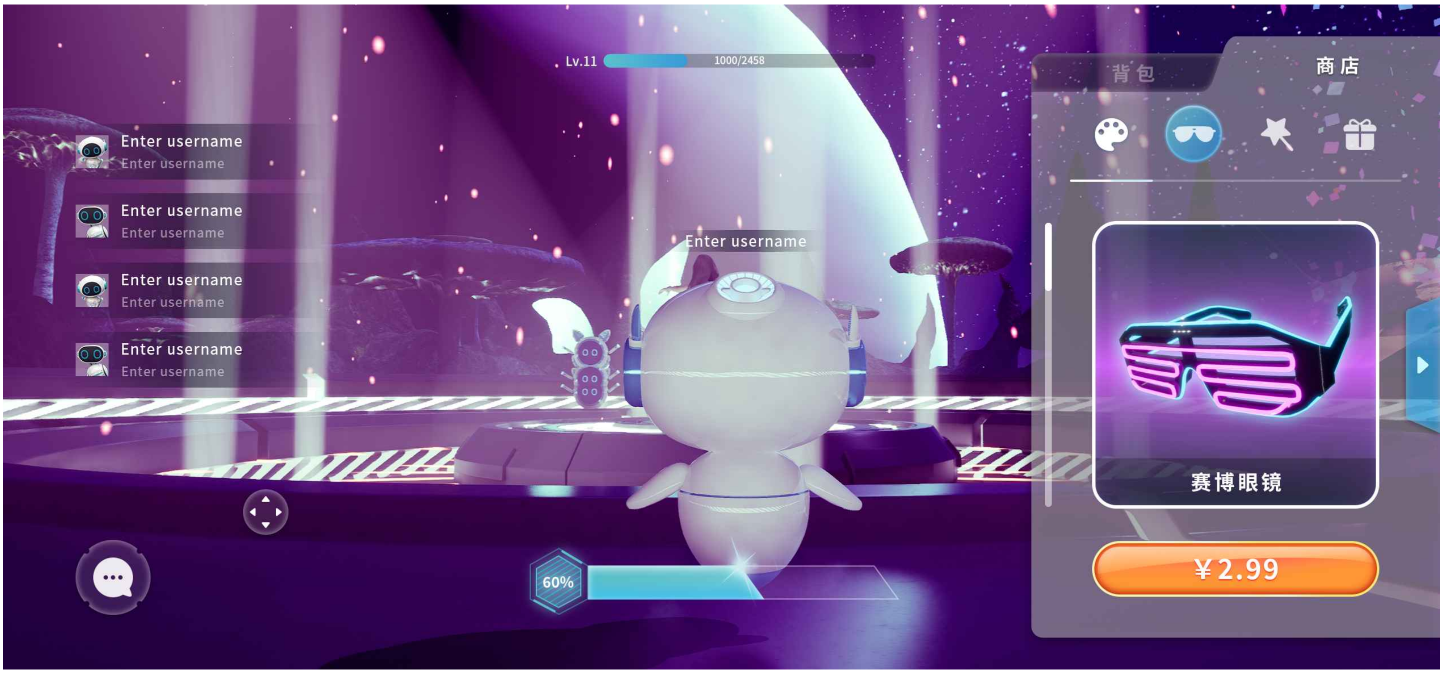
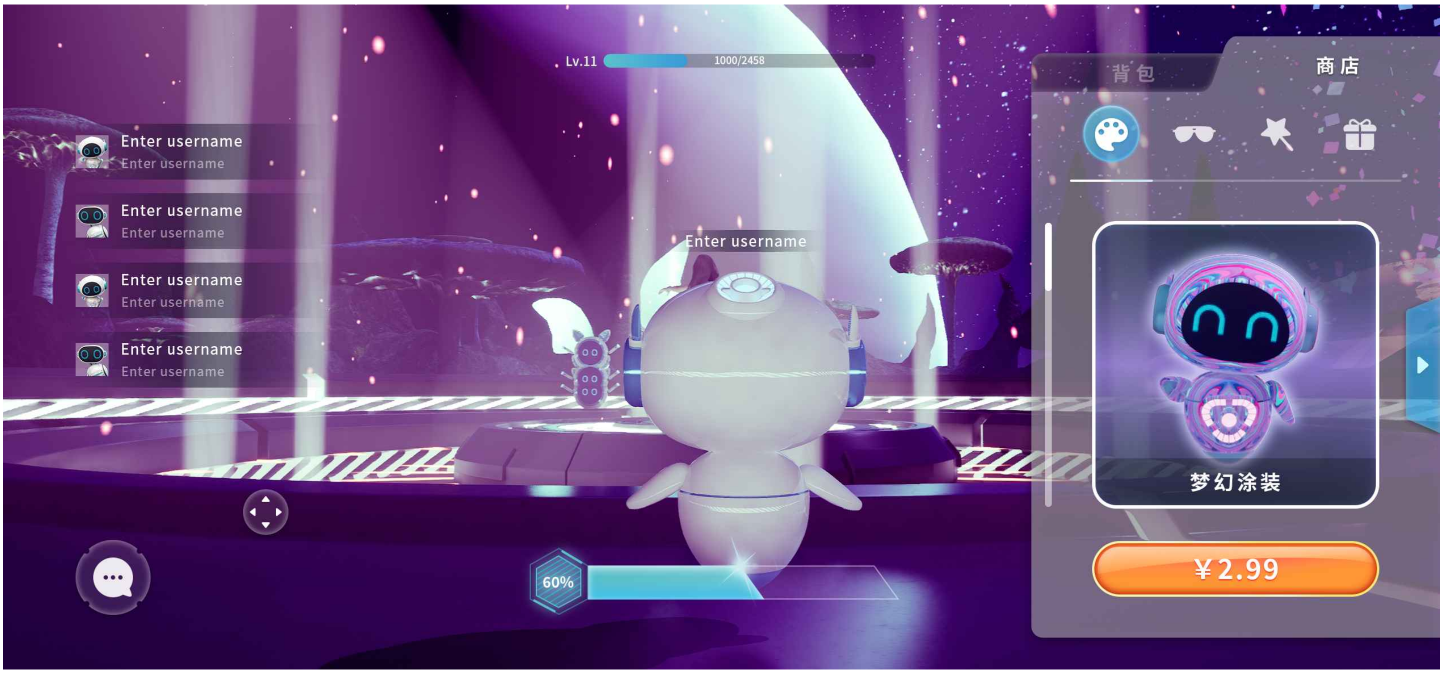
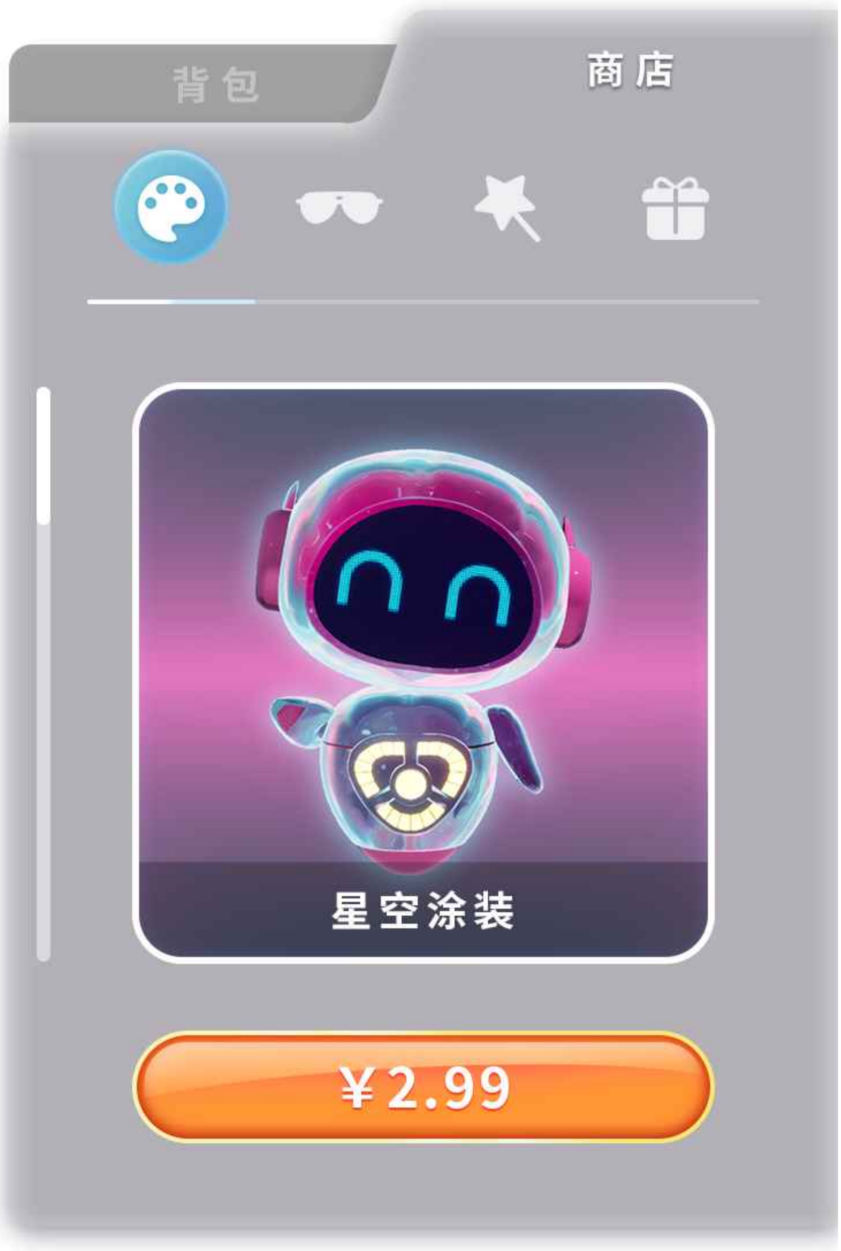
Front-end Shop/Bag Requirement







Task Description:

1. Make a shop/Bag module for an existing web page. The design is as above. Users could browser items in the shop, and after purchased, the items will show in their bag. Related assets such as icons, pictures for virtual items will be provided.
2. Back-end node.js, and mongoDB already setup.
3. We'll provide a remote desktop connection, where you could develop.
4. Need to realize these features:
   a. When opening the module, get the names, price and pictures for the items from the database. API provided
   b. Show the related products under that category, one item per page.
   c. Connect the payment api. We'll test on our end

Note:

1. The Shop/Bag tab is invisible on default, with a button on the right side of the page.
2. By clicking the button, the tab will be visible. Clicking the button again, the tab will disappear.

Folder for webpages:

C:\Users\Administrator\Desktop\MEgoStageWebServer\Master\SignallingWebServer7
C:\Users\Administrator\Desktop\MEgoStageWebServer\Master\SignallingWebServer7\Public\player.html

Need to add the shop/bag module on this page.

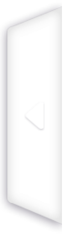
I have added something on the webpage, feel free to delete or modify.

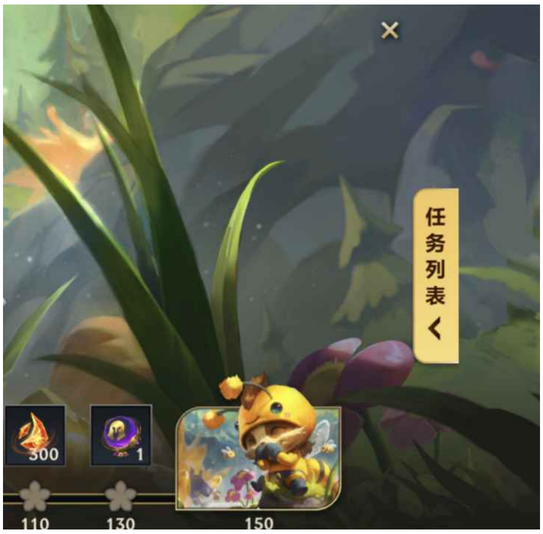
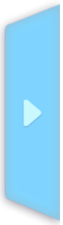
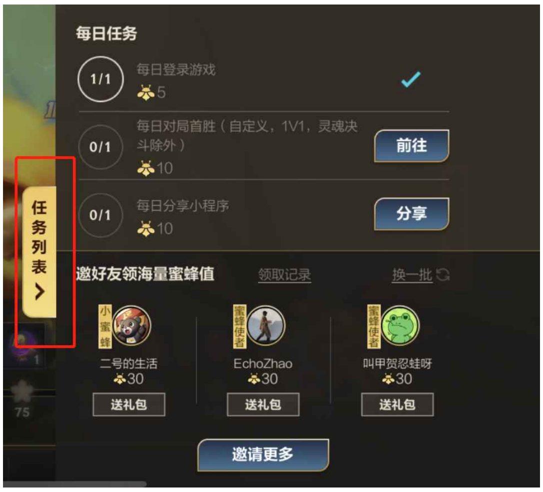
How to Test:

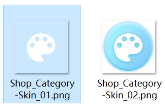
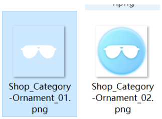
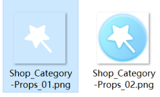
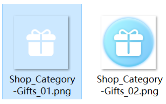
Access https://tests.mego-xr.com, then input phone number: 18621717615. I'll give you the verification code. Then you can login and get the cookie. You'll be direct to https://tests.mego-xr.com:457/ , where you can check player.html.

Read below if you are interested in the work.

Buttons explanation:

Button	Name	Feature	Design Requirement
	Shop_Side-Button_01	Click to make the Shop/Bag module visible Show the User-skin category's items by calling:	Add "背包/商城” above the arrow. Like shown below

			
	Shop_Side-Button_02	Click to make the Shop/Bag module invisible	**Make the button located at the LEFT side of the Shop/Bag module. Add "背包/商城” above the arrow.** Like shown below:  每日任务 1/1 每日登录游戏 5 0/1 每日对局首胜（自定义，1V1，灵魂决斗除外） 10 前往 0/1 每日分享小程序 10 分享 任务列表 邀好友领海量蜜蜂值 领取记录 换一批 小蜜蜂 二号的生活 30 蜜蜂使者 EchoZhao 30 蜜蜂使者 叫甲贺忍蛙呀 30 送礼包 送礼包 送礼包 邀请更多
	Shop-"商店”	Click to show the Shop Tab Show the Shop-skin category's items by calling: getShopList(categoryId)	
	Bag-“背包”	Click to show the Shop Tab Show the Bag-skin category's items by	

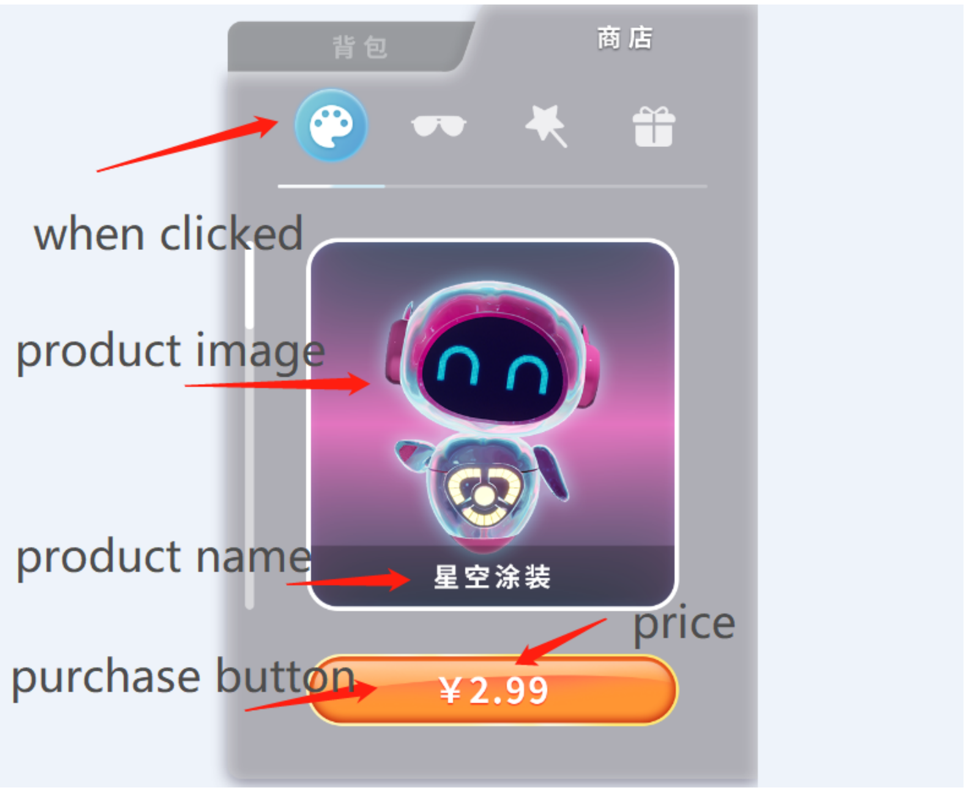
		calling: getShopList(categoryId)	
	Shop_Category-Skin_01 Shop_Category-Skin_02	When in shop tab, Get the Shop-skin category's items by calling: getShopList(categoryId) When in bag tab, Get the Bag-skin	when the tab is not selected, show 01; when the tab is selected, show 02
	Shop_Category-Ornament_01 Shop_Category-Ornament_02	When in shop tab, Get the Shop-ornament category's items by calling: getShopList(categoryId) When in bag tab, Get the Bag-ornament	when the tab is not selected, show 01; when the tab is selected, show 02
	Shop_Category-Props_01 Shop_Category-Props_02	When in shop tab, Get the Shop-prop category's items by calling: getShopList(categoryId) When in bag tab, Get the Bag-prop	when the tab is not selected, show 01; when the tab is selected, show 02
	Shop_Category-Gifts_01 Shop_Category-Gifts_02	When in shop tab, Get the Shop-gift category's items by calling: getShopList(categoryId) When in bag tab, Get the Bag-gift	when the tab is not selected, show 01; when the tab is selected, show 02
	Shop_Purchase-Button_01 Shop_Purchase-Button_02	When in shop tab, button shows the price. Purchase api: payover	
	Shop_Purchase-Button_03	When in shop tab, after clicking the purchase button. Meaning processing the payment. Button shows"获取中 “”	

	Shop_Purchase-Button_04	When in shop tab, if a product's price is 0, user can get it for free, button shows "免费获取”	
	Shop_Purchased-Button_01	When in shop tab, if a product's price is not 0, and got purchased, button shows "已购买"	
	Shop_Purchased-Button_02	When in shop tab, if a product's price is 0, and got accquried by the user, button shows "已获取"	
	Shop_Equip-Button_01	In Bag tab, click the button to equip/use the item. Button shows"装备” Related API: Use	
	Shop_Equip-Button_02	In Bag tab, click the button to take off the item. Button shows“卸下” Related API: Use	

Key files for development:

File	Description
Frontend	
\Desktop\MEgoStageWebServer\Master\SignallingWebServer7\Public\player.html	Web page that the bag/shop module should be on.
\Desktop\MEgoStageWebServer\Master\SignallingWebServer7\scripts\ShopAPI.js	Javascript file of the front-end Shop/bag module
Backend	
Desktop\MEgoStageWebServer\Master\node\controller\sys	All the back-end APIs for the Shop/Bag module
Desktop\MEgoStageWebServer\Master\node\controller\sys\bag	Return all the things the user has purchased

\GetUserBag.js	
Desktop\MEgoStageWebServer\Master\node\controller\sys\bag\use.js	When user equip or use a skin, ornament, prop of gift.
Desktop\MEgoStageWebServer\Master\node\controller\sys\shop\GetShoplist.js	Return the data(name, price, picture) related with an item in certain category. For example, when ask for skin, it'll return all the things in the skins category.
middlewares\auth.js	To authenticate based on the token saved in the cookie
middlewares\mongodb.js	To do all database related operations
middlewares\redis-client.js	To issue and authenticate token
\Desktop\MEgoStageWebServer\Master\node\routes	Routes for API



Communications with the backend:

After obtaining the wear(IsUse) information of the skins, accessories, and props, execute the following code for all the worn items, where 'item' refers to the corresponding item.

```
let descriptor = {
              item: xxxx,
              isEquipped: true,
          };
emitUIInteraction(descriptor);
```

When users enter the 'skins' category in their backpack, they can click the 'Equip' button to apply that skin to their character. After clicking, the 'Equip' button changes to 'Unequip'. [The database needs to be updated accordingly.] After confirming that it's equipped, execute the following code:

```
let descriptor = {
              item: xxxx,
              isEquipped: true,
          };
emitUIInteraction(descriptor);
```

If the user clicks 'Unequip', [the database needs to be updated accordingly]. After confirming that it has been unequipped, execute the following code:"

```
 let descriptor = {
               item: xxxx,
               isEquipped: false,
           };
emitUIInteraction(descriptor);
```