**How to use**

1. Extract the files from the zip folder and add them to the htdocs folder in the XAMPP folder on your computer.
2. Open the XAMPP control panel and start Apache and MySQL. Click Admin in the MySQL row. Create a new database called rito\_pls then import the file from the zip folder called rito\_pls.sql.
3. In the rito\_pls database, go to privileges and add a user account with Username = rito\_pls, Host name = localhost, Password and Re-type = rito\_pls. Select the “Data” privileges and click go.
4. Open a browse and type localhost in the search bar to start using our site.

**Note**: Only an admin user can create another admin. Use the username “admin” and the password “admin” to login as an administrator.

**Context**

Our product is targeted towards people that play League of Legends. It allows those users to buy skins for the champions that they play. The way our product would make money is by the users purchasing RP to buy the skins they wanted.

**Using our product (Video Demo)**

When a new user visits our website, they can register for a new account and then login with those credentials.

The shop button allows the user to see all the skins that are available for sale. The user can sort these skins in alphabetical order, by price, by which skins are on sale or to show/hide owned skins from the list.

The search bar can be used to search for all champions with the typed in name.

The user can add skins to the cart with the add to cart button. When the user is satisfied, they can click the checkout button at the bottom to visit their cart and purchase the skins.

In the cart page, the buy button will only show if the user has enough RP to purchase all the skins in their cart so in this case, the user needs to buy RP first before they can buy.

The buy RP button at the top will bring the user to the page where they can purchase RP. We did not implement a way to use real money to purchase RP so for the time being, users can just buy RP for free.

Every page has a back to something button that will bring them to a previous page for easier navigation.

The user can remove unwanted items from their cart with the remove from cart button.

When the user is satisfied, they can click the buy button to purchase all skins in their cart which will take the necessary amount of RP from their account and it will redirect them to their home page which shows a list of all of their owned skins.

We can now see in the shop page that the Lux skin is owned and can be filtered properly with the checkbox.

The user can change their username as well as change their password. Both operations log the user out so they can log back in with their new information.

Finally, the user can log out of their account.

Employees can also log in but only admins can create a new employee account so there isn’t the same option on the login page.

From the home page, an employee can sort all the available skins by price or by which skins are on sale.

There is also a search bar that will show only the skins that contain the typed in name.

By clicking on the add a skin button, an employee can add a new skin to be sold. The new skin does not have a picture so the add a picture button becomes available for it. This button brings you to a page to add a picture for the skin it is associated with. Now that the skin has a picture, the button to add a picture disappears.

The details page shows the information of a skin and gives the option to delete the picture associated with the skin.

In the edit page, we can change information about a skin. For example, we can change the discount of a skin and then view the changes in the list of skins. The list of skins on sale now includes this new skin.

Employees can also give a user RP by clicking on the users button at the top of the home page. We can see that these changes are saved by logging back into the users account that we added RP to.

Employees can change their password the same way that users can but only admins can change an employee’s username or give that employee admin privileges.

Admins can delete an employee in the employee’s page, and they can also create a new employee from this page.

As with users, employees can log out of their account at any time.

We can see by logging into an employee account without admin privileges that the employees button disappears because normal employees don’t have the power to add, edit or remove other employees.

**Challenges**

Emmanuel: A lot of the problems I had were just with doing things for the first time. There were things I was doing wrong just because I didn’t know how to do them properly or didn’t understand why I was doing them a certain way. I could see my progress throughout the project though and would see myself not making those same mistakes later on in the project. I think something I struggled with the most was having the same page reload with different information each time because I needed to make sure I sent all the right information in the controller to account for the different scenarios.

Julien: The biggest challenge for me was figuring out how to use forms such as checkboxes to filter the data. The way our program currently works, you cannot select multiple checkboxes at the same time to filter data because the form refreshes the page. I spent countless hours trying to use cookies and session variables to save the state of the checkboxes, but without success. We had to settle for a functional but less sophisticated solution.