**Tendable Tech Challenge**  https://www.tendable.com website:

For this challenge, you'll need to upload your solution to GitHub (https://github.com/). If you don't already have a GitHub account, please set one up before beginning.

Your task is to craft an automated test, preferably in C# or Java with Selenium or with any popular language of your choice, to evaluate the following elements on the1. Confirm accessibility of the top-level menus: Home, Our Story, Our Solution, and Why Tendable.

@Test(priority = 1) public void testTopLevelMenusAccessibility() { String[] topMenus = {"Home", "Our Story", "Our Solution", "Why Tendable"}; for (String menu : topMenus) { WebElement menuElement = wait.until(ExpectedConditions.visibilityOfElementLocated(By.linkText(menu))); Assert.assertTrue(menuElement.isDisplayed(), menu + " menu is not accessible."); menuElement.click();

2. Verify that the "Request a Demo" button is present and active on each of the aforementioned top-level menu pages.

WebElement demoButton = wait.until(ExpectedConditions.visibilityOfElementLocated(By.xpath("//a[text()='Request a Demo']"))); Assert.assertTrue(demoButton.isDisplayed(), "Request a Demo button is not present on " + menu + " page.");

3. Navigate to the "Contact Us" section, choose "Marketing", and complete the form— excluding the "Message" field. On submission, an error should arise. Ensure your script confirms the error message's appearance. If the error is displayed, mark the test as PASS. If absent, it's a FAIL.

@Test(priority = 2) public void testContactUsForm() { driver.findElement(By.linkText("Contact Us")).click(); // Select "Marketing" from dropdown WebElement inquiryType = wait.until(ExpectedConditions.elementToBeClickable(By.name("inquiry\_type"))); inquiryType.sendKeys("Marketing"); // Fill the email field driver.findElement(By.name("email")).sendKeys("test@example.com"); // Submit the form driver.findElement(By.name("submit")).click();

4. In the event the test for point 3 fails (meaning no error message is displayed when the "Message" field is empty), draft a bug report. This can be a Word document that you'll upload to GitHub alongside your code.

try{WebElement errorMessage = wait.until(ExpectedConditions.visibilityOfElementLocated(By.className("error-message"))); // Adjust the selector based on the actual error message element Assert.assertTrue(errorMessage.isDisplayed(), "Error message is not displayed when Message field is empty."); }

catch (Exception e)

{ generateBugReport(); Assert.fail("Error message was not displayed. Bug report generated."); } }