

# Download the SOAP API Test Framework

In this lesson, we will download and understand the framework structure.

## We'll cover the following

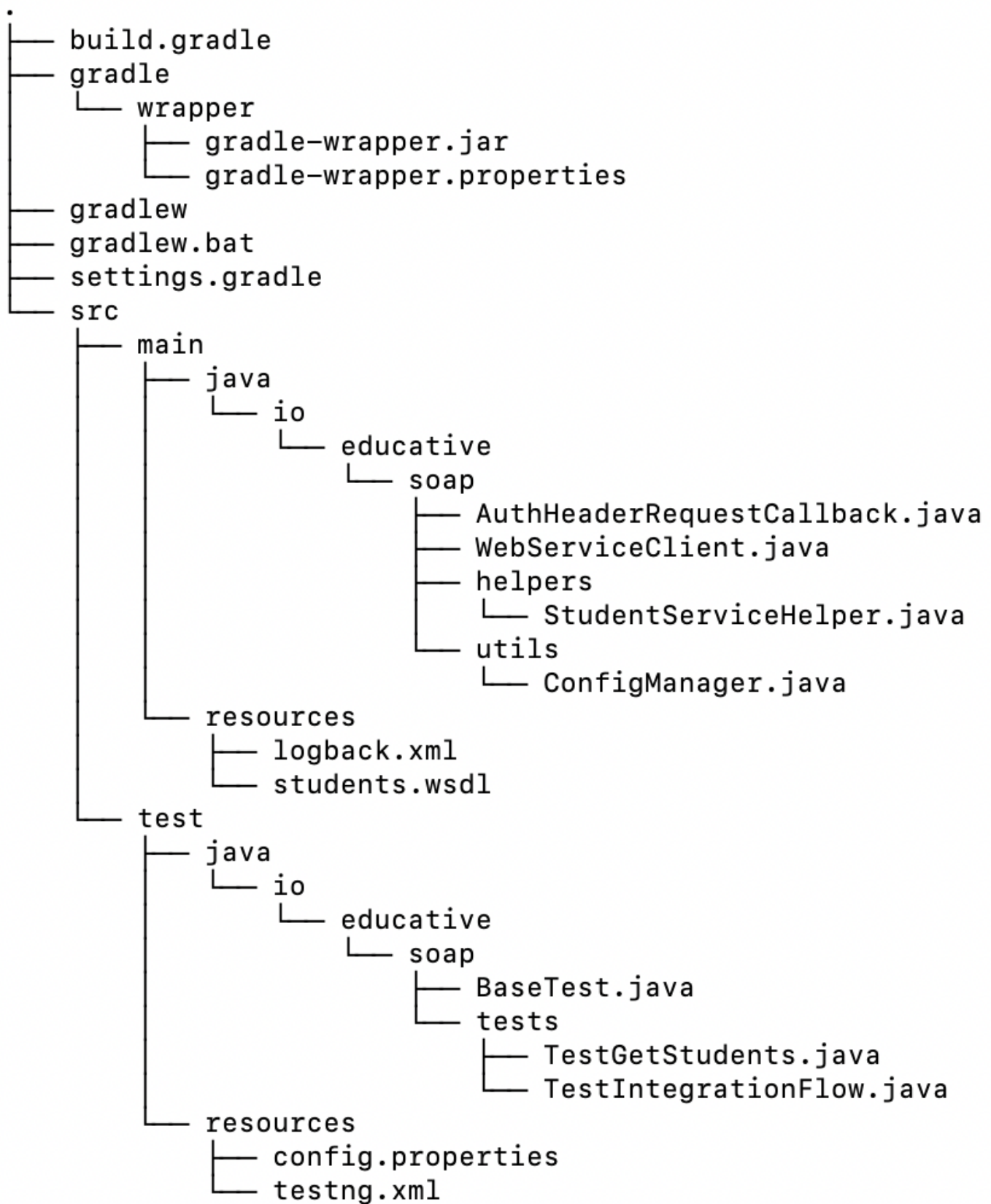


- Downloading the framework code
- Project structure
  - src/main/java
  - src/main/resources
- Sample tests written using the framework
  - src/test/java
  - src/test/resources

## Downloading the framework code #

This is a sample framework that can be extended and reused for any SOAP web service testing requirement. It can be downloaded [here](#).

## Project structure #



18 directories, 17 files

src/main/java #

- `io.educative.soap.AuthHeaderRequestCallback` – an implementation of `org.springframework.ws.client.core.WebServiceMessageCallback` that helps us construct `SOAP Header` for sending plain text authorization username and password

- `io.educative.soap.WebServiceClient` – a class that holds the bean definition of `WebServiceTemplate` which is used to make web service calls
- `io.educative.soap.helpers.StudentServiceHelper` – a class for writing helper methods used in making a request
- `io.educative.soap.utils.ConfigManager` – a class for configuration management that takes configuration parameters from project-specific `config.properties` which can be overridden by a JVM argument

`src/main/resources` #

- `logback.xml` – the configuration for logback
- `students.wsdl` – WSDL of the web services that we are writing tests for

## Sample tests written using the framework #

`src/test/java` #

- `io.educative.soap.BaseTest` – an abstract class that all test class extend which contains methods to load Spring Beans at the start of the test suite and initialize the `WebServiceTemplate` which the `io.educative.soap.helpers.StudentServiceHelper` will need to make web service calls
- `io.educative.soap.tests.TestGetStudents` – a sample test class for testing `GetStudents` web service - with and without authorization
- `io.educative.soap.tests.TestIntegrationFlow` – a sample test class that tests the complete flow of creation, fetch, update, and deletion of a `Student`

`src/test/resources` #

- `config.properties` – a property file that holds the project-specific configuration parameters and can be read using `ConfigManager`
- `testng.xml` – a suite of TestNG tests

---

Now, we are familiar with the framework code structure. In the next lesson, we will learn to build, run, and experiment with the downloaded tests.