

**Table of Contents:**

Login

Main Menu

My Items

Searching for Items

View Item

Propose Swap

Accept or Reject Swaps

Rate Swaps

Swap History

**Phase 2 Abstract Code w/SQL | CS 6400 - Spring 2022 | Team 103**

[Update My Information](#)

[Login](#)

Abstract Code

## Phase 2 Abstract Code w/SQL | CS 6400 - Spring 2022 | Team 103

### Login

Sign In

---

Email/phone number

Password

Login

---

New User

Register

- User Enters Email , Password
- If the data validation is successful, then click the Login button

## Phase 2 Abstract Code w/SQL | CS 6400 - Spring 2022 | Team 103

```
Select password from user where Email = '$Email';
```

- If user record not found then show message “User not found” go back to Login form
- If user record found but User.password!= '\$Password' then show message “Incorrect password” go back to Login form
- Else Go to Main Menu Page
- Email will be stored in '\$Email' variable
- Show Register button
- On click will go to Register User module

[User Registration](#)

Abstract Code

## Phase 2 Abstract Code w/SQL | CS 6400 - Spring 2022 | Team 103

User registration

Registration

Email

gburdell@gatech.edu

Nickname

YellowJacket1

Password

\*\*\*\*\*

City

Atlanta

First Name

George

State

Georgia

Last Name

Burdell

Postal Code

30332

Phone number (optional)

404-555-1212

☐ Show phone number in swaps

Type

Home

Work

Mobile

Register

- Email Textbox: User will be allowed to enter email address.All the necessary Data Validation will be in place
- Nickname Textbox: User will be allowed to enter Nickname.All the necessary Data Validation will be in place
- Password Textbox: User will be allowed to enter Password.All the necessary Data Validation will be in place

## Phase 2 Abstract Code w/SQL | CS 6400 - Spring 2022 | Team 103

- First Name Textbox: User will be allowed to enter First Name. All the necessary Data Validation will be in place
- Last Name Textbox: User will be allowed to enter Last Name. All the necessary Data Validation will be in place
- Postal Code Textbox: User will be allowed to enter Postal code. All the necessary Data Validation will be in place
- City and State will be populated based on the verified postal code

```
Select City,State from address where postalcode='$Postalcode';
```

- Phone Number Textbox: User will be allowed to enter Phone Number. All the necessary Data Validation will be in place
- Show phone number in swaps checkBox: User will be allowed to check the checkbox. All the necessary Data Validation will be in place
- Type drop down box: User will be allowed to select from the set of values for the phone type. This dropdown is enabled if there is a legit phone number filled in the phone number text box. Below query is executed for fetching the values.
- Postal data validation for each field the values will be stored in variables.

## Phase 2 Abstract Code w/SQL | CS 6400 - Spring 2022 | Team 103

- The register button is enabled only if all the above fields are validated. On click below insert query will be executed

```
Insert into user (Email,Password,FirstName,LastName,Nickname,Phonenumber,Postalcode)
values ('$Email', '$Password', '$FirstName', '$LastName', '$NickName', '$Phonenumber',
'$Postalcode');
```

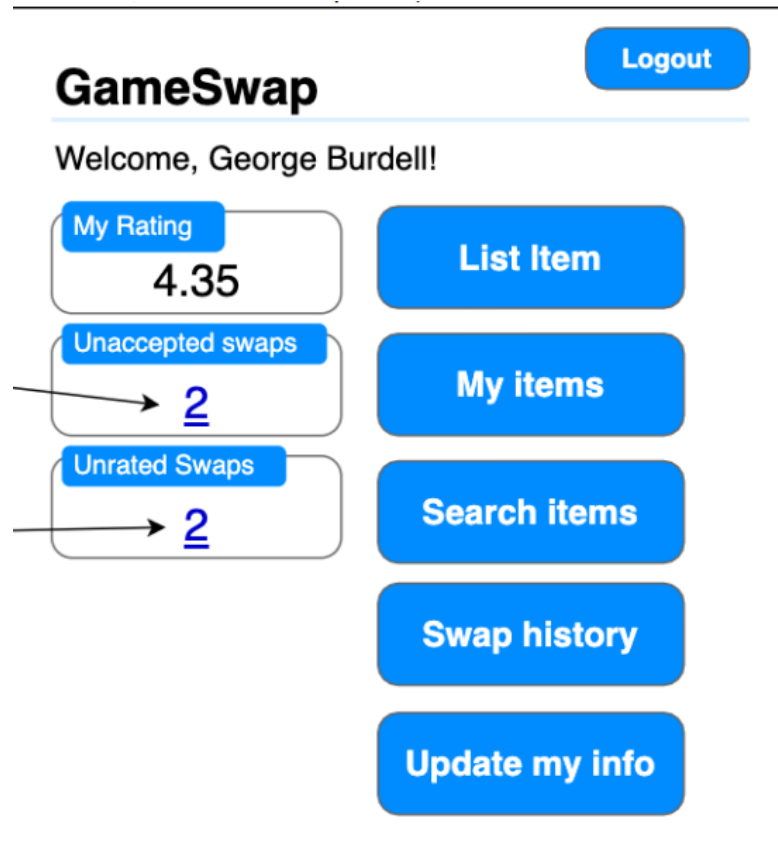
```
Insert into phone(Number,Type,ShareFlag) values ('$Phonenumber', '$Type', '$ShareFlg');
```

- If Insert Statement fails then show message “Registration Failed”
- Else Go to “Login Page”

[Main Menu](#)

Abstract Code

## Phase 2 Abstract Code w/SQL | CS 6400 - Spring 2022 | Team 103



- Show ListItem, My Items, Search Items, Swap History, Update my info, My Rating, Unaccepted Swaps, Unrated Swaps, Logout
- Upon
  1. Click ListItem to go ListItem module
  2. Click MyItems to go MyItems module
  3. Click Search Items to go Search Items module



## Phase 2 Abstract Code w/SQL | CS 6400 - Spring 2022 | Team 103

4. Click Swap History to go Swap History module
5. Click Update my Info to go Edit/View Profile
6. My rating will display your aggregated rating. Below query is executed  

`Select MyRating from user where Email='$Email';`
7. Unaccepted Swaps will show count of unaccepted swaps as a hyperlink. On click will go Swap History module
  - For getting the Unaccepted swaps below select query is executed  

`Select UnacceptedSwaps from user where Email='$Email';`
8. Unrated Swaps will show count of unrated swaps as a hyperlink. On click will go Swap History module.
  - For getting the Unrated swaps below select query is executed  

`Select UnratedSwaps from user where Email='$Email';`
9. Click Logout to go the Login module

[List Item](#)

## Phase 2 Abstract Code w/SQL | CS 6400 - Spring 2022 | Team 103

should be assigned by the system and a success message showing the item number should be displayed to the user.

### New Item Listing

Game type  
Jigsaw Puzzle ▼

Title  
"Central Park" by Michael Storrings

Condition  
Like New ▼

Piece count  
1000 ▲▼

Description  
Only assembled once. No missing pieces! Has beautiful glitter effect.

List Item

### Success

Your item has been listed!  
Your item number is 42.

OK

### Abstract Code

- Show Game Type Drop box. The values will be prepopulated by the code
- Based on the selection of the GameType further fields will be shown

## Phase 2 Abstract Code w/SQL | CS 6400 - Spring 2022 | Team 103

- **Jigsaw Puzzle**

- 1.Show the Title Text Box
- 2.Show Condition Drop Box. The values will be prepopulated by the code
- 3.Show the Piece Count
- 4.Show the Description Text Box ( With fixed number of characters)
- 5.Enable the List Item Button and Once clicked below Insert query will be executed

```
Insert into item (Email,Title,Condition,Description,Itemtype) values  
('$Email', '$Title', '$Condition', '$Description', '$ItemType');
```

The Itemnumber is incremental number which will be populated for each insert into Item table.

```
Select max(ItemNumber) ItemNumber from item where Title='$Title'  
and ItemType='$ItemType'and Email='$Email';
```

Above select query output is used to get the Itemnumber for the current transaction and stored into '\$ItemNumber' .

```
Insert into itemtype (ItemNumber,Name,PieceCount) values  
('$ItemNumber' , '$ItemType', '$PieceCount') ;
```

## Phase 2 Abstract Code w/SQL | CS 6400 - Spring 2022 | Team 103

- **BoardGame**

- 1.Show the Title Text Box
- 2.Show Condition Drop Box. The values will be prepopulated by the code
- 3.Show the Description Text Box ( With fixed number of characters)
- 4.Enable the List Item Button and Once clicked below Insert query will be executed

```
Insert into item (Email,Title,Condition,Description,Itemtype) values  
('$Email', '$Title', '$Condition', '$Description', '$ItemType');
```

The Itemnumber is incremental number which will be populated for each insert into Item table.

```
Select max(ItemNumber) ItemNumber from item where Title='$Title'  
and ItemType='$ItemType'and Email='$Email';
```

Above select query output is used to get the Itemnumber for the current transaction and stored into '\$ItemNumber' .

```
Insert into itemtype (ItemNumber,Name) values ('$ItemNumber' ,  
'$ItemType') ;
```

## Phase 2 Abstract Code w/SQL | CS 6400 - Spring 2022 | Team 103

- **CardGame**

- 1.Show the Title Text Box
- 2.Show Condition Drop Box. The values will be prepopulated by the code
- 3.Show the Description Text Box ( With fixed number of characters)
- 4.Enable the List Item Button and Once clicked below Insert query will be executed

```
Insert into item (Email,Title,Condition,Description,Itemtype) values  
('$Email', '$Title', '$Condition', '$Description', '$ItemType');
```

The Itemnumber is incremental number which will be populated for each insert into Item table.

```
Select max(ItemNumber) ItemNumber from item where Title='$Title'  
and ItemType='$ItemType'and Email='$Email';
```

Above select query output is used to get the Itemnumber for the current transaction and stored into '\$ItemNumber' .

```
Insert into itemtype (ItemNumber,Name) values ('$ItemNumber' ,  
'$ItemType') ;
```

## Phase 2 Abstract Code w/SQL | CS 6400 - Spring 2022 | Team 103

- **VideoGame**

- 1.Show the Title Text Box
- 2.Show Condition Drop Box. The values will be prepopulated by the code
- 3.Show the Description Text Box ( With fixed number of characters)
4. Show Platform DropBox. The values will be prepopulated by the code
5. Show Media DropBox. The values will be prepopulated by the code
- 6.Enable the List Item Button and Once clicked below Insert query will be executed

```
Insert into item (Email,Title,Condition,Description,Itemtype) values  
('$Email', '$Title', '$Condition', '$Description', '$ItemType');
```

The Itemnumber is incremental number which will be populated for each insert into Item table.

```
Select max(ItemNumber) ItemNumber from item where Title='$Title'  
and ItemType='$ItemType'and Email='$Email';
```

## Phase 2 Abstract Code w/SQL | CS 6400 - Spring 2022 | Team 103

Above select query output is used to get the Itemnumber for the current transaction and stored into '\$ItemNumber' .

```
Insert into itemtype (ItemNumber,Name,Platform,Media) values  
('$ItemNumber' , '$ItemType' , '$Platform' , '$Media') ;
```

- **ComputerGame**

- 1.Show the Title Text Box
- 2.Show Condition Drop Box. The values will be prepopulated by the code
- 3.Show the Description Text Box ( With fixed number of characters)
4. Show Platform DropBox. The values will be prepopulated by the code
- 5.Enable the List Item Button and Once clicked below Insert query will be executed

```
Insert into item (Email,Title,Condition,Description,Itemtype) values  
('$Email', '$Title', '$Condition', '$Description', '$ItemType');
```

The Itemnumber is incremental number which will be populated for each insert into Item table.

```
Select max(ItemNumber) ItemNumber from item where Title='$Title'  
and ItemType='$ItemType'and Email='$Email';
```

## Phase 2 Abstract Code w/SQL | CS 6400 - Spring 2022 | Team 103

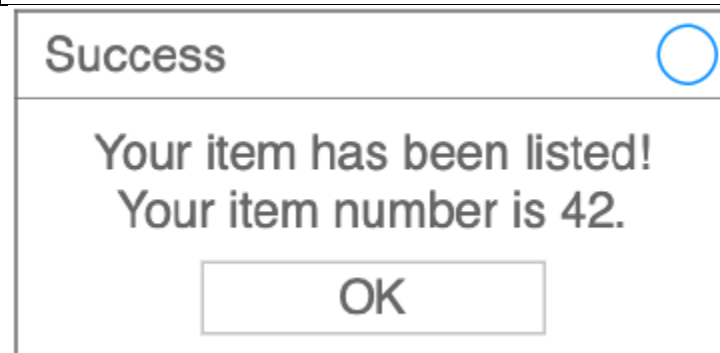
Above select query output is used to get the Itemnumber for the current transaction and stored into '\$ItemNumber' .

```
Insert into itemtype (ItemNumber,Name,Platform) values  
('$ItemNumber' , '$ItemType', '$Platform') ;
```

On Completion of Transaction below message might be displayed

- On Success – Below message box will be shown with Item Number. For fetching the item number below query is executed. The user will be redirected to the Main Menu.

```
Select max(ItemNumber) ItemNumber from item where Title='$Title'  
and ItemType='$ItemType'and Email='$Email';
```





## Phase 2 Abstract Code w/SQL | CS 6400 - Spring 2022 | Team 103

- On Failure – Message box will be shown “Item Listing Failed”

### My Items

Item counts

Board games	Card games	Computer games	Jigsaw puzzles	Video games	Total
1	1	1	1	1	5

Abstract Code

- Show Item Counts for the User. Below select query is executed for it

```
Select sum(BoardGames) BoardGames, sum (CardsGames)
CardsGames,sum(ComputerGames)
ComputerGames,sum(JigSawPuzzles)
JigSawPuzzles,sum(VideoGames)+sum(BoardGames) +sum
(CardsGames)+sum(ComputerGames)+sum(JigSawPuzzles)
+sum(VideoGames) Total from
```

## Phase 2 Abstract Code w/SQL | CS 6400 - Spring 2022 | Team 103

```
(Select case when ItemType='Board Game' then 1 else 0 end
BoardGames,
        case when ItemType='Card Game' then 1 else 0 end
CardsGames,
        case when ItemType='Computer Game' then 1 else 0 end
ComputerGames ,
case when ItemType='JigSaw Puzzle' then 1 else 0 end
JigSawPuzzles,
case when ItemType='Video Game ' then 1 else 0 end VideoGames
from item I join itemtype ty on I.ItemType=ty.Itemtype and
I.ItemNumber=ty. ItemNumber where I.Email='$Email') ;
```

- Show My Item Details by executing below select query

My Items

Item #	Game type	Title	Condition	Description	
23	Video game	Tetris	Lightly used		<a href="#">Detail</a>
34	Board game	Monopoly	Damaged/Missing parts	We used to play this game heavily, and started losing some of the figurines, the cards, and finally...	<a href="#">Detail</a>
106	Card game	UNO	Mint	Never opened!	<a href="#">Detail</a>
250	Jigsaw puzzle	Sushi	Like New		<a href="#">Detail</a>
345	Computer game	Myst	Moderately used	CD has some slight scratches but it's a classic	<a href="#">Detail</a>

## Phase 2 Abstract Code w/SQL | CS 6400 - Spring 2022 | Team 103

```
Select I.ItemNUmber, I.Itemtype GameType, I.Title, I.Condition,  
I.Description,ty.Plaform,ty.Media,ty.PieceCount,ty.Description  
from item I join itemtype ty on I.ItemType=ty.Itemtype and I.ItemNumber=ty.  
ItemNumber where Email ='${Email}';
```

- **The Details button for every Item will consume the data returned from the below query**

```
Select I.ItemNUmber, I.Itemtype GameType, I.Title, I.Condition,  
I.Description,ty.Platform,ty.Media,ty.PieceCount,ty.Description  
from item I join itemtype ty on I.ItemType=ty.Itemtype and I.ItemNumber=ty.  
ItemNumber where Email ='${Email}' and I.ItemNumber=${ItemNumber};
```

- **The Description restricting to 100 chars will be done in the code**

## Phase 2 Abstract Code w/SQL | CS 6400 - Spring 2022 | Team 103

### Searching for Items

---

Search

☐ By keyword:

☐ In my postal code

☐ Within  miles of me

☐ In postal code:

Search!

---

### Abstract Code

- **By Keyword:**
  1. Show the “By keyword” radiobutton
  2. Once the radiobutton is selected then enable the textbox
  3. On click of the “Search” button, below query will be executed

```
Select U.Email, I.ItemNUmber, I.Itemtype GameType, I.Title, I.Condition,  
I.Description,ty.Platform,ty.Media,ty.PieceCount,ty.Description,addr.Postal  
Code,addr.longitude,addr.latitude  
from user U join item I on U.Email=I.Email join itemtype ty on  
I.ItemType=ty.Itemtype and I.ItemNumber=ty. ItemNumber;  
join address addr on I.PostalCode= U.PostalCode
```

## Phase 2 Abstract Code w/SQL | CS 6400 - Spring 2022 | Team 103

```
where I.Email=$Email where U.Email <>'$Email' and (lower(Title) like  
'%lower($Keyword)%' or lower(Description) like '%lower($Keyword)%') ;
```

### Below query will provide the logged user address details

```
Select U.User, addr.PostalCode,addr.longitude,addr.latitude  
from user U join address addr on I.PostalCode= U.PostalCode  
where I.Email='$Email' where U.Email ='$Email';
```

### /\* Logged User details\*/

Using the longitude and latitude of the logged user and the searched item  
Owner distance between is calculated

4. On execution of the query
  - a. For return of records below screen will be displayed

## Phase 2 Abstract Code w/SQL | CS 6400 - Spring 2022 | Team 103

Search results: keyword "tech"

Item #	Game type	Title	Condition	Description	Distance	
1885	Jigsaw puzzle	Georgia Tech campus	Mint		0.0	<a href="#">Detail</a>
300	Video game	Wolfenstein: The New Order	Like New	Utilizing the idTech 5 engine, this game is an incredible sequel to the previous games in the serie...	3.5	<a href="#">Detail</a>
124	Board game	Hungry Hungry Hippos	Damaged/Missing parts	This is such a fun game from my childhood. It has small pieces so it is not recommended if you hav...	7.1	<a href="#">Detail</a>
927	Computer game	BattleTech	Moderately used		7.1	<a href="#">Detail</a>

b. For no rows returned Message will be shown “No results found”

- **In My postal code:**

1. Show the “In My postal code” radiobutton
2. On click of the “Search” button, below query will be executed

```
Select U.Email, I.ItemNumber, I.Itemtype GameType, I.Title, I.Condition,
I.Description,ty.Platform,ty.Media,ty.PieceCount,ty.Description,addr.Postal
Code,addr.longitude,addr.latitude
from user U join item I on U.Email=I.Email join itemType ty on
I.ItemType=ty.Itemtype and I.ItemNumber=ty.ItemNumber
join address addr on I.PostalCode= U.PostalCode
where I.Email='$Email' where U.Email <>' $Email' and
U.postalCode='$PostalCode' /*Logged in User Postal Code*/ ;
```

Below query will provide the logged user address details

## Phase 2 Abstract Code w/SQL | CS 6400 - Spring 2022 | Team 103

```
Select U.User, addr.PostalCode,addr.longitude,addr.latitude  
from user U join address addr on I.PostalCode= U.PostalCode  
where I.Email='$Email' where U.Email ='$Email';
```

**/\* Logged User details\*/**

**Using the longitude and latitude of the logged user and the searched item  
Owner distance between is calculated**

3. On execution of the query
  - a.For return of records below screen will be displayed

Search results: keyword "tech"

Item #	Game type	Title	Condition	Description	Distance	
1885	Jigsaw puzzle	Georgia Tech campus	Mint		0.0	<a href="#">Detail</a>
300	Video game	Wolfenstein: The New Order	Like New	Utilizing the idTech 5 engine, this game is an incredible sequel to the previous games in the serie...	3.5	<a href="#">Detail</a>
124	Board game	Hungry Hungry Hippos	Damaged/Missing parts	This is such a fun game from my childhood. It has small pieces so it is not recommended if you hav...	7.1	<a href="#">Detail</a>
927	Computer game	BattleTech	Moderately used		7.1	<a href="#">Detail</a>

- b.For no rows returned Message will be shown “No results found”

## Phase 2 Abstract Code w/SQL | CS 6400 - Spring 2022 | Team 103

- **Within X miles of me:**
  1. Show the “Within the X miles” radiobutton
  2. Once the radiobutton is selected then enable the mile selector
  3. On click of the “Search” button, below query will be executed

```
Select U.Email, I.ItemNUmber, I.Itemtype GameType, I.Title, I.Condition,  
I.Description,ty.Platform,ty.Media,ty.PieceCount,ty.Description,addr.PostalCode,addr.longitude,  
addr.latitude from user U join item I on U.Email=I.Email join itemtype ty on  
I.ItemType=ty.Itemtype and I.ItemNumber=ty.ItemNumber  
join address addr on I.PostalCode= U.PostalCode;  
where I.Email='$Email' where U.Email <>'$Email';
```

### **Below query will provide the logged user address details**

```
Select U.User, addr.PostalCode,addr.longitude,addr.latitude  
from user U join address addr on I.PostalCode= U.PostalCode  
where I.Email='$Email' where U.Email ='$Email' ;
```

**/\* Logged In User details\*/**

**Using the longitude and latitude of the logged in user and the searched item Owner distance between is calculated**



## Phase 2 Abstract Code w/SQL | CS 6400 - Spring 2022 | Team 103

4. The distance of each user with the logged user is calculated and then compared with the selected miles and only items whose distance with the logged in user is within the selected miles
5. On execution of the query
  - a. For return of records below screen will be displayed

Search results: keyword "tech"

Item #	Game type	Title	Condition	Description	Distance	
1885	Jigsaw puzzle	Georgia Tech campus	Mint		0.0	<a href="#">Detail</a>
300	Video game	Wolfenstein: The New Order	Like New	Utilizing the idTech 5 engine, this game is an incredible sequel to the previous games in the serie...	3.5	<a href="#">Detail</a>
124	Board game	Hungry Hungry Hippos	Damaged/Missing parts	This is such a fun game from my childhood. It has small pieces so it is not recommended if you hav...	7.1	<a href="#">Detail</a>
927	Computer game	BattleTech	Moderately used		7.1	<a href="#">Detail</a>

- b. For no rows returned Message will be shown "No results found"
- In postal code:
  1. Show the "In postal code" radiobutton
  2. Once the radiobutton is selected then enable the textbox. Do validate the postal code value.( It should match with the postal code reference table)

## Phase 2 Abstract Code w/SQL | CS 6400 - Spring 2022 | Team 103

Below query for getting the postal code from Address table

### Select distinct postal code from Address

3. On click of the “Search” button, below query will be executed

```
Select U.Email, I.ItemNUmber, I.Itemtype GameType, I.Title, I.Condition,  
I.Description,ty.Platform,ty.Media,ty.PieceCount,ty.Description,addr.Postal  
Code,addr.longitude,addr.latitude  
from user U join item I on U.Email=I.Email join itemtype ty on  
I.ItemType=ty.Itemtype and I.ItemNumber=ty. ItemNumber  
join address addr on I.PostalCode= U.PostalCode  
where I.Email='$Email' where U.Email <>'$Email' and  
U.postalCode='$Postalcode' /*Input from textbox*/;
```

### Below query will provide the logged user address details

```
Select U.User, addr.PostalCode,addr.longitude,addr.latitude  
from user U join address addr on addr.PostalCode= U.PostalCode  
where I.Email='$Email';
```

**/\* Logged In User details\*/**

**Using the longitude and latitude of the logged in user and the searched item Owner distance between is calculated**

## Phase 2 Abstract Code w/SQL | CS 6400 - Spring 2022 | Team 103

4. The distance of each user with the logged in user is calculated and then compared with the selected miles and only items whose distance with the logged in user is within the selected miles
5. On execution of the query
  - a. For return of records below screen will be displayed

---

Search results: keyword "tech"

Item #	Game type	Title	Condition	Description	Distance	
1885	Jigsaw puzzle	Georgia Tech campus	Mint		0.0	<a href="#">Detail</a>
300	Video game	Wolfenstein: The New Order	Like New	Utilizing the idTech 5 engine, this game is an incredible sequel to the previous games in the serie...	3.5	<a href="#">Detail</a>
124	Board game	Hungry Hungry Hippos	Damaged/Missing parts	This is such a fun game from my childhood. It has small pieces so it is not recommended if you hav...	7.1	<a href="#">Detail</a>
927	Computer game	BattleTech	Moderately used		7.1	<a href="#">Detail</a>

- b. For no rows returned Message will be shown "No results found"

## Phase 2 Abstract Code w/SQL | CS 6400 - Spring 2022 | Team 103

[View Item](#)

Abstract Code

- Once the Details button is clicked from the above list then below screen is shown

Item Details			
Item #	3	Offered by	Calamity G.
Title	Legend of Zelda: Breath of the Wild	Location	Redmond, WA 98052
Game type	Video game	Rating	0.45
Platform	Nintendo	Distance	2168.1 miles
Media	Game card	<div>Propose swap</div>	
Condition	Lightly used		

- Below select query is executed for getting the above details

## Phase 2 Abstract Code w/SQL | CS 6400 - Spring 2022 | Team 103

```
Select I.ItemNumber,U.NickName, I.Itemtype GameType, I.Title, I.Condition,  
I.Description,ty.Platform,ty.Media,ty.PieceCount,ty.Description,addr.longitude,addr.latitude,u.myrating,u.Email/*not  
to be shown on screen*/ from user U join item I on U.Email=I.Email join itemtype ty on I.ItemType=ty.Itemtype and  
I.ItemNumber=ty. ItemNumber join address addr on addr.PostalCode= U.PostalCode where  
ItemNumber='$ItemNumber' /*itemselected*/;
```

**Below query will provide the logged user address details**  
**/\* Logged In User details\*/**

```
Select U.User, addr.PostalCode,addr.longitude,addr.latitude  
from user U join address addr on I.PostalCode= U.PostalCode  
where I.Email=$Email where U.Email ='$Email';
```

- Using the longitude and latitude of the logged in user and the searched item Owner distance between is calculated
- Show “Propose Swap” button and once clicked show the “Propose Swap Page”

## Phase 2 Abstract Code w/SQL | CS 6400 - Spring 2022 | Team 103

### [Propose Swap](#)

#### Abstract Code

- Once the “Propose Swap” button is clicked then below screen is shown

Propose swap

⚠ The other user is 2168.1 miles away! ⚠

You are proposing a trade for  
Legend of Zelda: Breath of the Wild

**Confirm**

Please choose your proposed item:

Item #	Game type	Title	Condition	
23	Video game	Tetris	Lightly used	<input type="radio"/> Select
34	Board game	Monopoly	Damaged/Missing parts	<input type="radio"/> Select
106	Card game	UNO	Mint	<input type="radio"/> Select
250	Jigsaw puzzle	Sushi	Like New	<input type="radio"/> Select
345	Computer game	Myst	Moderately used	<input type="radio"/> Select

- Below select query is executed for getting the above details

## Phase 2 Abstract Code w/SQL | CS 6400 - Spring 2022 | Team 103

```
Select I.ItemNUmber,U.NickName, I.Itemtype GameType, I.Title, I.Condition,  
I.Description,ty.Platform,ty.Media,ty.PieceCount,ty.Description  
from user U join item I on U.Email=I.Email join itemtype ty on  
I.ItemType=ty.Itemtype and I.ItemNumber=ty. ItemNumber  
where U.Email ='$Email'/* logged in */ and I.Item not in (select distinct Item  
from Swap) ;
```

- The Selected Item Details from the previous screen are persisted for this page
- All the Items for the logged in user will displayed with select radiobutton for every Item
- Once logged in User selects one item then “Confirm” Button will be enabled

## Phase 2 Abstract Code w/SQL | CS 6400 - Spring 2022 | Team 103

- Once the “Confirm” Button is clicked then below insert statement will be executed

Insert into **swap**

(ProposedEmail,DateProposed,Status,Other User,CounterPartyEmail,ProposedItem,DesiredItem)

Values ('\$Email'/\*logged in\*/, systemdate,"Proposed", '\$Nickname'/\*Select Previous screen\*/,  
'\$CounterPartyEmail'/\*Select Previous screen\*/, '\$Item','\$DesiredItem'/\*Select Previous screen\*/);

**SwapId will be an incremental Number which will be populated for every insert**



## Phase 2 Abstract Code w/SQL | CS 6400 - Spring 2022 | Team 103

### [Accept or Reject Swaps](#)

#### Abstract Code

- You can come on this screen by clicking “Unaccepted Swaps” link then below screen is shown

to the item's item page, and an accept/reject mechanism should be provided.

Accept/reject swaps

Date	Desired Item	Proposer	Rating	Distance	Proposed Item	
1/15/2020	<a href="#">Cards Against Humanity</a>	HeroOfTime	4.99	8.2 miles	<a href="#">Super Mario Maker</a>	<div>Accept</div> <div>Reject</div>

If the swap is accepted, display a dialog with the proposer's email, first name, and phone number/type, if available and if sharing option is set. (If phone number is available, but not shareable, it can also be displayed as not available.)

Swap Accepted

Contact the proposer to swap items!

Email: royal.bodyguard@hyrule.gov

Name: Link

No phone number available

OK

- Below select query is executed for getting the above details

## Phase 2 Abstract Code w/SQL | CS 6400 - Spring 2022 | Team 103

```
Select ProposedDate,Desired ,Swapid,Item,Proposer,ProposedItem  
,addr.longitude,addr.latitude from user U join swap S on  
U.Email=S.CounterPartyEmail join address addr on  
u.postalcode=addr.postalcode where S.Status='Proposed';
```

**/\* Getting the Proposer rating,nickname, and longitude and  
latitude,phonenumber,email and first name\*/**

```
Select ed U.myrating,u.nickname,addr.longitude,addr.latitude  
from user U join swap S on U.Email=S.ProposedEmail  
join address addr on u.postalcode=addr.postalcode  
where S.Status='Proposed';
```

- If Accept button is clicked then below Update Statement is executed

```
Update swap set status='Accepted' ,DateResponded=system date where  
CounterpartyEmail='$CounterpartyEmail' and Swapid='$Swapid';
```

## Phase 2 Abstract Code w/SQL | CS 6400 - Spring 2022 | Team 103

And below message will be displayed

Swap Accepted ☐

---

Contact the proposer to swap items!

Email: royal.bodyguard@hyrule.gov

Name: Link

No phone number available

OK

- If Reject button is clicked then below Update Statement is executed  
Update `swap` set status='Rejected' ,DateResponded=system date where CounterpartyEmail='`$CounterpartyEmail`' and Swapid='`$Swapid`';

## Phase 2 Abstract Code w/SQL | CS 6400 - Spring 2022 | Team 103

### [Rate Swaps](#)

#### Abstract Code

- Below table will be shown. Select query will be executed for this table

Rate Swaps

Acceptance Date	My role	Proposed Item	Desired Item	Other User	Rating
06/02/2021	Proposer	Mastermind	Skip-Bo	PrincessZ	<input type="text" value=""/>
05/25/2021	Counterparty	Connect Four	Doom 3	KingRhoam	<input type="text" value=""/>

0

1

2

3

4

5

Select dateresponded as acceptancedatee, 'Proposer',ProposedItem, DesiredItem from [swap](#) where ProposedEmail='\$Email' and ProposerRating is null

Union

Select dateresponded as acceptancedatee, 'CounterParty',ProposedItem, DesiredItem from [swap](#) where CounterPartyEmail='\$Email' and CounterRating is null;

## Phase 2 Abstract Code w/SQL | CS 6400 - Spring 2022 | Team 103

- Every row in the table will have a Rating dropdown with prepopulated values
- Once the rating is select for that swap then below update Stmt is executed

```
Update swap set counterpartyrating='$Rating' where  
CounterpartyEmail='$Email' and Swapid='$SwapId';
```

```
Update swap set proposerrating = '$Rating' where  
ProposedEmail='$Email' and Swapid='$SwapId';
```

## Phase 2 Abstract Code w/SQL | CS 6400 - Spring 2022 | Team 103

### Swap History

#### Abstract Code

- Below table will be shown. Select query will be executed for this table

Swap history

My role	Total	Accepted	Rejected	Rejected %
Proposer	2	1	1	50.0%
Counterparty	2	2	0	0.0%

```
Select 1 Total, Case when Status='Accepted' then 1 else 0 end  
Accepted, Case when Status='Rejected' then 1 else 0 end Rejected,  
'Proposer' from swap where ProposedEmail='$Email' Union  
  
Select 1 Total, Case when Status='Accepted' then 1 else 0 end  
Accepted  
, Case when Status='Rejected' then 1 else 0 end Rejected,  
'CounterParty' from swap where CounterpartyEmail='$Email' ;
```

- Below table will be shown. Select query will be executed for this table

## Phase 2 Abstract Code w/SQL | CS 6400 - Spring 2022 | Team 103

Proposed Date	Accepted/ Rejected Date	Swap status	My role	Proposed Item	Desired Item	Other User	Rating	
06/01/2021	06/02/2021	Accepted	Proposer	Mastermind	Skip-Bo	PrincessZ	<input type="text"/> ▼	<a href="#">Detail</a>
05/15/2021	05/25/2021	Accepted	Counterparty	Connect Four	Doom 3	KingRhoam	<input type="text"/> ▼	<a href="#">Detail</a>
04/15/2021	04/23/2021	Rejected	Proposer	Sushi	Trivial Pursuit	Mario		<a href="#">Detail</a>
04/21/2021	04/23/2021	Accepted	Counterparty	Apples to Apples	Cribbage	GreenBro2	4	<a href="#">Detail</a>

```
Select ProposedDate,DateResponded Accepted_Rejected_Date,Status, 'Proposer'
MyRole, ProposedItem,DesireedItem,OtherUser,ProsperRating
from swap where ProposedEmail =' $Email'

Union

Select ProposedDate,DateResponded Accepted_Rejected_Date,Status,
'CounterParty' MyRole,
ProposedItem,DesireedItem,OtherUser,CounterpartyRating
from swap where CounterpartyEmail =' $Email';
```

## Phase 2 Abstract Code w/SQL | CS 6400 - Spring 2022 | Team 103

- If the any Rating is null then Drop down rating will provided to select the rating

Proposed Date	Accepted/ Rejected Date	Swap status	My role	Proposed Item	Desired Item	Other User	Rating
06/01/2021	06/02/2021	Accepted	Proposer	Mastermind	Skip-Bo	PrincessZ	<input type="text"/>
05/15/2021	05/25/2021	Accepted	Counterparty	Connect Four	Doom 3	KingRhoam	<input type="text"/>

Once rating is selected then below Update stmt will be executed

Update **swap** set counterpartyrating=\$rating where CounterpartyEmail='\$Email'and Swapid='\$SwapId';

Update **swap** set Proposerrating =\$rating where ProposedEmail='\$Email' and Swapid='\$SwapId';

- Once the Detail link/Button is clicked below screen is shown



## Phase 2 Abstract Code w/SQL | CS 6400 - Spring 2022 | Team 103

Swap Details		User details	
<b>Proposed</b>	04/21/2021	<b>Nickname</b>	GreenBro2
<b>Accepted/Rejected</b>	04/23/2021	<b>Distance</b>	10.3 miles
<b>Status</b>	Accepted	<b>Name</b>	Luigi
<b>My role</b>	Counterparty	<b>Email</b>	luigi@plumberbros.com
<b>Rating left</b>	4	<b>Phone</b>	404-555-6464 (work)
Proposed Item		Desired Item	
<b>Item #</b>	73	<b>Item #</b>	121
<b>Title</b>	Apples to Apples	<b>Title</b>	Cribbage
<b>Game type</b>	Board game	<b>Game type</b>	Card game
<b>Condition</b>	Damaged/Missing pieces	<b>Condition</b>	Mint
<b>Description</b>	We play this game with our friends, and have lost some of the cards. But it is still a fun game to play!		

Below select queries need to be executed

```
Select ProposedDate,DateResponded Accepted_Rejected_Date,Status,  
'Proposer' MyRole, ProposedItem,DesireedItem,OtherUser,CounterpartyRating  
from swap where CounterpartyEmail='$Email';
```

## Phase 2 Abstract Code w/SQL | CS 6400 - Spring 2022 | Team 103

```
Select I.ItemNUmber, I.Itemtype GameType, I.Title, I.Condition,  
I.Description,ty.Platform,ty.Media,ty.PieceCount,ty.Description  
from item I join itemtype ty on I.ItemType=ty.Itemtype where  
ItemNumber='$ProposedItem';
```

```
Select I.ItemNUmber, I.Itemtype GameType, I.Title, I.Condition,  
I.Description,ty.Platform,ty.Media,ty.PieceCount,ty.Description  
from item I join itemtype ty on I.ItemType=ty.Itemtype where  
ItemNumber='$DesiredItem';
```

```
Select U.User, addr.PostalCode,addr.longitude,addr.latitude from user U join  
address addr on I.PostalCode= U.PostalCode where I.Email='$Email' where  
U.Email='$ProposedEmail';  
Select U.User, addr.PostalCode,addr.longitude,addr.latitude from user U join  
address addr on I.PostalCode= U.PostalCode where I.Email=$Email where U.Email  
='$CounterpartyEmail' ;
```

## Phase 2 Abstract Code w/SQL | CS 6400 - Spring 2022 | Team 103

### [Update My Information](#)

Abstract Code

Below table will be shown. Select query will be executed

Update my information

Email

gburdell@gatech.edu

Nickname

YellowJacket1

Password

\*\*\*\*\*

City

Atlanta

First Name

George

State

Georgia

Last Name

Burdell

Postal Code

30332

Phone number (optional)

404-555-1212

Type

Home

Work

Mobile

☐ Show phone number in swaps

Update

## Phase 2 Abstract Code w/SQL | CS 6400 - Spring 2022 | Team 103

```
Select  
Nickname,password,FirstName,LastName,City,State,Postalcodr,PhoneNumber,PhoneType  
from user U join address addr on U.postalCode=addr. postalCode  
where Email='$Email';
```

- Nickname Textbox: User will be allowed to change Nickname.All the necessary Data Validation will be in place
- Password Textbox: User will be allowed to change Password.All the necessary Data Validation will be in place
- First Name Textbox: User will be allowed to enter First Name.All the necessary Data Validation will be in place
- Last Name Textbox: User will be allowed to enter Last Name.All the necessary Data Validation will be in place
- Postal Code Textbox: User will be allowed to change Postal code.All the necessary Data Validation will be in place
- City and State will be populated based on the verified postal code  

```
Select City,State from address where postalcode=$postalcode;
```
- Phone Number Textbox: User will be allowed to change Phone Number.All the necessary Data Validation will be in place

## Phase 2 Abstract Code w/SQL | CS 6400 - Spring 2022 | Team 103

- Show phone number in swaps checkBox: User will allowed to uncheck the checkbox.All the necessary Data Validation will be in place
- Type drop down box: User will be allowed to change from the set of values for the phone type.This dropbox is enabled if there is a legit phone number filled in the phone number text box.
- The Update button is enabled only if all the above fields are validated. On click below update query will be executed

Update **user** set

password='\$Password',FirstName='\$FirstName',LastName='\$LastName',Nickname'\$NickName',PostalCode='\$PostalCode' where Email='\$Email';

Update **phone** set Type='\$Type',ShareFlag='\$ShareFlg' where Number  
='\$PhoneNumber';