|  |  |
| --- | --- |
| **CS230 – Web Information Processing**  **Interim Report Submission** | |
| **Team Name: TeamD2** |  |

**Team Details:**

Members:

1. Alan O’Driscoll
2. Arwa Al-Ghaithi
3. Dara Folan
4. Dean Gilmore (leader)
5. Ionatan Sala
6. Ciaran

**List of Team Members and Responsibilities Assigned:**

* Sample HTML / CSS 🡪 Group
* PostgreSQL 🡪 Dean, Alan
* HTML 🡪 Ionatan, Ciaran
* CSS 🡪 Arwa, Ionatan
* Interim Report 🡪 Alan, Arwa, Dara, Dean, Ionatan
* Javascript 🡪 Arwa, Dara, Ionatan
* PHP 🡪 Alan, Dara, Dean
* Presentation 🡪 Dean
* Submission 🡪 Dean

**Project Description:**

For our group project we are building an E-commerce website that focuses on the sale of computer games. The users will be able to register with our site after which they can log in and browse our game database. There will also be a search function for the user to find a specific title.

Items can then be added to a shopping cart where they will be stored until the customer wishes to checkout at which time the total cost of the cart will be calculated and displayed to the user on a payment page.

**Functional Requirements**

* Login and registration page – users can register for the site in which case their details are added to a user database for which they can then log in.
* Browse – a top navigation bar lists each console and when clicked, loads a list of games for that console. The list can then be sorted by price, title, etc.
* Search – the navigation bar also contains a search bar that searches the databases for entries containing the searches string and returns the results.
* Cart – items can then be selected for purchase and added to a cart which stores the product, price and quantity. Items must be able to be added and removed from the cart by the user.
* Checkout – all cart items are quantified and prices totaled. This will likely lead to a mock payment page where the customer can purchase the items.

**Non Functional Requirements**

* External javascript and css files must be used
* Same navigation bar across all pages
* Search function must be available on all pages
* All non trivial code must be commented
* Site must be tested with multiple browsers

\*\* Gantt Chart Separate to Word Document\*\*

**System Architecture Diagram**



**Database Design and Queries**

For our project we decided to build 2 basic tables in which to hold the data for our Ecommerce site.

The first table contains the title, publisher, price, platform and description of the game in question. This is the main table which we will query and return results from.

The second table includes the customers name, details, username and password. This table will be queried on the login section of our homepage.

The home page will contain a search bar for items by:

* Title (title = “Call of Duty” or partial title like title = “%Call%)
* Publisher (publisher = “Blizzard”)

There will then be a navigation bar at the top of the home page listing all platforms. Clicking into one of these sections will display all games for that platform.

* (SELECT \* FROM database WHERE platform = “Xbox One”;)

The homepage will also contain a login section where a user can enter a username and password.

* (SELECT name FROM userbase WHERE username = “$input1“ AND password = “ $input2“ ;)

These details are checked against entries in the user table and the customers name is returned with a welcome message along with a list of recommended games or offers for the customer to purchase.

**PHP Specifications:**

uniqid()

* Create a unique id for a user or transaction

pg\_connect()

* Open connection to games and user database

pg\_close()

* Upon logout, close connection to database

pg\_insert()

* Allow admin to insert into database
* Allow user to insert into cart

pg\_update()

* Updates view after item is added or deleted

pg\_socket()

* Read only view for users when accessing the database

pg\_query()

* Query the database for specific searches

pg\_num\_rows() & pg\_num\_fields()

* Apply specifics to a search. Able to return more specific results rather than the whole table.

pg\_convert()

* Input search parameters from an array into an sql statement.
* May be used, depending on how we intake search parameters.

pg\_dbname()

* Get the database name. Can be useful if we match the database name with the cart / user.

pg\_string\_escape()

* May be able to take parameters out of a search if string does not

match. For example, a game was not released for a specific console

but exists for another.

**Javascript Specifications:**

.collapse

* Hides the content of the items

.collapse in

* Shows the content of the items

. collapsing

* added when the transition starts, and removed when it finishes

.match()

* used to match a regular expression against a string

.search

* executes the search for a match between the regular expression and a specified string

.getCartItems

* returns the items in our cart

.getCartCount

* returns a count of how many items are in our cart

.getCartTotal

* returns total price of cart items

.selectProduct

* sets which product option is currently selected

.addToCart

* adds the currently selected product to the cart and opens the cart

.closeCart

* close the cart

.openCart

* opens the cart

.removeFromCart

* removes item from the cart and closes the cart

.showCart

* shows the cart

.setCookie

* stores the name of the visitor in a cookie variable

.getCookie

* returns the value of a specified cookie

.checkCookie()

* checks if a cookie is set and displays greeting if true

**User Interface Design**

(a) **Home Page**

The homepage has a fixed navigation bar at the top where the user can navigate to specific pages on the project. The search box is also located in the navigation bar, a user can use the search box to search for a specific item. There is also a cart icon in the navigation bar, a user can click on the cart icon to bring up his cart at any point in time. The user can also click log in or sign up in the navigation bar. This navigation bar will be across all pages.

Below the navigation bar there is a jumbotron, a jumbotron is a box box that calls for extra attention.

After the jumbotron there is a featured items section where we have all of our featured items listed. The user can click on any of the featured items and this will bring him to a page with all the information about that specific item, in that page he can also add his item to the cart.

The last section in the homepage is the footer, with our copyright notice, and contact info.

(b) **Query page**

On the query page the users can use advanced search options to search for their specific item.

(c) **Results page**

The result page is the results of the query page. It’s gets all the items that matched your search options and lists them out for the user.

(d) **Shopping page**

The shopping page is a browse page the user click’s on the item he wants, he is then redirected to a page with all the information about the item as well as an option to add it to the cart.

(e) **Payment page**

The payment page is were the user can checkout from his cart and pay for his items.