With respect to the unit “Create Student Account”, two test cases were analyzed:

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
|  | Create Student Account Test Cases | |  |  |
| Test Case | | **Description** | **Input provided** | **Expected Output** |
| 1 | | Email and password combination provided is invalid | Email = “test.com”  Password = “passw” | User would be notified of their error, and the account would not be created |
| 2 | | Email and password combination provided are valid | Email = “test@case.com”  Password = “password” | Account would be created, user would be redirected to main page |

For a detailed list of all possible test cases related to the requirements of “Requesting a New Account” (i.e.: The requirement that makes up this unit), refer to section 1.2.2. It is important to note that a test case was not created for when first or last names are not provided. As both fields are required, the result will never stray from simply asking for both again should they not be present (seen in the test stubs/driver code, as both fields are required). The results of the testing can be seen below. These two test cases were derived through grouping, and stubs were used in order to create mock student account (and other testing components), while a driver was created to simulate a new completed account and authentication of its values was allowed as a result. In order to simplify these two test cases, only two test functions were made: Create1 and Create2. Create1 would always return false (signifying an invalid email/password combination), while Create2 would always return true. In this way, though several different inputs could realistically be used, the results were simplified to two categories: Valid and Invalid. This simplified the act of testing and allowed the unit to be easily tested as a whole