

**School of Engineering and**

**Information Technology**

**ICT286 MAJOR ASSIGNMENT TWO**

**CHECK LIST**

**Group Members (full name and student number):**

**Member 1:**\_Mitchell Panicciari - 32674186\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

**Member 2:**\_Lachlan Blackhall - 32667253\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

**Tutor’s Name:** \_Mohammad Alhabashneh\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

**Assignment Due Date: \_7/Nov/2016 12:00PM**\_ **Date Submitted: \_7/Nov/2016**\_\_\_\_\_\_\_\_

**Your assignment should meet the following requirements. Please confirm this (by ticking boxes) before submitting your assignment.**

**✓**All details above are complete.

**✓**The archive file (a zip file) contains the file Assignment2.doc.

**✓**The archive file contains a sub-directory named “WebApp” which contains all files related to the web application.

**✓**The archive file contains a sub-directory named “MobileApp” which contains the files for the mobile application.

**✓**The documentation was prepared according to Documentation Requirements specified in Major Assignment 2’s question sheet.

**✓**This archive file will be submitted to ICT286 Unit LMS.

**✓**We have kept a copy of this assignment, including this archive file.

**✓**Our web application is hosted on the server ceto.murdoch.edu.au. Our web application’s url is

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

**✓We have signed the Group Declaration in the next page.**

**Group Declaration**

As a group assignment, each member of the group is expected to make an equal contribution to the assignment and receives the same mark for the assignment.

However, we recognise that on some occasions and due to various reasons, the actual contributions to the assignment from the two members could be unequal despite the best efforts of each member. In this case, we can still accept your assignment provided that all members of the group reach an agreement on their percentages of contribution to the assignment, and the agreement accurately reflects the real contribution by each member. In such a case, a member’s mark is linked to his or her agreed contribution to the assignment and is calculated using the following formula:

A member’s mark = minimum ( group mark x the member’s percentage of contribution x 2, group mark + 10, 100 )

On some rare occasions, the two members of the group fail to reach an agreement on their contributions to the assignment. In such a case, in order for your assignment to be marked, each member of the group must complete a detailed *Task Breakdown List* and state his or her own claim of the percentage of contribution to the assignment. Your tutor will then award each member a mark based on his assessment of the quality of the assignment as whole as well as his assessment of that member’s contribution to the assignment.

Please complete and sign ***one*** of the three declarations below:

|  |
| --- |
| *We have made an* ***equal*** *contribution to this assignment. We understand that each of us will receive the same mark for this assignment.*  Signature (member 1): \_mpanicciari\_\_\_\_\_\_\_\_\_Date: \_06/11/2016\_\_\_\_\_\_\_\_\_\_\_\_\_  Signature (member 2): \_\_lblackhall\_\_\_\_\_\_\_\_\_\_Date: \_06/11/2016\_\_\_\_\_\_\_\_\_\_\_\_\_ |

**The task description:** Your task is to design and implement a web and a mobile application for a fictitious company so that it can conduct its trade over the Internet.

The fictitious company for which your application will be designed must meet all requirements specified in the next section.

You are required to use the following technologies to implement the web and mobile application: HTML5, CSS, JavaScript, PHP, MySQL, and Apache Cordova. You should use Ajax technology to speed up the application and to reduce unnecessary page loading, eg, searching, user authentication, and database update. You must not use any web content management framework/system to create this application unless you have obtained a written approval from the Unit Coordinator.

**Discussion of your solution:** For our solution we chose to make our whole site a one page application just for ease of use and to cut down on loads times. The navigation is always going to be consistent because we use the one page and have a section inside the body loaded to different HTML using Ajax. The result of our solution does meet our expectations and has come out a lot better than we first expected the simple look of the site allows for users to navigate and find what they want easily.

We believe a strong point of our solution is that it is very easy to add new features as it is very modular our scripts are separated and we are using Ajax to load html into our page without having it refreshed. Almost everything has its own element and container which allows for changing of elements with ease without needing to change a lot.

An improvement that could be made to our solution would be how the layout is we found it difficult to have the login area fit in nicely with the surrounding elements, once you have logged in the way the buttons are aligned does not look as good as it could in our opinion.

**Self-diagnosis and evaluation**:

**Features fully working:**

Our solution contains the basic information about the company as created in the fictitious company description, it is accessible via a button on the nav bar this button is named about us and contains all necessary information.

Our navigation system is consistent throughout the whole solution because we only have one page and use Ajax to load in different html elements as the user uses the website.

Our website provides the user with a help button which will open a new tab with a PDF document explaining how to use our website.

Our website offers a sitemap which shows the structure of our site.

Our site implements some of the html5 semantic elements such as header, footer, nav and section to define the overall structure of our web page. We have also used CSS externally to define our styles and visual layout of the solution.

Our website has been primarily tested on the latest version of Google Chrome but it also has full functionality on both the latest versions of Mozilla Firefox and Internet Explorer. Our web page is served through CETO and is therefore using ApacheHTTP and PHP is enabled.

|  |  |  |
| --- | --- | --- |
| **Test documentation for your web application** | | |
| **Test** | **Outcome** | **Evidence(Screenshot)** |
| About Us button displays information about the website | About us button changes content of the container to show information about the company |  |
| Navigation system stays the same throughout | Content is served dynamically through AJAX. In this example price <20 is selected in the first picture and genre racing is selected in the second picture |  |
| Help button shows page for how to use the site | Help button opens a PDF document in a separate window with information about how our site works |  |
| Site map shows structure of the website | Site map shows how the site is structured |  |
| Page loads on Google Chrome | Page loads on Google Chrome |  |
| Page loads on IE | Page loads on IE |  |
| Page loads on Mozilla | Page loads on Mozilla |  |
| User searches valid search | Products are displayed depending on search criteria |  |
| User tries to search with empty field | No action is performed and the page does not change |  |
| Users last search entry is saved | Users last search value is saved and kept even if they go to another option |  |
| User uses price button on the menu to show products | Different products are displayed depending on the button pressed, <20 pressed and >50 pressed |  |
| User uses genre button on the menu to show products | Different products are displayed depending on the button pressed, FPS pressed and Fighter pressed |  |
| User enters correct login information | User is able to login to the website and access other features |  |
| User enters incorrect login information | User is unable to login to the website and is alerted that they have entered it incorrectly |  |
| User presses log out button | User is logged out and the login screen is shown |  |
| User presses edit user details button | USer is displayed with a form to edit their information |  |
| Staff presses add products button | Add products form is shown |  |
| User edits their details and submits | Their details are changed in the database |  |
| Staff enters product information and submits | The new product is placed into the database and shown when searching for products |  |
| Registered user adds product to their cart | The added product will show up in their cart |  |
| User presses the home button | User is taken back to the home page which displays all of the products |  |

|  |  |  |
| --- | --- | --- |
| **Test documentation for your mobile application**: | | |
| **Purpose of the test** | **Outcome** | **Evidence(Screenshot)** |
| About Us button displays information about the website | Button displays about us page |  |
| Navigation system stays the same throughout | As the content is served on a single page through Ajax the nav system is not affected |  |
| Help button shows page for how to use the site | The help content is loaded in place of the search data |  |
| Site map shows structure of the website | The site map is loaded, the user can return to the app by tapping their back button |  |
| Page loads on Android | The app loads on a Nexus 6 emulator running android, the normal page can also be viewed via a web browser. |  |
| User searches valid search | The search function displays results for what the user searches |  |
| User tries to search with empty field | If the user tries to search with a blank field nothing shows and the results of the last search are kept |  |
| Users last search entry is saved | As can be seen in the same picture as before the last search is shown in a faded grey when the search bar is blank |  |
| User uses price button on the menu to show products | When the user clicks a price button only results matching that condition are returned |  |
| User uses genre button on the menu to show products | When the user clicks on a genre button only products related to that genre is shown |  |
| User enters correct login information | When the user enters the correct login information they are logged in and given the appropriate functions |  |
| User enters incorrect login information | The app displays a message informing the user incorrect information was entered |  |
| User presses log out button | The app refreshes and displays the login fields |  |
| User presses edit user details button | The product display section is filled with the fields to edit user details |  |
| User presses go to cart button | If the user has products in their cart it takes them to the confirmation screen, if not it prompts them to add products |  |
| User presses add products button | The app loads the menu to add product details |  |
| User edits their details and submits | The details are saved and the user is returned to the home screen |  |
| User enters product information and submits | The product information is saved and the app refreshes to show the home screen with the new product |  |
| Registered user adds product to their cart | Product is viewable in the cart menu option |  |
| User presses the home button | App refreshes to display basic product view page |  |