

Project team 7 - Gift Registry

Team Members: [Dhwani Contractor](#), [Sai Vikhyat Parepalli](#), [Prutha Shiroadkar](#), [Jeongwan Park](#)

PROJECT PROPOSAL

The project implements a Gift registry system, where the ProductManager of the shop can add the lists of the items available in the store.

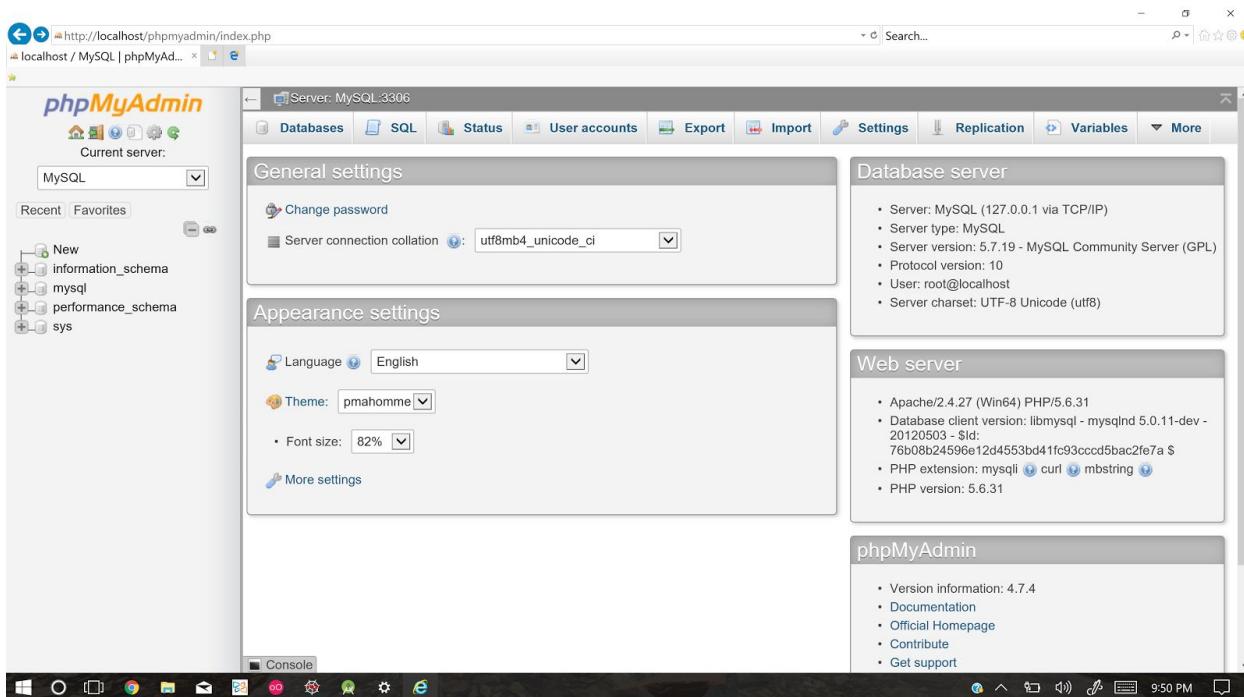
The inviter can add the wish list of the items they actually need and invitees can buy the items for them from the same list. By this gift registry system, inviters will always receive the gifts they need. So the invitees' money will be worth spending!

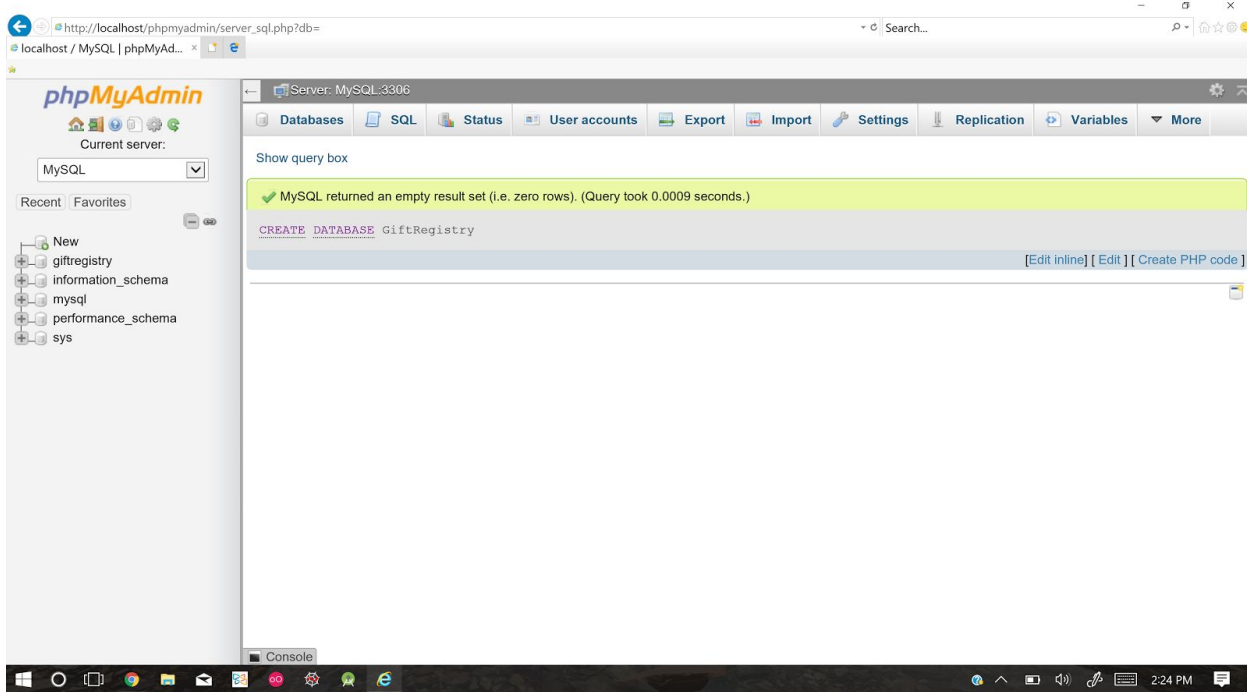
The database takes care of item availability, checks if no two invitees have bought the same gifts. The invitees can choose the gift according their budget.

The Invitees don't have to go to different places to find the buy/chose the gift.

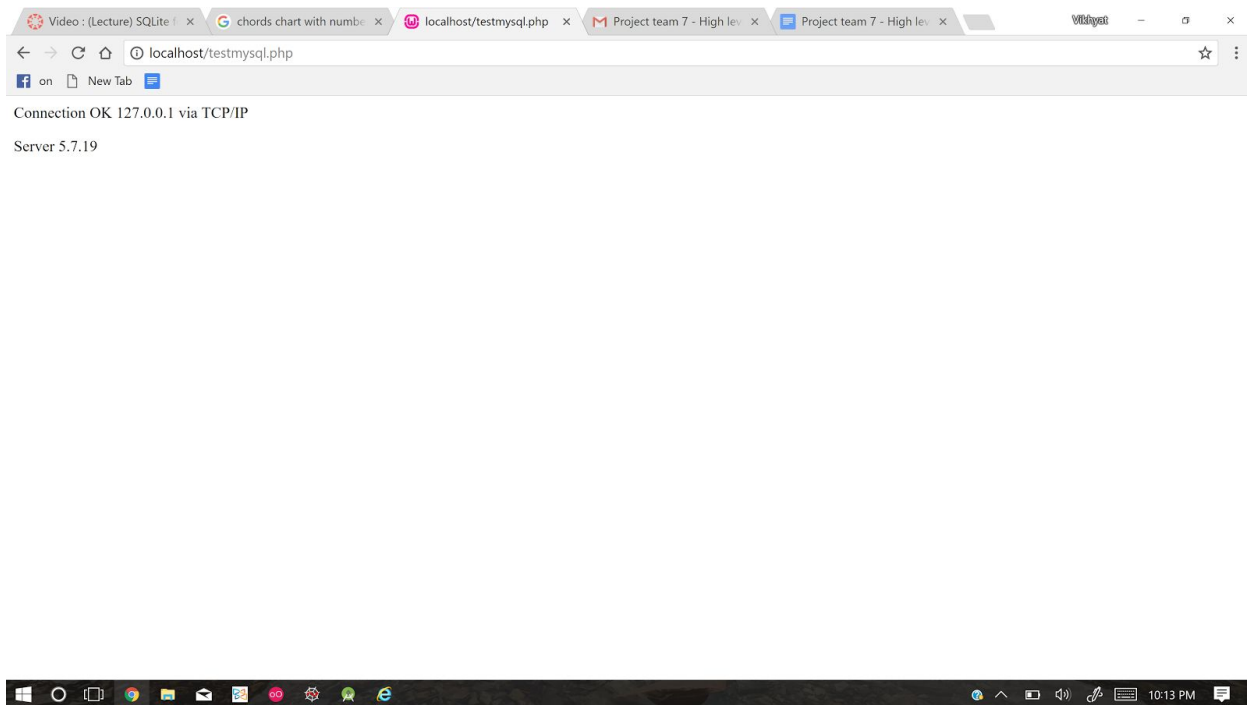
PROJECT ENVIRONMENT

We will be using phpMyAdmin(support for UI) with mySql as the environment setup by WAMP.





Database connection



<?php

```
$mysqli = new mysqli('127.0.0.1', 'root', '', 'giftregistry');
```

```

if ($mysqli->connect_error) {

    die('Connect Error (' . $mysqli->connect_errno . ') '

        . $mysqli->connect_error);

}

echo '<p>Connection OK '. $mysqli->host_info.'</p>';

echo '<p>Server '. $mysqli->server_info.'</p>';

$mysqli->close();

?>

```

HIGH LEVEL REQUIREMENTS

Initial user roles

| User Role | Description |
|----------------|--|
| Inviter | Inviters who have a registered account with the system. These Inviters can create registry and register their event and add items to their registry. Also they can create invitee list and send a message. They can view the registry list who is buying the items for them. |
| ProductManager | ProductManager will add/remove the product information. They add or delete product registry category and check the availability of the product in inventory. |
| Invitee | Invitee who have free access to inviter's registry. They can buy the registry items. Also they add shopping cart and place an order. They can view their order history and delivery status. |

Initial user story descriptions

| Story ID | Story description |
|----------|---|
| US1 | As a ProductManager, I want to sign-up into the application. |
| US2 | As a ProductManager, I want to login in my account. |
| US3 | As an ProductManager, I want to add items so that the Inviter can search the registry items. |
| US4 | As a ProductManager ,I want to add registry product's category tag. |
| US5 | As an Inviter, I want login my account |
| US6 | As an Inviter, I want create my event so that I can make registry with the event. |
| US7 | As an Inviter, I want create my registry so that I can manage my gift items to my registry. |
| US8 | As an Inviter, I want to create invitee list so that I send a message to access my registry. |
| US9 | As an Inviter, I want view my registry list so that I can check what items are ordered. |
| US10 | As an invitee, I want to login into my account. |
| US10 | As an Invitee, I want to view my inviter's registry so that I can check the inventory for availability. |
| US11 | As an Invitee, I want to add an item to my shopping cart from the available registry items. |
| US12 | As an Invitee, I want to place an order for the registry items. |
| US13 | As an Invitee, I want to check my order history and track the delivery status. |

HIGH LEVEL CONCEPTUAL DESIGN

Entities:

Product

Inviter

ProductManager

Registry
Invitee
Event
Cart
Order
Category

Relationships:

Inviter creates an event.
Inviter create a registry of their event.
Inviter edits registry.
Inviter lookup order status of their registry products.
Inviter create Invitee list.
ProductManager adds a product
ProductManager tag category to a product.
ProductManager remove product.
Invitee views Inviter's registry.
Invitee add registry items to their shopping cart.
Invitee order registry item.
Invitee lookup their order history and track the order status.

Sprint 1

REQUIREMENTS

| Story ID | Story description |
|----------|---|
| US1 | As a ProductManager, I want to sign-up into the application so that I can access the features. |
| US2 | As a ProductManager, I want to login in my account so that I can manage products to the inventory. |
| US3 | As a ProductManager, I want to add items so that the Inviter can search for the registry items. |
| US4 | As an Inviter, I want to sign-up into the application so that I can create a registry for my event. |
| US5 | As an Inviter, I want login my account so that I can create and manage my created event. |
| US6 | As an Inviter, I want create my event so that I can make registry with the event |

CONCEPTUAL DESIGN

Entity: **ProductManager**

Attributes:

- username
- name[composite]
 - first_name
 - middle_name
 - last_name
- password
- address [composite]
 - address_line1
 - address_line2
 - city
 - state
 - zip_code
- email_address
- phone_number

Entity: **Product**

Attributes:

- id
- name
- description
- price
- quantity
- company
- Insert_date

Entity: **Inviter**

Attributes:

- username
- name [composite]
 - first_name
 - middle_name
 - last_name
- phone_number [multi-valued]
 - phone_number1
 - phone_number2
- address[composite]
 - address_line_1
 - address_line_2
 - city

state
Zip_code
email_address

Entity: **Gathering**

Attributes:

id
name
date
time
description
venue[composite]
 address_line_1
 address_line_2
 city
 state
 zip_code

Relationship: **ProductManager** adds **Product**

Cardinality: One to Many

Participation:

ProductManager has partial participation
Product has total participation

Relationship: **Inviter** creates **Gathering**

Cardinality: One to Many

Participation:

Inviter has partial participation
Event has total participation

LOGICAL DESIGN

Table: **ProductManager**

Columns:

username
first_name
middle_name
last_name
password
address_line1
address_line2
city
state
zipcode
email_address
phone_number1
phone_number2

Primary key Justification: username will be unique for each Product Manager. So username becomes the primary key of the table ProductManager.

Table: **Product**

id
name
description
price
quantity
company
insert_date
pm_username[foreign key;references username of **ProductManager**]

Primary key Justification: id will be unique for each Product. Hence, it becomes the primary key for the table Product.

Foreign key justification: As *username* is the primary key of the table *ProductManager*, it can perfectly connect *ProductManager* table with *Product* table to identify which Product managers are managing the products. So pm_username becomes the foreign key for the Product table.

Table: **Inviter**

username
first_name
middle_name
last_name
password
address_line1
address_line2
city
state
zipcode
email_address
phone_number1
phone_number2

Primary key Justification: username will be unique for each Inviter. Hence, it becomes the primary key for the table Inviter.

Justification: Add two attribute(phone_number1, phone_number2) to represent the multi-valued attribute.

Table: **Gathering**

id
name
date
time
description
address_line1
address_line2
city
state
zip_code
inviter_username[foreign key;references username of **Inviter**]

Primary key Justification: id will be unique for each Event. Hence, it becomes the primary key for the table Gathering.

Foreign key justification: As *username* is the primary key of the table *Inviter*, it can perfectly connect Gathering table with *Inviter* table to identify which Inviter has created the event. Hence, inviter_username becomes the foreign key for the table Gathering.

SQL QUERIES

Query 1) To retrieve the total expense required for purchasing the products .

`select sum(price * quantity) as total_expense from product`

The screenshot shows the phpMyAdmin web interface. The left sidebar displays the database structure with the 'product' table selected under the 'registry' database. The main panel shows the SQL query: `select sum(price * quantity) as total_expense from product`. The query has been executed successfully, returning a single row with the value 9394.80001449585 for the 'total_expense' column. The interface includes a top navigation bar with tabs for Browse, Structure, SQL, Search, Insert, Export, Import, Privileges, Operations, and Triggers. A warning message at the top states: 'Current selection does not contain a unique column. Grid edit, checkbox, Edit, Copy and Delete features are not available.' Below the query results, there are options to show all rows, filter rows, and sort by key. The bottom of the interface shows a console area.

Query 2) To retrieve the Full name of the Product manager who has added the product "Stroller"

`select concat(first_name,' ',last_name) as name from productmanager p1 inner join product p2 on p1.username = p2.pm_username where p2.name = 'Stroller'`

http://localhost/phpmyadmin/tbl_sql.php?db=registry&table=productmanager

localhost / MySQL / registry

Amazon.com - Online Sh... Priceline.com

phpMyAdmin

Current server: MySQL

Recent Favorites

Server: MySQL:3306 » Database: registry » Table: productmanager

Browse Structure SQL Search Insert Export Import Privileges Operations Triggers

Show query box

⚠ Current selection does not contain a unique column. Grid edit, checkbox, Edit, Copy and Delete features are not available.

Showing rows 0 - 0 (1 total, Query took 0.0005 seconds.)

```
select concat(first_name, ' ', last_name) as name from productmanager p1 inner join product p2 on p1.username = p2.pm_username where p2.name = 'Stroller'
```

Profiling [Edit inline] [Edit] [Explain SQL] [Create PHP code] [Refresh]

Show all | Number of rows: 25 | Filter rows: Search this table

+ Options

name
Dhwani Contractor

Show all | Number of rows: 25 | Filter rows: Search this table

Query results operations

Print Copy to clipboard Export Display chart Create view

Console

Query 3) List the product names added by the Product manager whose username is "pshirodk"

select p2.name from productmanager p1 inner join product p2 on p1.username = p2.pm_username where p2.pm_username="pshirodk"

http://localhost/phpmyadmin/tbl_sql.php?db=registry&table=productmanager

localhost / MySQL / registry

Amazon.com - Online Sh... Priceline.com

phpMyAdmin

Current server: MySQL

Recent Favorites

Server: MySQL:3306 » Database: registry » Table: p2

Browse Structure SQL Search Insert Export Import Privileges Operations Triggers

Show query box

⚠ Current selection does not contain a unique column. Grid edit, checkbox, Edit, Copy and Delete features are not available.

Showing rows 0 - 2 (3 total, Query took 0.0009 seconds.)

```
select p2.name from productmanager p1 inner join product p2 on p1.username = p2.pm_username where p2.pm_username="pshirodk"
```

Profiling [Edit inline] [Edit] [Explain SQL] [Create PHP code] [Refresh]

Show all | Number of rows: 25 | Filter rows: Search this table

+ Options

name
Perfume chrome
baby shusher
Dinning table

Show all | Number of rows: 25 | Filter rows: Search this table

Query results operations

Print Copy to clipboard Export Display chart Create view

Console

Query 4) Write a query which gives Inviter's full name, event name and description added by the inviter whose username is "a6"

`select concat_ws(' ',first_name,middle_name,last_name)as "inviter name ",name ,description from gathering g inner join inviter i on g.inviter_username=i.username where i.username= "a6"`

The screenshot shows the phpMyAdmin interface for a MySQL database named 'registry'. The 'gathering' table is selected. The SQL query is entered in the query box:

```
select concat_ws(' ',first_name,middle_name,last_name)as "inviter name ",name ,description from gathering g inner join inviter i on g.inviter_username=i.username where i.username= "a6"
```

The query results are displayed in a table with 2 rows:

| inviter name | name | description |
|--------------|----------|----------------------|
| Robert Chase | Wedding | Monica weds Chandler |
| Robert Chase | birthday | Emma's Birthday |

The interface also shows the table structure for 'gathering' and 'inviter' on the left sidebar. The 'gathering' table has columns: id, inviter_username, name, description, city, date, address_line1, address_line2, state, zip_code. The 'inviter' table has columns: first_name, middle_name, last_name, email_address.