

The LockManager App Case Study: Test Driver Implementation

Douglas C. Schmidt

d.schmidt@vanderbilt.edu

www.dre.vanderbilt.edu/~schmidt

Professor of Computer Science

**Institute for Software
Integrated Systems**

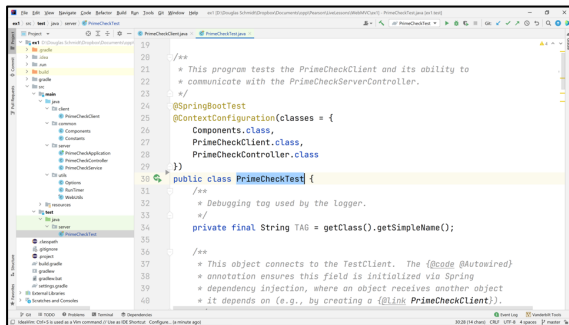
**Vanderbilt University
Nashville, Tennessee, USA**



Learning Objectives in this Part of the Lesson

- Understand the implementation of the LockManagerTest class & associated client code that invoke synchronous methods on the LockManagerController

LockManagerTest



```
19 /**
20  * This program tests the PrimeCheckClient and its ability to
21  * communicate with the PrimeCheckServerController.
22  */
23 //
24 @SpringBootTest
25 @ContextConfiguration(classes = {
26     Components.class,
27     PrimeCheckClient.class,
28     PrimeCheckController.class,
29 })
30 public class PrimeCheckTest {
31     /**
32      * Debugging tag used by the logger.
33      */
34     private final String TAG = getClass().getSimpleName();
35
36     /**
37      * This object connects to the TestClient. The @Code @Autowired
38      * annotation ensures this field is initialized via Spring
39      * dependency injection, where an object receives another object
40      * it depends on (e.g., by creating a @Link PrimeCheckClient?).
41      */
42 }
```



*Synchronous
HTTP GET/POST
requests/
responses*

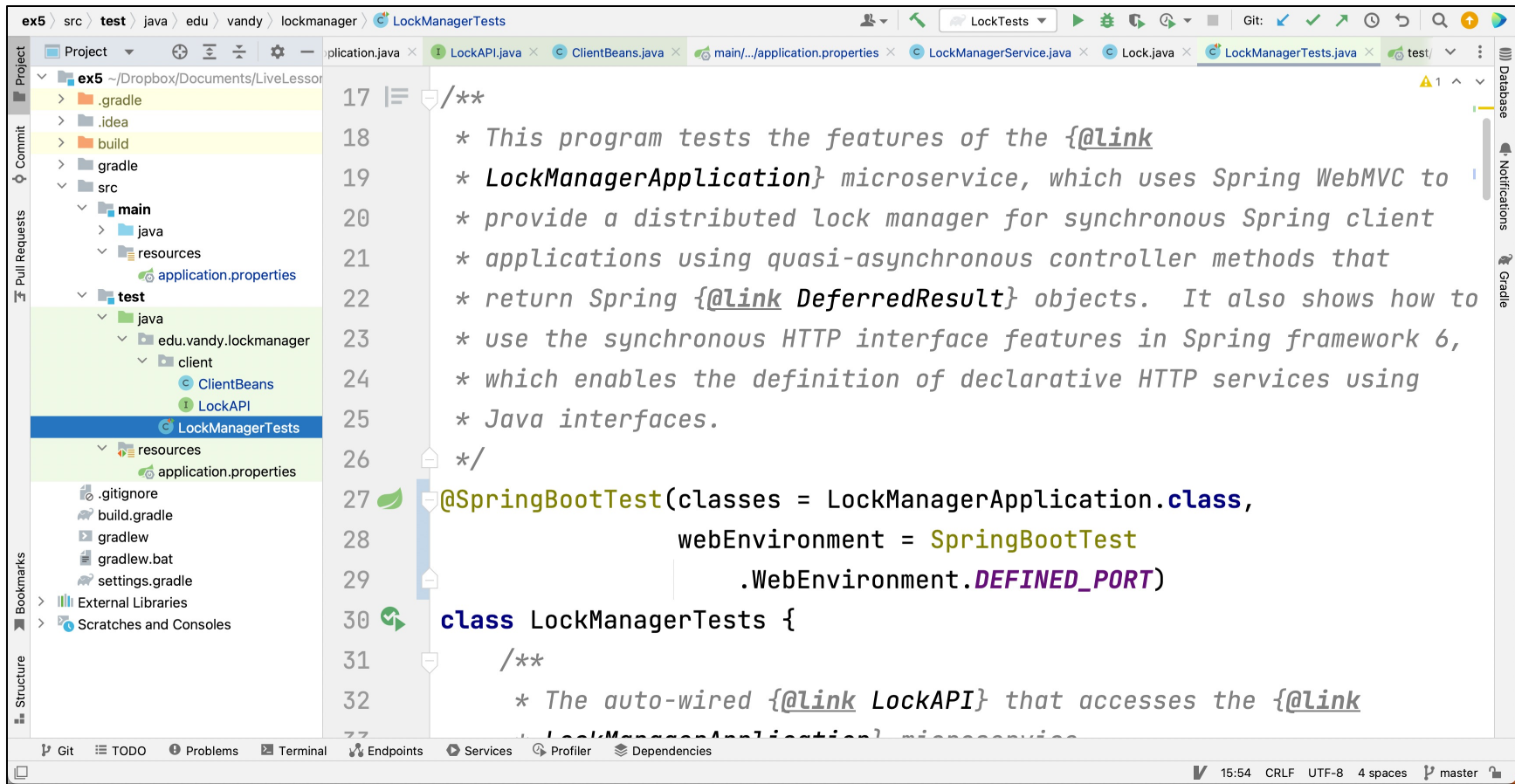
LockManagerApplication



See github.com/douglasraigschmidt/LiveLessons/tree/master/WebMVC/ex5

Implementing the LockManagerTest Driver

Implementing the LockManagerTest Driver



```
ex5 > src > test > java > edu > vandy > lockmanager > LockManagerTests

Project: ex5 ~~/Dropbox/Documents/LiveLesson
  > .gradle
  > .idea
  > build
  > gradle
  > src
    > main
      > java
      > resources
        > application.properties
    > test
      > java
        > edu.vandy.lockmanager
          > client
            > ClientBeans
            > LockAPI
          > LockManagerTests
        > resources
          > application.properties
      > .gitignore
      > build.gradle
      > gradlew
      > gradlew.bat
      > settings.gradle
  > External Libraries
  > Scratches and Consoles

17 /**
18  * This program tests the features of the {@link
19  * LockManagerApplication} microservice, which uses Spring WebMVC to
20  * provide a distributed lock manager for synchronous Spring client
21  * applications using quasi-asynchronous controller methods that
22  * return Spring {@link DeferredResult} objects. It also shows how to
23  * use the synchronous HTTP interface features in Spring framework 6,
24  * which enables the definition of declarative HTTP services using
25  * Java interfaces.
26  */
27 @SpringBootTest(classes = LockManagerApplication.class,
28                  webEnvironment = SpringBootTest
29                  .WebEnvironment.DEFINED_PORT)
30 class LockManagerTests {
31     /**
32     * The auto-wired {@link LockAPI} that accesses the {@link
33     * LockManagerApplication} microservice
```

See [WebMVC/ex5/src/test/java/edu/vandy/lockmanager](https://github.com/ex5/src/test/java/edu/vandy/lockmanager)

End of the LockManager App Case Study: Test Driver