**Integration Testing Strategy *(6.2.3.)***

When the overall architecture of the application considered, we have concluded to use the *“Bottom-Up Approach”* for the integration and testing. By using this approach, we have first tested our low level components. Then, we have used these components to test the following higher level components. We have preferred this approach in order to make the localization of the possible faults easier.

We have used the below figure, to briefly explain the hierarchical representation of the components in our application. On the figure, the numbers that are given in the right corner of each component identifies the testing order of the component and the arrow between each component represents the *“uses relation”* between the two components. *(the arrow is pointing from the component to the one which it uses)*



Figure 1 - Overview of the component testing order

Since, we are following *“Bottom-Up Approach”*, we need to use drivers for each module. These drivers represent a higher level component for the module which has been currently going through the testing process. Then, for all the following steps, we will replace those drivers with the module itself and use its drivers to proceed with the testing.

The figure which is included below, gives a brief representation of the testing procedure by simulating the integration testing of the components; *“LoginService”* and *“DataAcquisitionService”.*

**Integration Testing of *LoginService*:**

During the integration testing of *“LoginService”,* two components; *“DataAcquisitionService”* and *“AccountService”* which uses it are represented as drivers.

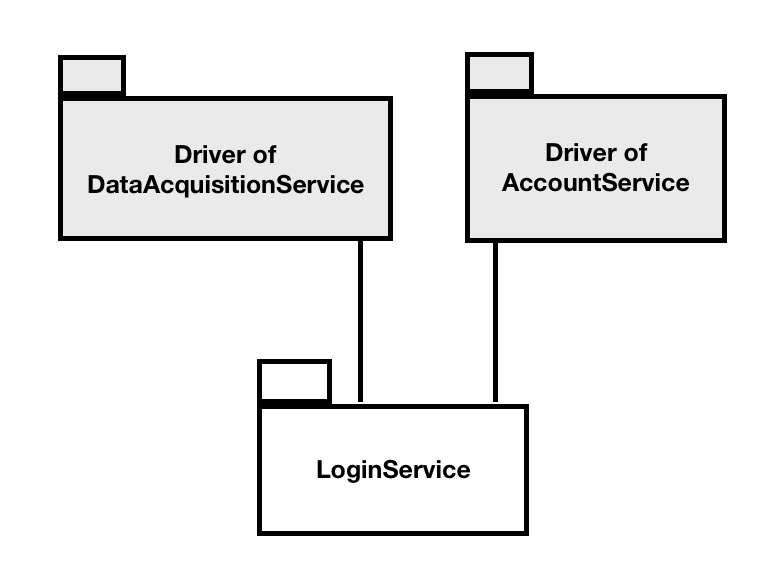


Figure 2 - Integration testing of LoginService

**Integration Testing of *DataAcquisitionService*:**

During the integration testing of *“DataAcquisitionService”,* two components; *“SOSService”* and *“RunService”* which use it are represented as drivers. On the other, the other two components; *“LoginService”* and *“DataService”* which are used by *“DataAcquisitionService”* are represented as normal components since they were already tested.

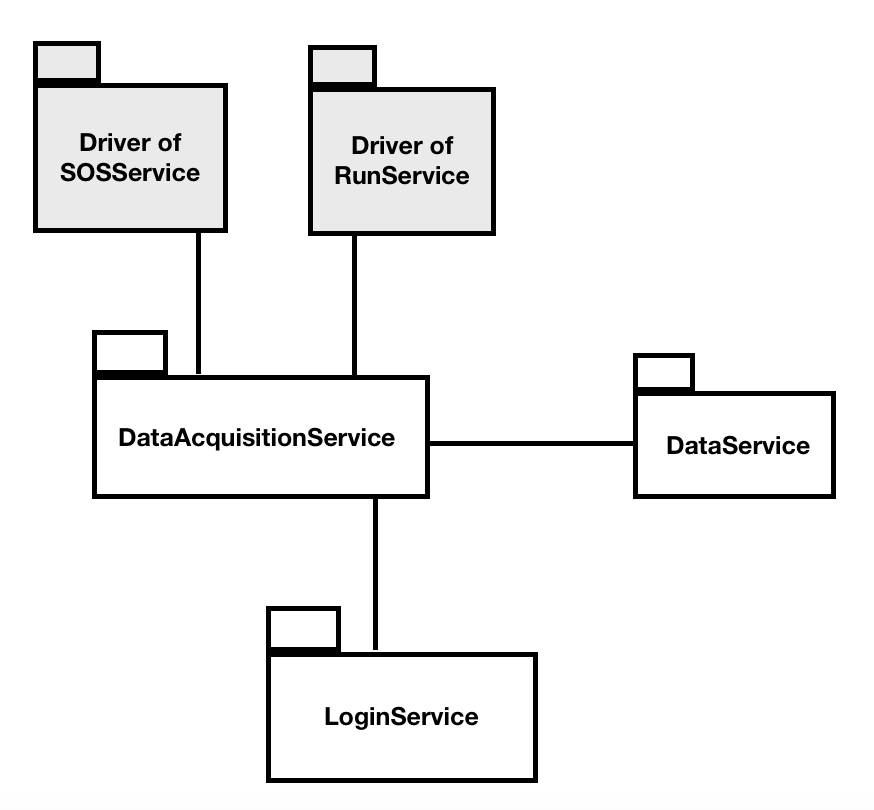
****

Figure 3 - Integration testing of DataAcquisitionService

**References**

<https://en.wikipedia.org/wiki/Integration_testing>

<https://www.quora.com/What-is-bottom-up-integration-testing>