Planetary Purchasing

CST336 Final Project

A Node based website to purchase planets and stars

By: Alejandro Caicedo
Ivan Alejandre
Randy Son
Matthew Chan

Introduction

This project consists of a Login Page which requires the user to enter credentials which will be compared against a MySQL database. The user will be welcomed to a welcome page which will have the most recent space based news as well as a navigation bar that updates depending on the user. From this page the user can search for planets to purchase, if they are an admin they can go to the admin page and edit the tables that make up the MySQL back end. The search page houses the item catalogs and allows the user to query the available items. Here the user is able to add items they like to the cart which is stored in the database. The shopping cart displays the user's cart, gives the user the ability to change any of their item's quantity as well as giving them the ability to remove items. A subtotal is displayed at the bottom for the user and a checkout button to lead them to the next step. The items in the user's cart is then displayed in the checkout preview which gives the user a chance to review their purchase before checking out. When the user proceeds to place an order, they will have their items recorded onto a transaction ID and removed from their cart. In the admin page, the administrator is able to add, update, or delete items from the products table in the database. They also have the ability to retrieve three different reports with different sub queries.

Login Page

The login page requires the user to enter a user name and password which are then POSTed to the appropriate route. In this case the "ac_login" route. In this route a DB connection is created, then a query is formed to return the users credential information. If the user name matches what is in the DB then the password hash is compared to the hash of the password that was entered. If those match then the "authenticated" boolean turns true. The isAdmin boolean is then parsed to determine if the user is an admin. These booleans along with the user name are saved in an express session for use in the other routes. If the user was authenticated in the previous step the welcome page is loaded, to it are passed API information on space news and the NASA picture of the day along with other user specific relevant information for rendering.

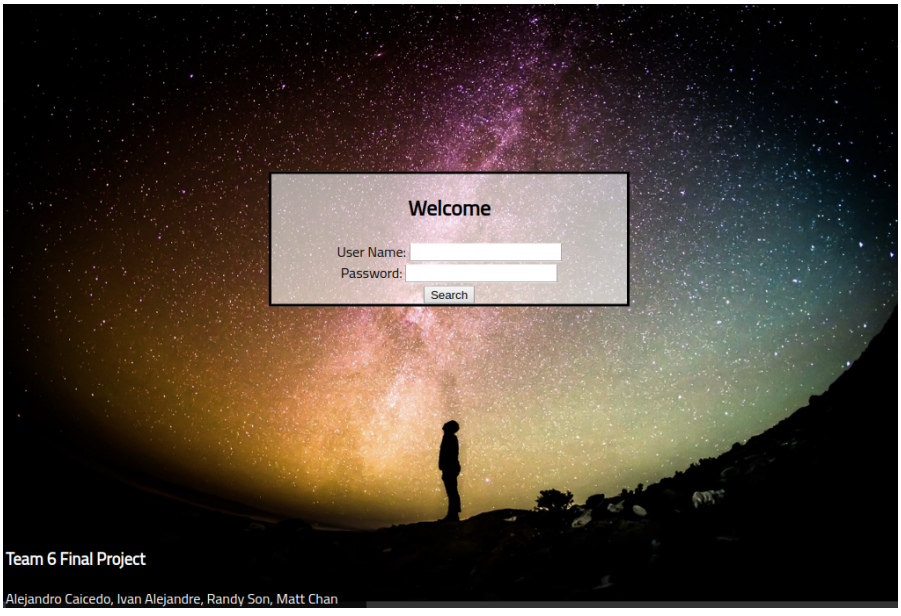
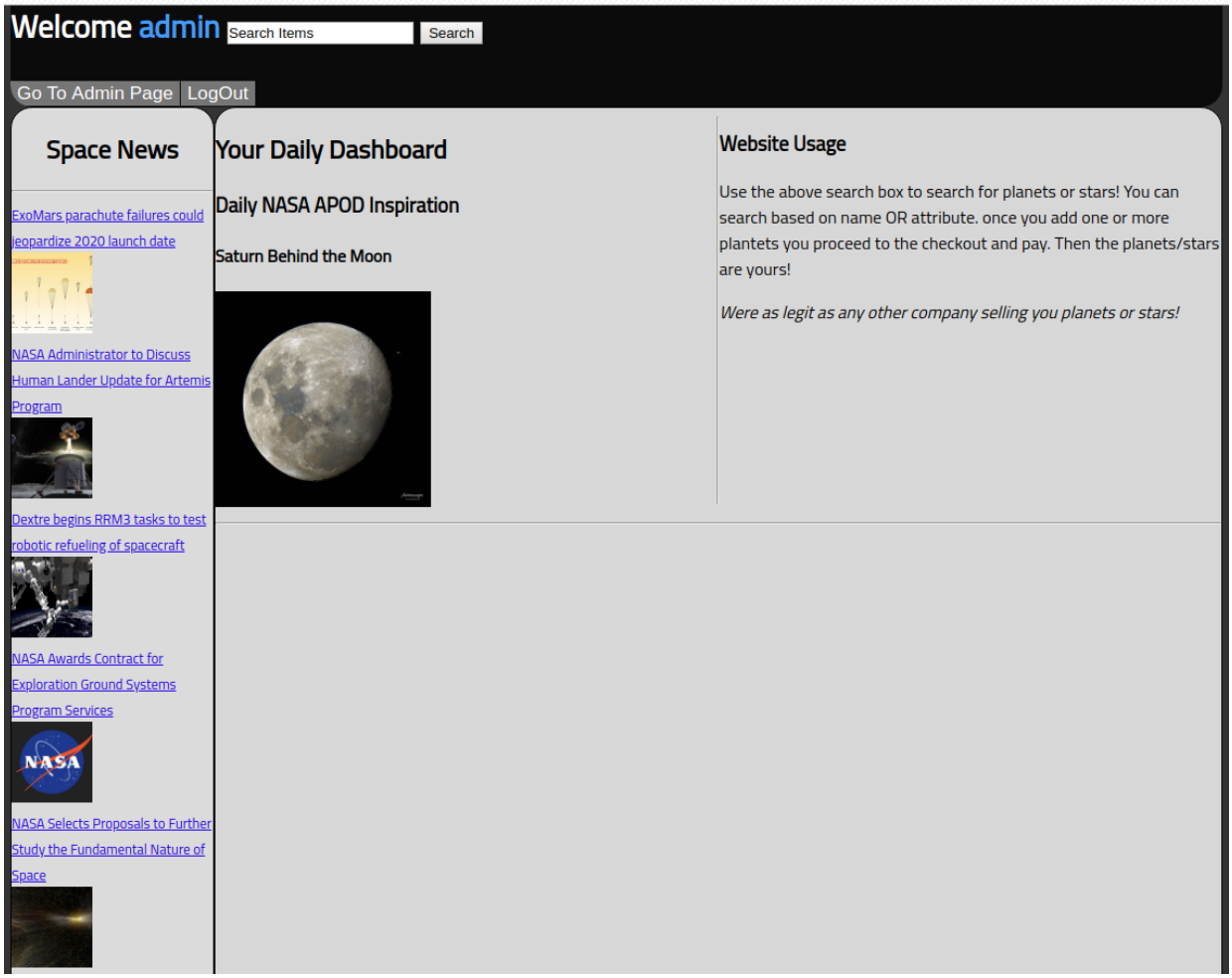


Image shown below.

Welcome Page

The welcome page shows the information passed to it from the “/ac_login” route. This includes 8 of the top space news stories from the “spaceflightnews.new” API as well as the NASA Picture of the day. The news is displayed in a column aptly called “Space News”, and the NASA picture of the day is in a section called Daily Inspiration. The buttons at the top of the admin page load based on the users “isAdmin” boolean. If the user is an admin then the “Go To Admin Page” button loads, if not then this does not load. The logout button loads irrespective of the users “isAdmin” status.



Search

The search page details the products available. It utilizes AJAX to fetch product information and can search the database using three criteria: the item’s name, the item’s price and the item’s description. This data is sent to the */api/querySearch* route. Here the SQL query is generated based on the checked filters and search query. Upon query success, each row from the query generates a product card containing details about the item. At the bottom of each product card is a quantity input and an “**Add to Cart**” button. Another API, */api/cartAction* handles adding, updating and removing items from

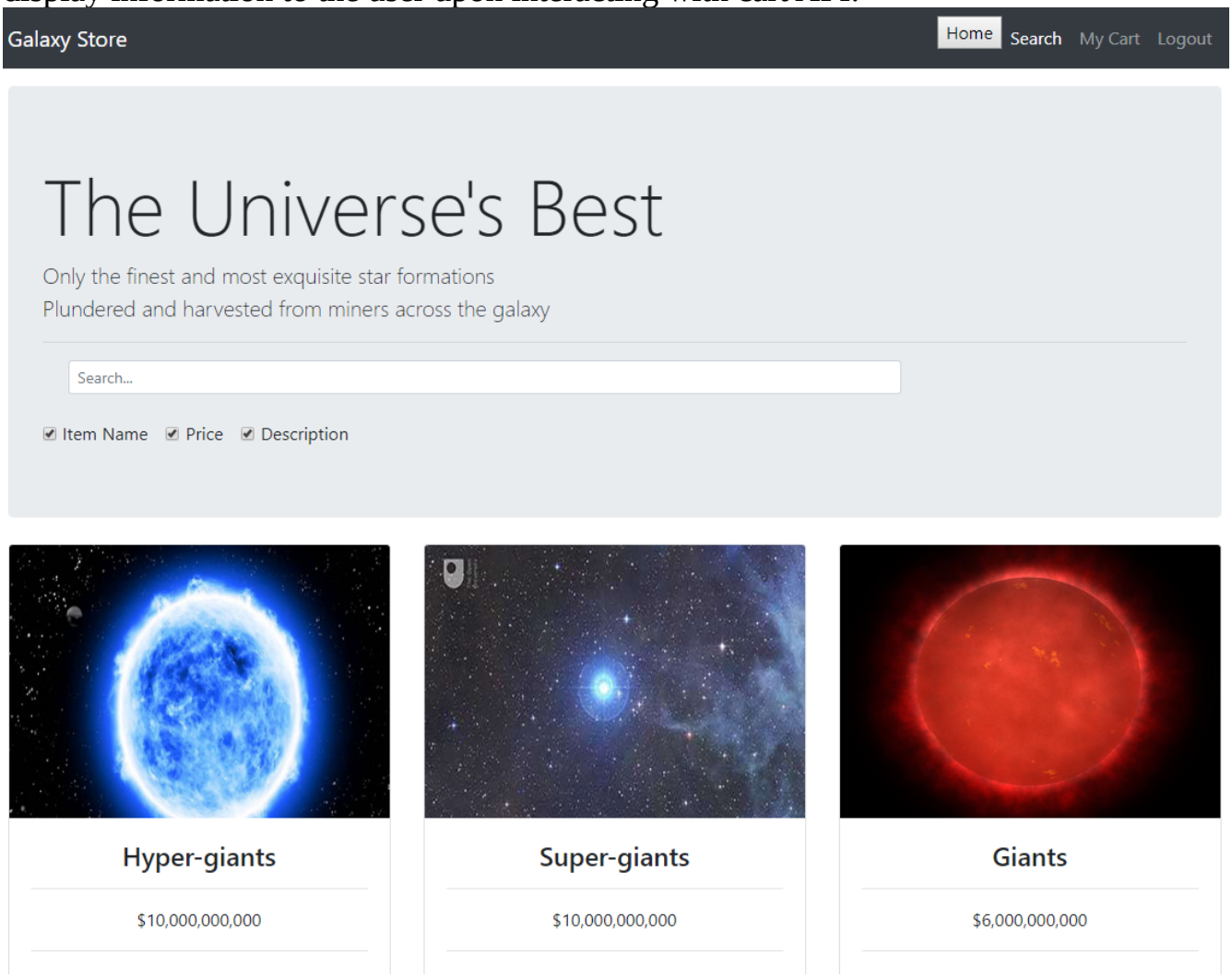
the shopping cart, (cst336_project.usercart). The API first checks if the **userID** and **itemID** combination exists. If it doesn't, it will insert a new row with that combination along with the item quantity. If the **userID** and **itemID** combination exists, then it will update that row with the newly calculated total item quantity.

The schema for **usercart** is as follows:

```
CREATE TABLE `usercart` (
  `userID` int(11) NOT NULL,
  `itemID` int(11) NOT NULL,
  `itemQuantity` int(11) NOT NULL
);
```

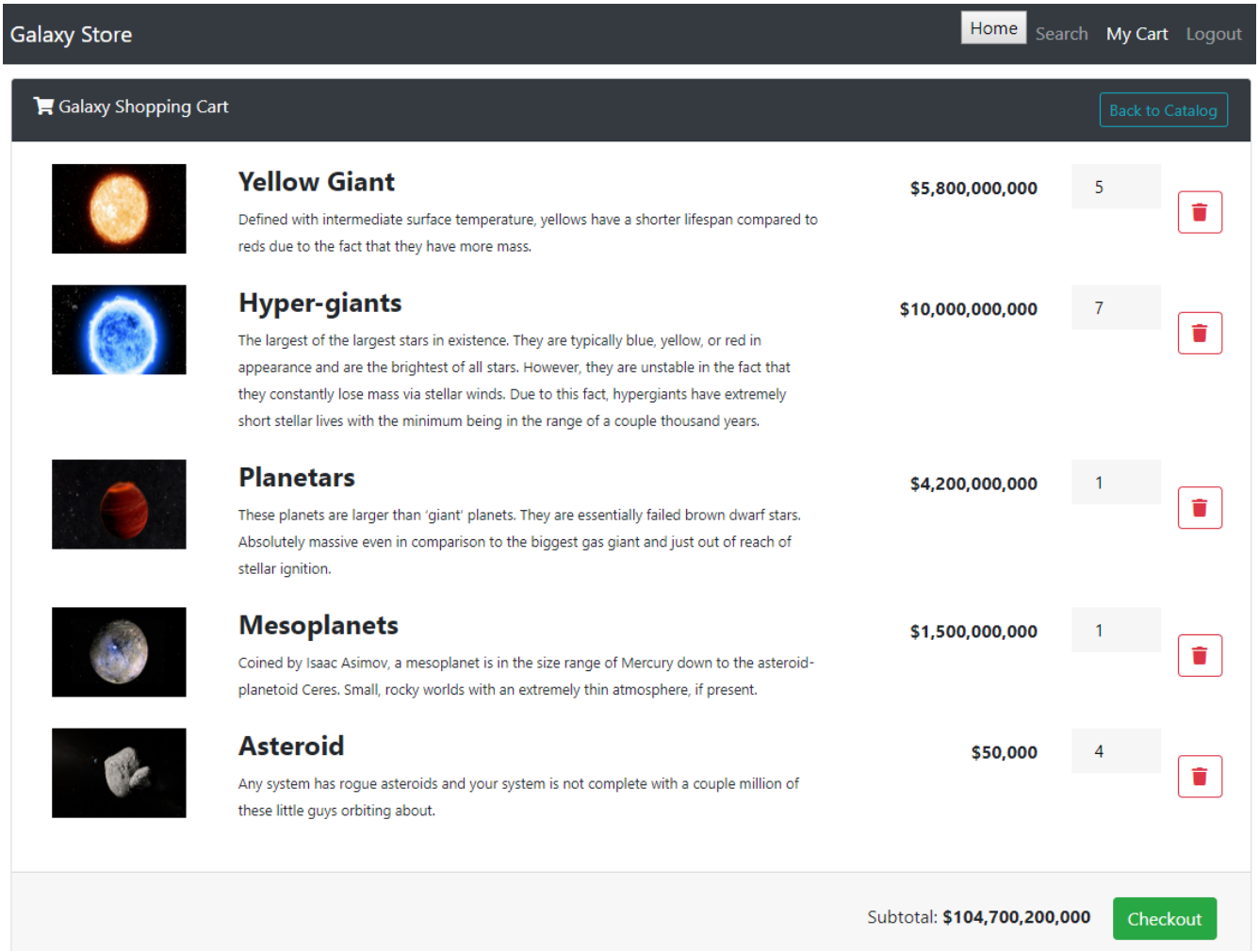
The cartAction API utilizes req.session.userID which is set upon login for the user's ID. The last two parameters, **itemID** and **itemQuantity** are passed as arguments and into the generated SQL query.

Search utilizes Bootstrap for design and a toast library (https://kamranahmed.info/toast) to display information to the user upon interacting with cart API.

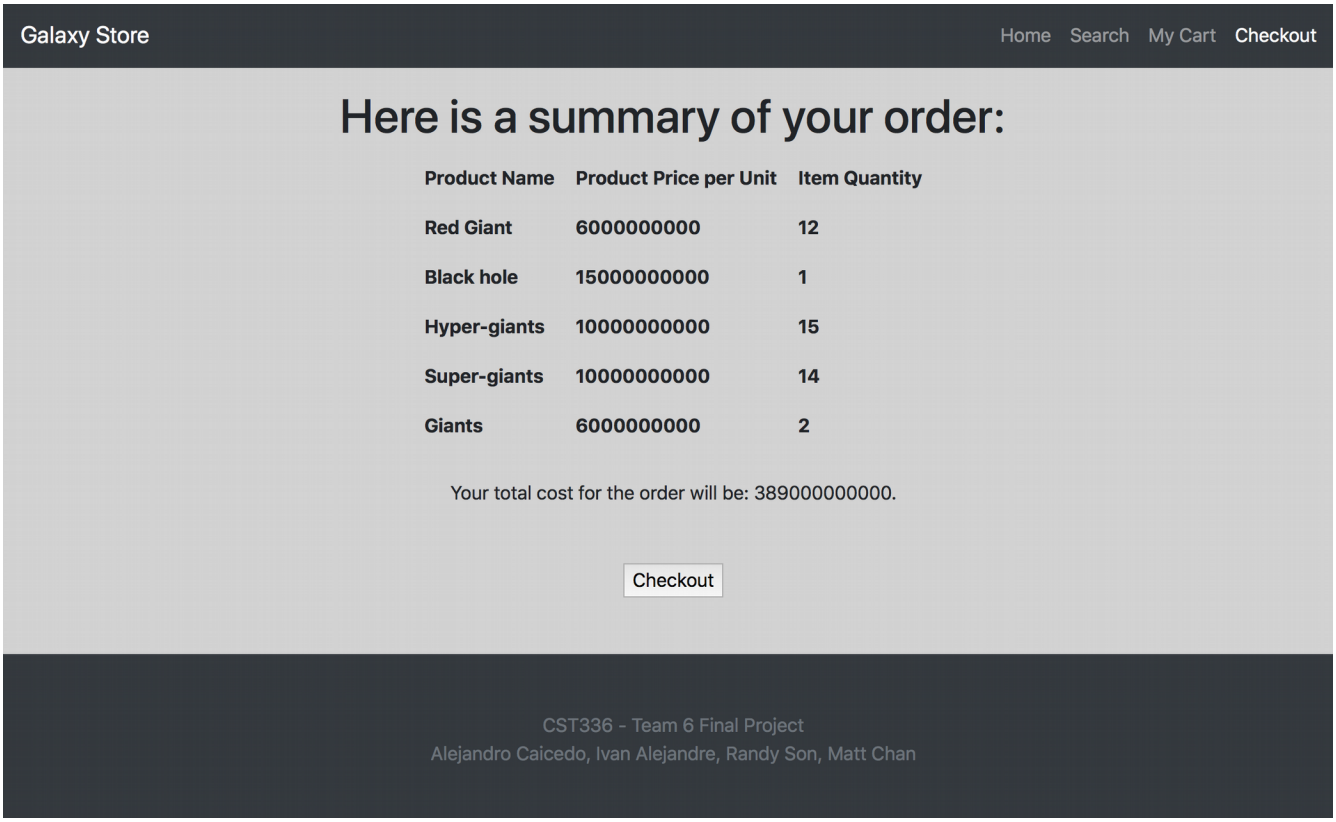


Shopping Cart

The shopping cart page displays the items for the user which the user had added into the table **usercart.** On this page, jQuery and AJAX calling the cartAction API handle the main functionality. Increasing or decreasing the quantity input updates the values in database. In addition, the trash can button deletes the **userID** and **itemID** combination from the **usercart** table. The subtotal at the bottom is calculated using a query that multiplies the quantity and the price of items together. AJAX calls this query and displays it. Two buttons exist: one to send the user back to the item catalog and the other to head to check out.

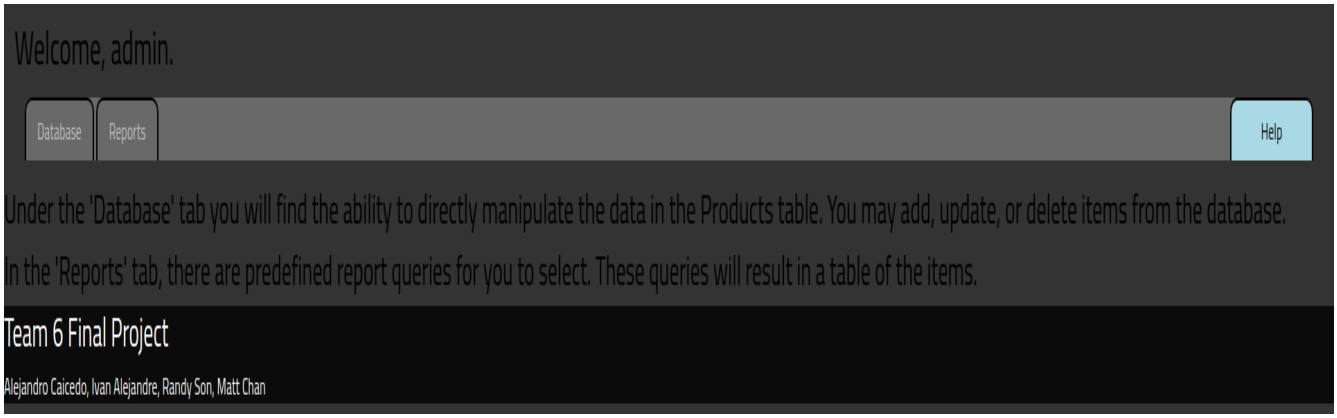


Once the user proceeds to checkout, the route will lead them to the checkout page. A connection to the database is formed as it utilizes req.sessions.userID to determine which user's information is pulled from the usercart table. It will then take the **product name**, the **price for each product**, and the **quantity of each product** and display them on a table for the user to review. The total cost for the order is also displayed for the user to review. The user then can either return to edit their shopping cart or checkout with the button on the bottom.



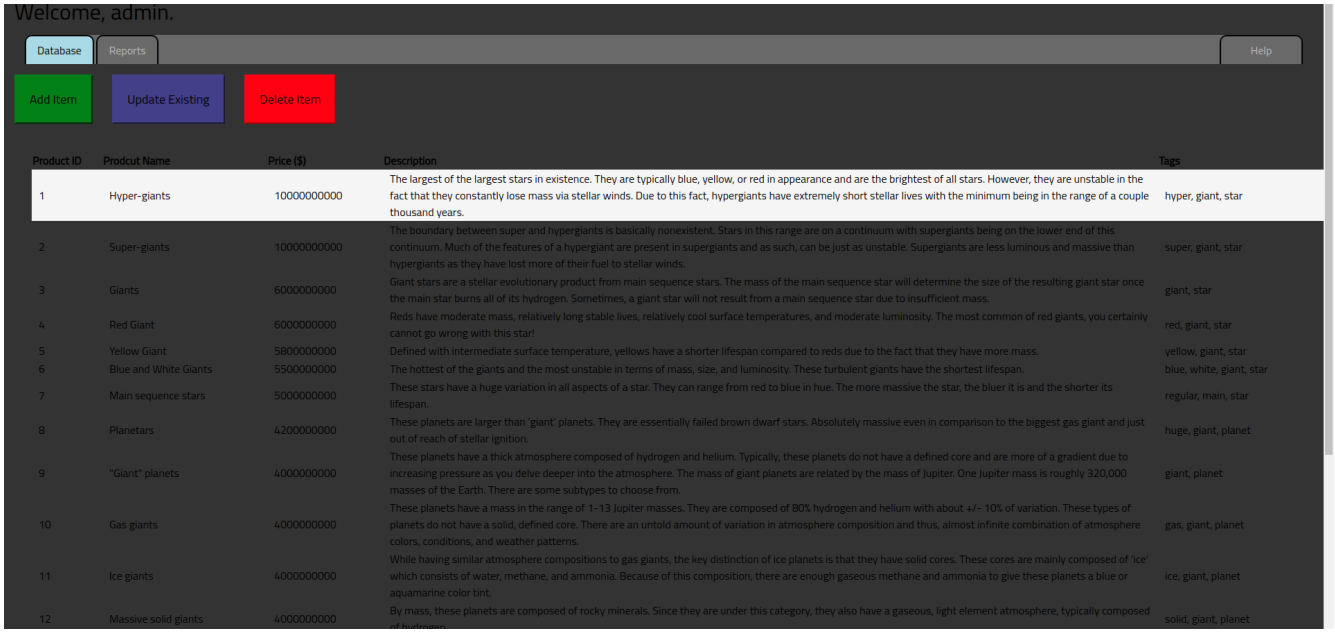
The checkout button creates a connection to transfer the data from **usercart** table to the **generaltransaction** and **detailedtransaction** tables. The generaltransaction table creates a new **transID** with the **userID** that is placing the order as well as the **timestamp** and the **total cost** of the order. It returns the **transID** to be used to transfer the information from the usercart to detailedtransactions. The items in the usercart is then deleted after it has been transferred and the user will be led to a page where it will display the transaction ID for the user to reference in the future and they can choose to return to the home page.

Admin Page



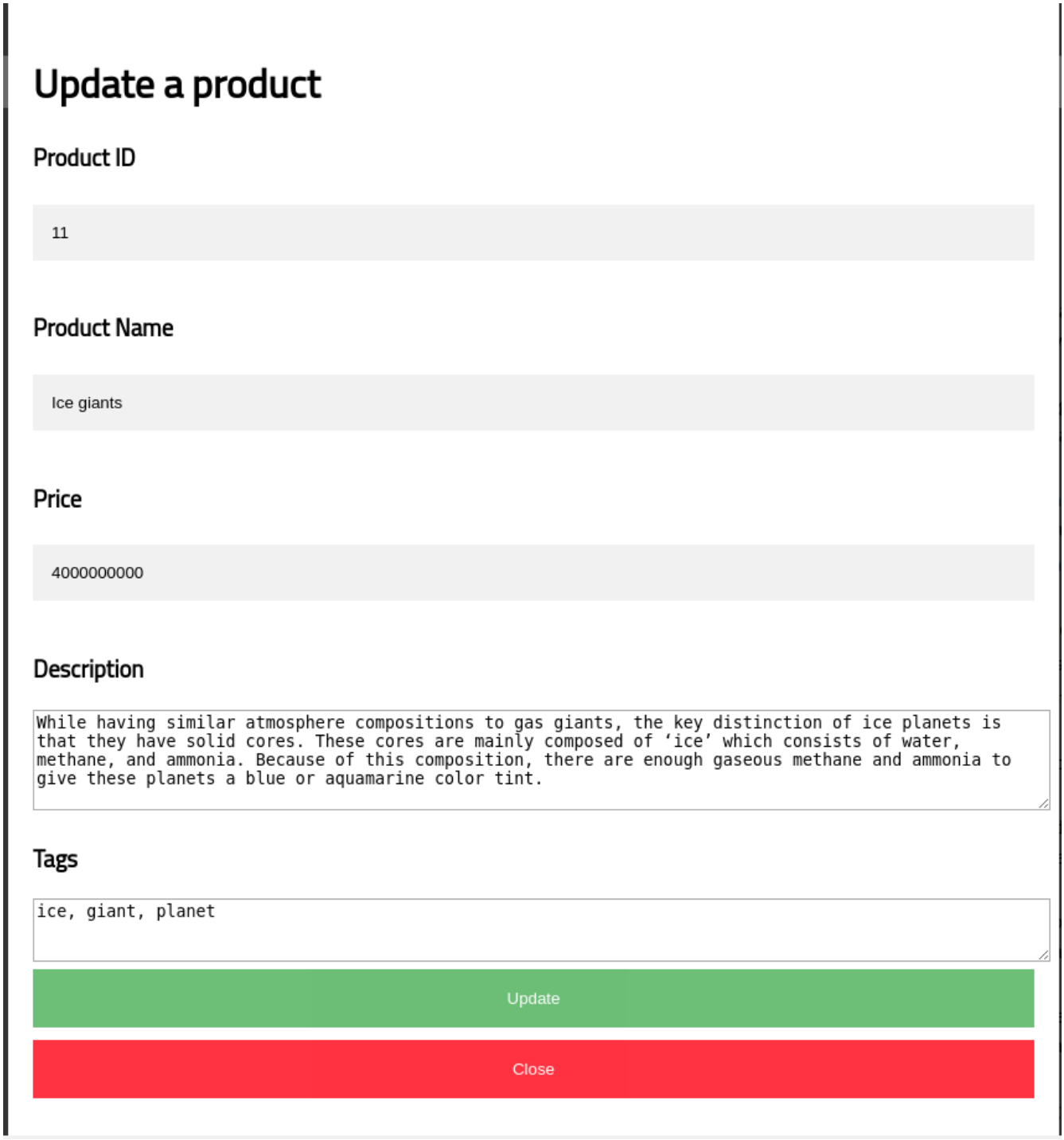
The admin page is where an administrator can directly manipulate the **product** table of the database. This page is only accessible if the **isAdmin** session boolean that was set upon login is true. The admin may add a new item, update any field of an existing item, or delete an item entirely. Upon first landing on the admin page, the user is greeted with a brief explanation of what each tab does.

The admin then has two choices. By clicking on the ‘Database’ tab, an AJAX call redraws a section of the page to contain three buttons (add, update, delete) and a table of all items in the **products** table.



Product ID	Prodcut Name	Price ($)	Description	Tags
1	Hyper-giants	10000000000	The largest of the largest stars in existence. They are typically blue, yellow, or red in appearance and are the brightest of all stars. However, they are unstable in the fact that they constantly lose mass via stellar winds. Due to this fact, hypergiants have extremely short stellar lives with the minimum being in the range of a couple thousand years.	hyper, giant, star
2	Super-giants	10000000000	The boundary between super and hypergiants is basically nonexistent. Stars in this range are on a continuum with supergiants being on the lower end of this continuum. Much of the features of a hypergiant are present in supergiants and as such, can be just as unstable. Supergiants are less luminous and massive than hypergiants as they have lost more of their fuel to stellar winds.	super, giant, star
3	Giants	6000000000	Giant stars are a stellar evolutionary product from main sequence stars. The mass of the main sequence star will determine the size of the resulting giant star once the main star burns all of its hydrogen. Sometimes, a giant star will not result from a main sequence star due to insufficient mass.	giant, star
4	Red Giant	6000000000	Reds have moderate mass, relatively long stable lives, relatively cool surface temperatures, and moderate luminosity. The most common of red giants, you certainly cannot go wrong with this star!	red, giant, star
5	Yellow Giant	5800000000	Defined with intermediate surface temperature, yellows have a shorter lifespan compared to reds due to the fact that they have more mass.	yellow, giant, star
6	Blue and White Giants	5500000000	The hottest of the giants and the most unstable in terms of mass, size, and luminosity. These turbulent giants have the shortest lifespan.	blue, white, giant, star
7	Main sequence stars	5000000000	These stars have a huge variation in all aspects of a star. They can range from red to blue in hue. The more massive the star, the bluer it is and the shorter its lifespan.	regular, main, star
8	Planetars	4200000000	These planets are larger than 'giant' planets. They are essentially failed brown dwarf stars. Absolutely massive even in comparison to the biggest gas giant and just out of reach of stellar ignition.	huge, giant, planet
9	"Giant" planets	4000000000	These planets have a thick atmosphere composed of hydrogen and helium. Typically, these planets do not have a defined core and are more of a gradient due to increasing pressure as you delve deeper into the atmosphere. The mass of giant planets are related by the mass of Jupiter. One Jupiter mass is roughly 320,000 masses of the Earth. There are some subtypes to choose from.	giant, planet
10	Gas giants	4000000000	These planets have a mass in the range of 1-13 Jupiter masses. They are composed of 80% hydrogen and helium with about +/- 10% of variation. These types of planets do not have a solid, defined core. There are an untold amount of variation in atmosphere composition and thus, almost infinite combination of atmosphere colors, conditions, and weather patterns.	gas, giant, planet
11	Ice giants	4000000000	While having similar atmosphere compositions to gas giants, the key distinction of ice planets is that they have solid cores. These cores are mainly composed of 'ice' which consists of water, methane, and ammonia. Because of this composition, there are enough gaseous methane and ammonia to give these planets a blue or aquamarine color tint.	ice, giant, planet
12	Massive solid giants	4000000000	By mass, these planets are composed of rocky minerals. Since they are under this category, they also have a gaseous, light element atmosphere, typically composed of hydrogen	solid, giant, planet

Here, the admin has full power to add an item to the database by filling all fields (it will not accept empty fields), update an existing item by first entering the product ID (and the existing fields will auto-populate), or delete an item by entering the product ID.



Under the Reports tab, the admin can select from three report options with each of them having some sort of specifier. After clicking on the search button, an AJAX call redraws the table section of the page to display the results. If, for some reason the return is empty, some text will appear in the table area stating an empty return.