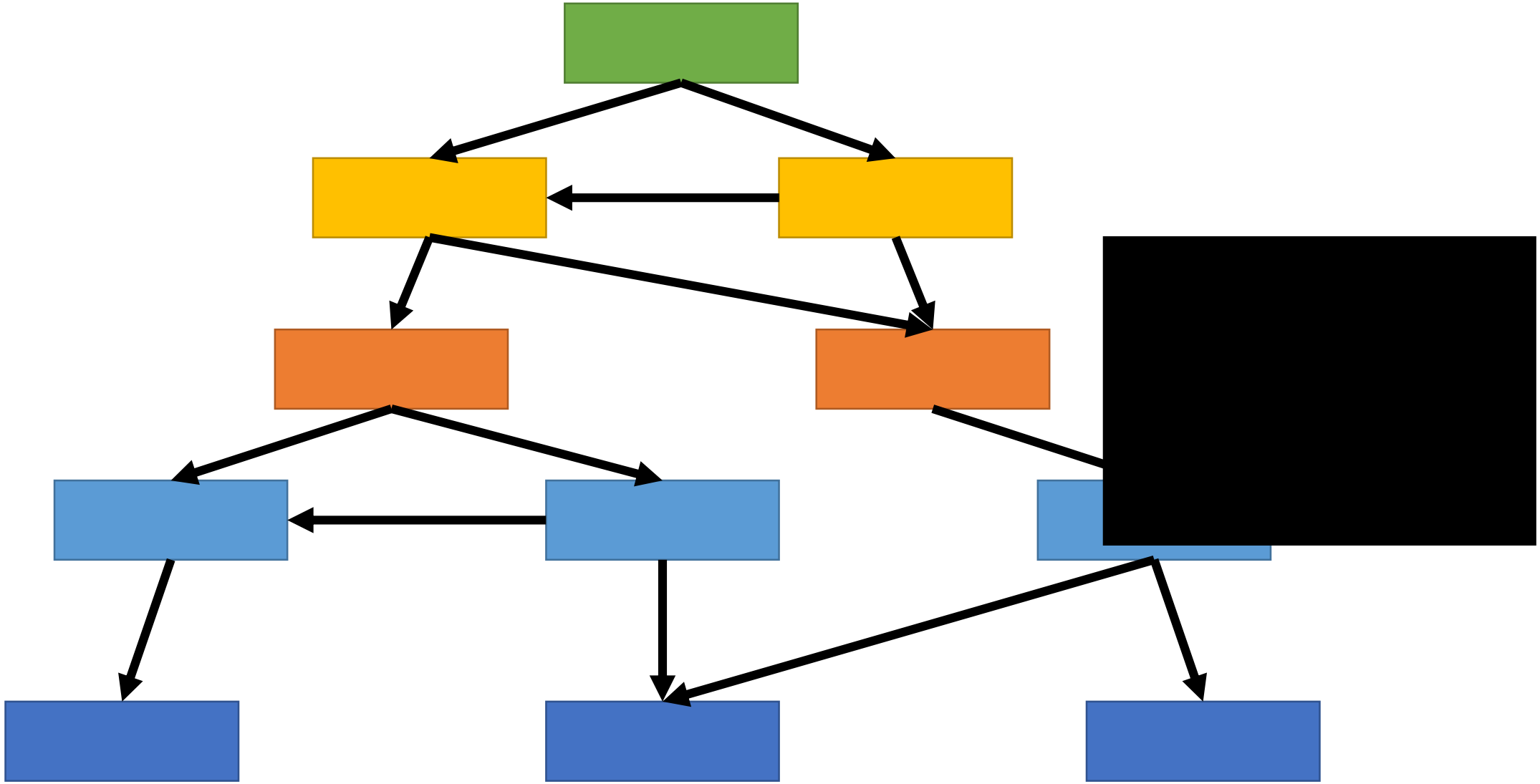
```
HttpClient client = new HttpClient() {
    @Override
    public String post(String url,
                       String body) {
        assertThat(body, is("MSG: Hi"));
        return "OK";
    }
};
```

Changing code

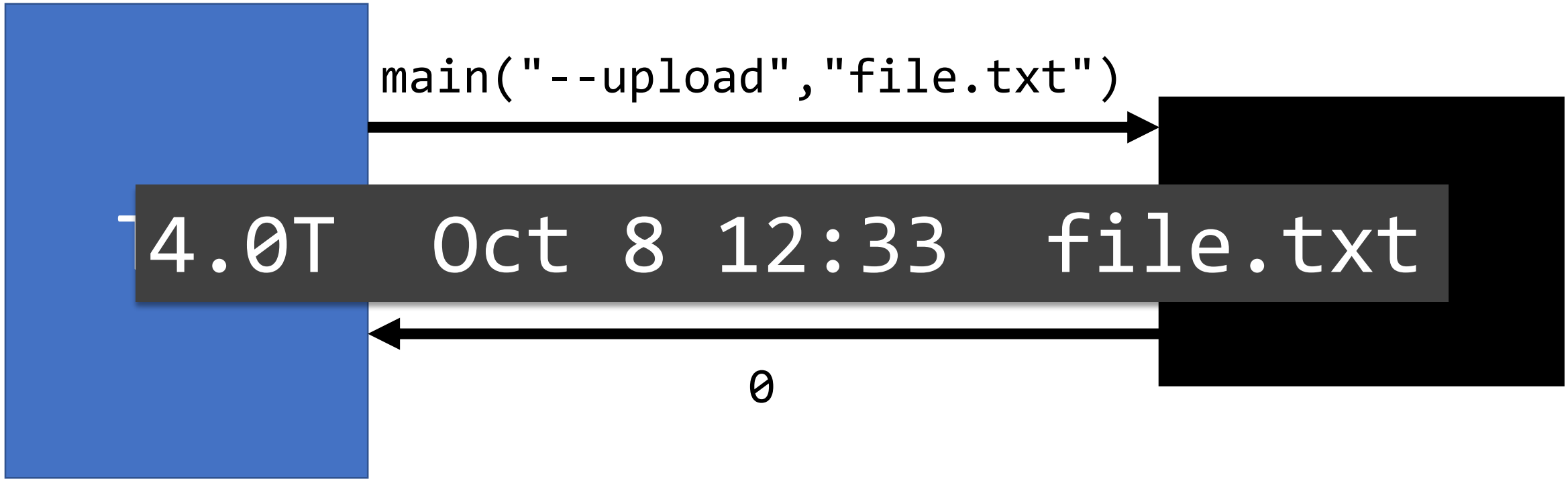


Changing code

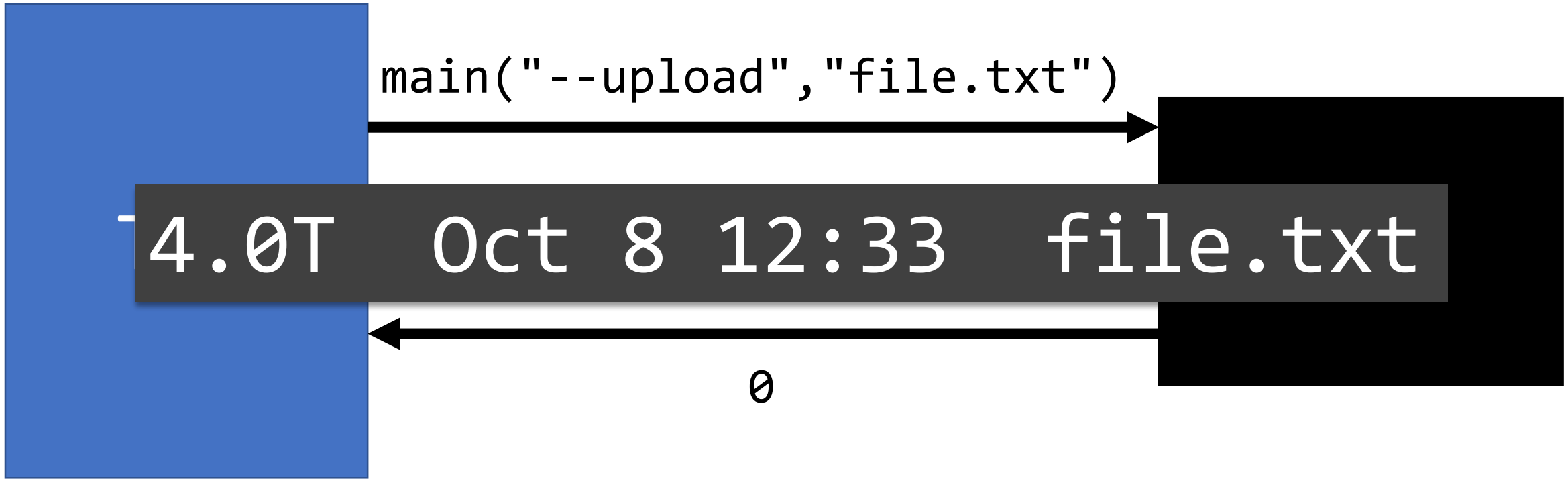
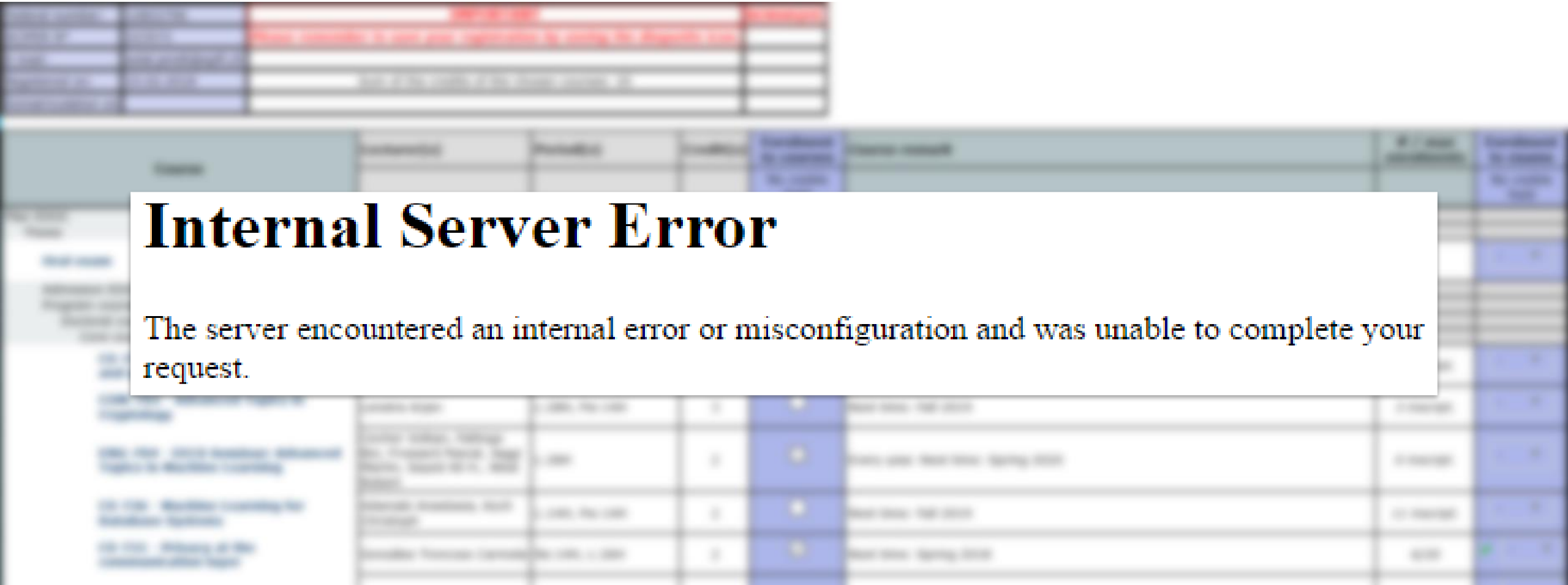




End-to-end testing



Stress testing



Stress testing