

Table of Contents

Table of Contents	1
Abstract Code	2
Login Form	2
View Outstanding Request Report	3
View Meals Remaining Report	5
View Available Bunks Report	7
Client Search	8
View Client Report	9
Client Check In	9
Edit client & Service Log	11
Sign Up Form	12
Food Bank Service Actions	13
Manage Items	13
Kitchen Service Actions	15
Shelter Service Actions	15
Manage Services	17
Add Service Screen	17
Edit Service Screen	19
Delete Service Screen	21
Item Request	22

Abstract Code

Login Form

Sub-tasks: no sub-tasks

- User enters email ('\$email'), password ('\$password') input fields
- If data validation is successful for both username and password input fields, then:
 - When **Enter** button is clicked:

```
SELECT username FROM `User` WHERE User.email= '$email' AND  
User.password=$password;
```

- If User record is found:
 - Store User information as session variables:
 - '\$userName'
 - '\$userSiteSID'
 - Go to **Main Menu**
- Else:
 - Go back to **Login** form, with error message

View Outstanding Request Report

Sub-tasks: generate outstanding requests report, sort outstanding requests report, mark partially fulfilled, fulfilled as is, marked unable to fulfill, reduce num of items

- After login, User starts on the **Main** screen
- Database is queried and checks to see if the User is associated with a site that has a food bank

```
SELECT site_sid FROM `FoodBank` WHERE site_sid='$userSiteSID';
```

- If User is associated with a Food Bank, the **Outstanding Request Report** is clickable
 - User clicks on **Outstanding Request Report** button
 - Query the database and build a report that contains the items requested to the associated site: item ('\$item'), storage type ('\$storageType'), category ('\$category') and sub-category ('\$subCategory'), and the quantity of items ('\$itemQuantity')

```
SELECT I.storage_type, S.supply_category, F.food_category, R.site_sid, R.item_qty, R.status
FROM `Item` AS I
INNER JOIN `Supplies` AS S ON I.item_name=S.item_name
INNER JOIN `Food` as F ON S.item_name=F.item_name
INNER JOIN `Request` as R ON F.item_name=R.item_name
WHERE site_sid='$userSiteSID';
```

- Query the database to see if more than one request for a particular item exists.

```
SELECT item_qty, item_name FROM `Request` WHERE site_sid='$userSiteSID';
```

- Query the database to see if the total number of items requested is greater than the total number available.

```
SELECT units_num, item_name FROM `FoodBank` WHERE site_sid='$userSiteSID';
```

- If so, highlight (red) the quantity of items ('\$itemQuantity') on the report
- Each report row will contain a dropdown with options: **Make a Selection, Fulfilled in Full, Partially Fulfilled, and Unable To Be Fulfilled.**
- Else if User is not associated with a Food Bank, **Outstanding Request Report** is not clickable
- If **Partially Fulfilled** or **Unable To Be Fulfilled** options are selected
 - Show input box for that request allowing the user to reduce the quantity of the item ('\$itemQuantity')

```
UPDATE `FoodBank` SET units_num=('$itemQuantity' - '$requestedQuantity') WHERE
item_name='$item';
```

- If user hits **Save** button, commit the changes
 - If Request fulfilled option drop-down is still **Make a Selection**
 - Skip the request and move onto the next one
 - Else If Request fulfilled option drop-down is **Fulfilled in Full, Partially Fulfilled, or Unable To Be Fulfilled**
 - Update the request in the database with the fulfilled, partially fulfilled, or unable to be fulfilled information
 - If the request is partially fulfilled or unable to be fulfilled and the user entered a new item quantity for the request, update the request with the new quantity
 - If the request is fulfilled, calculate the new item quantity and update the database
- Else If the user hits the **Cancel** Button
 - If user selects **Yes** button, they will be returned to the **Main** screen
 - If user selects **No** button, the user will remain on the **Outstanding Request Report**
- Else if user does not work with a site that has a food bank, display a message stating 'You do not work with a site that has a food bank. Unable to display report'

View Meals Remaining Report

Sub-tasks: no sub-tasks

- Query the database

```
SELECT sum(units_num) AS totalVegNutsGrainsBeans FROM `Item`
INNER JOIN `Food` ON Food.food_category = 'vegetables/nuts/grains/beans'
UNION
SELECT sum(units_num) AS totalMeatSeafood FROM `Item`
INNER JOIN `Food` ON Food.food_category = 'meat/seafood'
UNION
SELECT sum(units_num) AS totalDairyEggs FROM `Item`
INNER JOIN `Food` ON Food.food_category = 'dairy/eggs';
```

- Store into session variables: '\$totalVegNutsGrainsBeans', '\$totalMeatSeafood', and '\$totalDairyEggs'
- Set the smallest number from '\$totalVegNutsGrainsBeans', '\$totalMeatSeafood', and '\$totalDairyEggs' set '\$mealsRemaining' to that number
- display '\$mealsRemaining' on the **Meals Remaining report**
- Find the largest number from '\$totalVegNutsGrainsBeans', '\$totalMeatSeafood', and '\$totalDairyEggs' and store into session variable '\$donationsCompareNumber'.
- If largest number is '\$totalVegNutsGrainsBeans'
 - Subtract '\$totalMeatSeafood' from '\$totalVegNutsGrainsBeans' and store in session variable '\$meatSeafoodDonations'.
 - Subtract '\$totalDairyEggs' from '\$totalVegNutsGrainsBeans' and store in session variable '\$dairyEggsDonations'.
 - Set '\$vegNutsGrainsBeansDonations' to 0
- Else If largest number is '\$totalMeatSeafood'
 - Subtract '\$totalVegNutsGrainsBeans' from '\$totalMeatSeafood' and store in session variable '\$vegNutsGrainsBeansDonations'.
 - Subtract '\$totalDairyEggs' from '\$totalMeatSeafood' and store in session variable '\$dairyEggsDonations'.
 - Set '\$totalMeatSeafood' to 0
- Else If largest number is '\$totalDairyEggs'
 - Subtract '\$totalVegNutsGrainsBeans' from '\$totalDairyEggs' and store in session variable '\$vegNutsGrainsBeansDonations'.
 - Subtract '\$totalMeatSeafood' from '\$totalDairyEggs' and store in session variable '\$totalMeatSeafood'.
 - Set '\$dairyEggsDonations' to 0
- If '\$dairyEggsDonations', '\$totalVegNutsGrainsBeans', and '\$totalDairyEggs' are all 0
 - Set '\$dairyEggsDonations', '\$totalVegNutsGrainsBeans', and '\$totalDairyEggs' variables to '1+'

- Display '\$vegNutsGrainsBeansDonations', '\$meatSeafoodDonations', and '\$dairyEggsDonations' on the **Meals Remaining report**
- If user hits **Close** Button, return to the **Main** screen

View Available Bunks Report

Sub-tasks: no sub-tasks

- The database is queried and returns a list of available shelters along with their information.

```
SELECT Sh.male_bunk_num, Sh.female_bunk_num, Sh.mixed_bunk_num Si.short_name,  
Si.st_number, Si.street, Si.city, Si.state, Si.zip_code, CS.hours_of_op, CS.conditions_of_use,  
CS.description  
FROM `Shelter` AS Sh  
INNER JOIN `Site` AS Si ON Si.site_sid=Sh.site_sid  
INNER JOIN `ClientServices` AS CS ON Sh.site_sid=CS.site_sid  
WHERE Si.site_id='$userSiteSID' and CS.service_type='shelter' ;
```

- If no bunks are available, a message that says 'Sorry, all shelters are currently at maximum capacity.' will be displayed.
- Else if bunks are available
- Each table row is populated with the shelter's '\$name', '\$siteLocation', '\$phone', '\$hoursAndConditions', '\$maleBunkNumber', '\$femaleBunkNumber', and '\$mixedBunkNumber'
- If user hits **Close** button, return to the **Main** screen

Client Search

Sub-tasks: no sub-tasks

- User enters full name ('\$firstName' and '\$lastName') and/or ID ('\$clientIDNum') into input fields
- If user hits the **Search** button, the database will be queried based on '\$clientIDNum', '\$firstName' and '\$lastName':

- Query to be run if only '\$clientIDNum' provided:

```
SELECT firstname, lastname, phone_num, id_type, id_num FROM `Client`
WHERE Client.id_num='$clientIDNum' ;
```

- Query to be run if '\$firstName' and '\$lastName' provided:

```
SELECT firstname, lastname, phone_num, id_type, id_num FROM `Client`
WHERE Client.firstname='$firstName' AND Client.lastname='$lastName';
```

- Query to be run if '\$firstName', '\$lastName' and '\$clientIDNum' provided:

```
SELECT firstname, lastname, phone_num, id_type, id_num, gender FROM `Client`
WHERE Client.id_num='$clientIDNum'
OR (Client.firstname='$firstName' AND Client.lastname='$lastName');
```

- If >1 but <5 matches occur:
 - Display client name and descriptions in a drop-down list
 - User selects client and then the Client Report (**View Client Report**) will appear
- Else If 1 match occurs
 - Client Report (**View Client Report**) for the searched client will appear and
 - Client.client_sid will be stored in session variable '\$clientSID' and Client.gender will be stored in session variable '\$clientGender'
- Else If user uses a search combination that provides >4 clients
 - User is prompted to enter a more unique search combination
- Else If there isn't a match
 - A message will appear on the screen stating the client was not found in the database
- If user hits **Cancel** button,
 - The user will be prompted to confirm cancellation
 - If user selects **Yes**, they will be returned to the **Main** screen
 - If user selects **No**, the user will remain on the **Search Client** Screen

View Client Report

Sub-task: client check in, edit client, service log

Client Check In

- User may type in optional notes ('\$notes') into input box.
- If user hits **Check In Client** button,

- Query to be run

```
SELECT male_bunk_num, female_bunk_num, mixed_bunk_num
FROM `Shelter`
WHERE Shelter.site_sid='$userSiteSID';
```

- Store the results in session variables '\$maleBunkNum', '\$femaleBunkNum', and '\$mixedBunkNum'

- If '\$clientGender' is 'male'
 - If '\$maleBunkNum' is > 0
 - If true, set '\$bunkAvailable' = 'true'
 - Else if '\$mixedBunkNum' is > 0
 - If true, set '\$bunkAvailable' = 'true'
 - Else '\$bunkAvailable' = 'false'
 - Else If '\$clientGender' is 'female'
 - If '\$femaleBunkNum' is > 0
 - If true, set '\$bunkAvailable' = 'true'
 - Else if '\$mixedBunkNum' is > 0
 - If true, set '\$bunkAvailable' = 'true'
 - Else '\$bunkAvailable' = 'false'
 - If '\$bunkAvailable' is 'true'

```
INSERT INTO `UsageLog` VALUES (UsageLog.client_sid='$clientSID',
                                UsageLog.date_time_stamp='$currentTime',
                                UsageLog.notes='$notes',
                                UsageLog.site_sid='$userSiteSID');
```

- If '\$clientGender' is 'male'
 - If '\$maleBunkNum' is > '\$mixedBunkNum' or '\$maleBunkNum' is equal to '\$mixedBunkNum'

```
UPDATE `Shelter`
SET Shelter.male_bunk_num = Shelter.male_bunk_num - 1
WHERE Shelter.site_sid='$userSiteSID';
```

- Else if '\$maleBunkNum' is < '\$mixedBunkNum'

```
UPDATE `Shelter`
SET Shelter.male_bunk_num = Shelter.mixed_bunk_num - 1
WHERE Shelter.site_sid='$userSiteSID');
```

- Else If '\$clientGender' is 'female'
 - If '\$femaleBunkNum' > '\$mixedBunkNum' or '\$femaleBunkNum' is equal to '\$mixedBunkNum'

```
UPDATE `Shelter`
SET Shelter.female_bunk_num = Shelter.female_bunk_num - 1
WHERE Shelter.site_sid='$userSiteSID');
```

- Else If '\$femaleBunkNum' < '\$mixedBunkNum'

```
UPDATE `Shelter`
SET Shelter.female_bunk_num = Shelter.mixed_bunk_num - 1
WHERE Shelter.site_sid='$userSiteSID');
```

- Else If '\$bunkAvailable' is 'false'
 - A message will appear stating that there are no bunks available
- Else If user hits **Cancel** button,
 - The user will be prompted to confirm cancellation
 - If user selects Yes, they will be returned to the **Client Report** screen
 - If user selects No, the user will remain on the **Bunk Check In Screen**

Edit client & Service Log

- If user selects **Edit Client Information** button, go to **Edit Client** screen
 - User searches for Client by the ID number ('\$clientID') or by name ('\$firstName') and lastname ('\$lastName') and selects client from a drop-down list.
 - The client's firstname ('\$firstName'), lastname ('\$lastName'), phone number ('\$phone') input fields are populated with the proper data pulled from the database

```
SELECT firstname, lastname, phone_num FROM `Client` WHERE id_num='$clientID' OR
(firstname='$firstName' AND lastname='$lastName');
```

- User may modify firstname ('\$firstName'), lastname ('\$lastName'), and phone number ('\$phone') fields

```
UPDATE `Client` SET (firstname='$firstName', lastname='$lastName', phone_num='$phone')
WHERE id_num='$clientID' OR (firstname='$firstName' AND lastname='$lastName');
```

- If user hits **Save** button, the system will check what fields were changed and will write out the changes to a log file. Then the database will be updated
- If user hits **Cancel** button,
 - The user will be prompted to confirm cancellation.
 - If user selects **Yes** button, they will be returned to the **Main** screen
 - If user selects **No** button, the user will remain on the **Modify Client Screen**
- Else If user selects **Add New Service Log Entry** option, go to **Service Log Entry**
- If user hits **Save** button, the system will submit the new log entry containing client_sid ('\$clientSID'), date_time_stamp ('\$currentTime'), notes ('\$notes'), and site_sid ('\$userSiteSID') into the database

```
INSERT INTO `UsageLog` VALUES (UsageLog.client_sid='$clientSID',
                                UsageLog.date_time_stamp='$currentTime',
                                UsageLog.notes='$notes',
                                UsageLog.site_sid='$userSiteSID');
```

- If user hits **Cancel** button
 - The user will be prompted to confirm cancellation
 - If user selects Yes, they will be returned to **Client Report**
 - If user selects No, the user will remain on the **New Service Log Entry** screen

Sign Up Form

Sub-tasks: no sub-tasks

- User enters mandatory fields of '\$firstname', '\$lastname', '\$clientID', '\$clientIDType' and optionally a '\$ClientPhoneNum'
- User clicks on **Enroll New Client** button:
 - If mandatory fields but no '\$clientPhoneNum':

```
IF EXISTS (SELECT * FROM `Client` WHERE Client.id_num='$clientID')
  UPDATE `Client` SET
    Client.firstname='$firstName',
    Client.lastname='$lastName',
    Client.id_num='$clientID',
    Client.id_type='$clientIDType'
  WHERE Client.id_num='$clientID'
ELSE INSERT INTO `Client` VALUES (Client.firstname='$firstName',
    Client.lastname='$lastName',
    Client.id_num='$clientID',
    Client.id_type='$clientIDType');
```

- If mandatory fields and '\$ClientPhoneNum':

```
IF EXISTS (SELECT * FROM `Client` WHERE Client.id_num='$clientID')
  UPDATE `Client` SET
    Client.firstname='$firstName',
    Client.lastname='$lastName',
    Client.id_num='$clientID',
    Client.id_type='$clientIDType'
    Client.phone_num='$clientPhoneNum'
  WHERE Client.id_num='$clientID'
ELSE INSERT INTO `Client` VALUES (Client.firstname='$firstName',
    Client.lastname='$lastName',
    Client.id_num='$clientID',
    Client.id_type='$clientIDType'),
    Client.phone_num='$clientPhoneNum');
```

- Else If user hits **Cancel** button:
 - The user will be prompted to confirm cancellation
 - If user selects **Yes**, the fields on the **Sign Up Form** will be cleared
 - If user selects **No**, the prompt will disappear and data will remain

Food Bank Service Actions

Sub-tasks: generate inventory report, add item to inventory, edit item in inventory, remove item from inventory

- Show **Generate Inventory Report**, **Manage Items**, options
- Upon:
 - Click **Generate Inventory Report** button
 - Jump to the **Generate Inventory Report** screen
 - Click **Manage Items** button
 - Jump to the **Manage Items** screen

Manage Items

- Display options to **Add Item**
- If **Add Item** is selected, display the **Add Item** form
 - User puts in new item ('\$itemName') in input field
 - User selects from drop-down whether item is 'Food' or a 'Supply'
 - If the user selects **Food**
 - A drop-down will appear that contain the following categories: **Vegetables**, **Nuts/Grains/Beans**, **Meat/Seafood**, **Dairy/Eggs**, **Sauce/Condiment/Seasoning**, and **Juice/Drink**
 - User will select food category ('\$foodCategory') from the drop-down list
 - User selects storage type ('\$storageType') from a drop-down
 - User will be asked to input an expiration date if any ('\$expirationDate')
 - User will be asked to input a quantity ('\$itemQuantity') into input field

```
IF EXISTS (SELECT * FROM `Item` WHERE Item.item_name='$itemName')

    UPDATE `FoodBank` SET
        FoodBank.units_num = ((SELECT units_num FROM `FoodBank` WHERE
FoodBank.item_name='$itemName' AND FoodBank.site_sid='$userSiteSID')+'$itemQuantity')
        WHERE FoodBank.item_name='$itemName'

    UPDATE `Food` SET food_Category='$foodCategory' WHERE
Food.item_name='$itemName'

ELSE

INSERT INTO `Item` VALUES (Item.item_name='$itemName',
Item.storage_type='$storageType', Item.exp_date='$expirationDate')

UPDATE `FoodBank` SET FoodBank.units_num = ((SELECT units_num FROM `FoodBank`
WHERE FoodBank.item_name='$itemName' AND
FoodBank.site_sid='$userSiteSID')+'$itemQuantity');
```

- Else If the user selects **Supply**

- A drop-down will appear that contain the following categories: **Personal Hygiene, Clothing, Shelter, and Other**
- User will select supply category ('\$supplyCategory') from the drop-down list
- User selects storage type ('\$storageType') from a drop-down. The drop-down options are: **Dry Good, Refrigerated, and Frozen**
- User will type in expiration date ('\$expirationDate') into input field
- User will type in number of item ('\$itemQuantity') into input field
- If user hits **Save** button
 - Inserts new item from the input field into the database: '\$itemName', '\$storageType', '\$itemCategory', '\$itemQuantity', '\$expirationDate'.

```

IF EXISTS (SELECT * FROM `Item` WHERE Item.item_name='$itemName')

  UPDATE `FoodBank` SET
    FoodBank.units_num = ((SELECT units_num FROM `FoodBank` WHERE
FoodBank.item_name='$itemName' AND FoodBank.site_sid='$userSiteSID')+'$itemQuantity')
    WHERE FoodBank.item_name='$itemName'

  UPDATE `Supply` SET supply_category='$supplyCategory' WHERE
Supply.item_name='$itemName'

ELSE

INSERT INTO `Item` VALUES (Item.item_name='$itemName',
Item.storage_type='$storageType', Item.exp_date='$expirationDate')

UPDATE `FoodBank` SET FoodBank.units_num = ((SELECT units_num FROM `FoodBank`
WHERE FoodBank.item_name='$itemName' AND
FoodBank.site_sid='$userSiteSID')+'$itemQuantity');

```

- Go back to **Manage Items** task, displaying to-do options
- If user hits **Cancel** button
 - Go back to **Manage Items** task, displaying to-do options

Kitchen Service Actions

Sub-tasks: no sub-tasks

- Display the current kitchen seat count and the button **Reduce Soup Kitchen Seats**

```
// application provides $userSiteSID and $SERVICE_SID
SELECT SoupKitchen.seat_count
FROM 'SoupKitchen'
WHERE SoupKitchen.site_sid='$userSiteSID';
```

- If user clicks **Reduce Soup Kitchen Seats**
 - Display prompt for user to confirm
 - If user selects **Yes**
 - Reduce the available Soup Kitchen Seats count by one in the DB

```
// application provides $userSiteSID
UPDATE SoupKitchen.seat_count
SET SoupKitchen.seat_count=SoupKitchen.seat_count - 1
FROM 'SoupKitchen'
WHERE SoupKitchen.site_sid='$userSiteSID';
```

- Display a message notifying the user that the seat count was reduced and go back to the **Kitchen Service Actions** screen
 - Else If user hits **Cancel** button
 - Go back to **Kitchen Service Actions**

Shelter Service Actions

Sub-tasks: reset bunk bed count, view bunk bed count, reduce bunk bed count

- Display the current Bunk Bed Count in three categories:
 - Male Bunk Capacity ('\$maleBunkCapacity')
 - Female Bunk Capacity ('\$femaleBunkCapacity')
 - Mixed Bunk Capacity ('\$mixedBunkCapacity')
- Below display the following buttons: **Reset Bunk Bed Count** and **Reduce Bunk Bed Count**
- If user clicks **Reduce Bunk Bed Count**
 - Display drop-down menu to select from Male, Female or Mixed bunks
 - User selects from Male, Female or Mixed bunk options.
 - Save user selection to storage variable '\$bunkSelection'
 - Display **Confirmation** screen
 - If user selects **Yes**
 - Reduce the respective available Bunk Bed type count by one in the DB

```
UPDATE `Shelter`
SET Shelter.`$bunkSelection` = Shelter.`$bunkSelection` - 1
WHERE Shelter.site_sid='$userSiteSID';
```

- Display a message notifying the user that the bunk count was reduced and go back to the **Shelter Service Actions** screen
 - Else If user hits ***Cancel*** button
 - Go back to the **Shelter Service Actions** screen where the current Bunk Bed Count will be displayed
- Else if user clicks ***Reset Bunk Bed Count***
 - Display prompt for user to confirm
 - If user selects ***Yes***
 - Reset the available Bunk Bed count to the max availability

```
UPDATE `Shelter`
SET Shelter.`$bunkSelection` = '50'
WHERE Shelter.site_sid='$userSiteSID';
```

- Display a message notifying the user that the bunk count was reset and go back to **Shelter Service Actions** where the current Bunk Bed Count will be displayed
 - Else If user hits ***Cancel*** button
 - Go back to **Shelter Service Actions** where the current Bunk Bed Count will be displayed

Manage Services

Sub-tasks: add service, edit service, request DBA for site deletion, delete service

- Display options as buttons: **Add Service**, **Edit Service**, **Delete Service**
- If user clicks **Add Service** option, jump to **Add Service** screen
- Else If user clicks **Edit Service** option, jump to **Edit Service** screen
- Else If user clicks **Delete Service** option, jump to **Delete Service Screen**
- Else If user clicks **Cancel** button, jump to the **Main** screen

Add Service Screen

- Query the database to see if site offers Shelter based on site ID
 - If shelter is not offered yet
 - add **Shelter** as a drop-down (availableServices) option

```
IF EXISTS (SELECT Shelter.SITE_ID
           FROM 'Shelter'
           WHERE Shelter.site_sid='$userSiteSID')
```

- Query the database to see if site offers Soup Kitchen based on site ID
 - If soup kitchen is not offered yet
 - add **Soup Kitchen** as a drop-down (availableServices) option

```
IF EXISTS (SELECT SoupKitchen.SITE_ID
           FROM 'SoupKitchen'
           WHERE SoupKitchen.site_sid='$userSiteSID')
```

- Query the database to see if site offers Food Pantry based on site ID
 - If food pantry is not offered yet
 - add **Food Pantry** as a drop-down (availableServices) option

```
IF EXISTS (SELECT FoodPantry.SITE_ID
           FROM 'FoodPantry'
           WHERE FoodPantry.site_sid='$userSiteSID')
```

- Query the database to see if site offers Food Bank based on site ID
 - If food bank is not offered yet
 - add **Food Bank** as a drop-down (availableServices) option

```
IF EXISTS (SELECT FoodBank.SITE_ID
           FROM 'FoodBank'
           WHERE FoodBank.site_sid='$userSiteSID')
```

- Display a drop-down (availableServices) with available services

- Drop-down is dynamic and may include **Shelter, Soup Kitchen, Food Pantry, Food Bank**

- User selects from drop-down (**Available Services**) option
 - If selected option is **Shelter, Soup Kitchen, or Food Pantry**
 - Description ('\$serviceDescription'), hours of operation ('\$operationHours'), and Conditions for use ('\$serviceCondtions') input boxes appear on the screen
 - If selected option is **Food Pantry**
 - User types in Description ('\$serviceDescription'), hours of operation ('\$operationHours'), and Conditions for use ('\$serviceCondtions') input boxes

```
INSERT INTO `FoodPantry` VALUES (FoodPantry.site_sid='$userSiteSID');
INSERT INTO `ClientServices` VALUES (ClientServices.site_sid='$userSiteSID',
ClientServices.SERVICE_TYPE='Food Pantry',
ClientServices.HOURS_OF_OP='$operationHours',
ClientServices.CONDITIONS_OF_USE='$serviceConditions',
ClientServices.DESCRPTION='$serviceDescription');
```

- Else If selected option is **Soup Kitchen**
 - Seat Capacity ('\$seatingCapacity') input box appears on the screen
 - User types in Description ('\$serviceDescription'), hours of operation ('\$operationHours'), and Conditions for use ('\$serviceCondtions') input boxes

```
INSERT INTO `SoupKitchen` VALUES (SoupKitchen.site_sid='$userSiteSID',
SoupKitchen.seat_count='$seatingCapacity');
INSERT INTO `ClientServices` VALUES (ClientServices.site_sid='$userSiteSID'
ClientServices.SERVICE_TYPE='Soup Kitchen',
ClientServices.HOURS_OF_OP='$operationHours',
ClientServices.CONDITIONS_OF_USE='$serviceConditions',
ClientServices.DESCRPTION='$serviceDescription');
```

- If selected option is **Shelter**
 - User types into Male Bunk Capacity ('\$maleBunkCapacity') input box
 - User types into Female Bunk Capacity ('\$femaleBunkCapacity') input box
 - User types into Mixed Bunk Capacity ('\$mixedBunkCapacity') input box

```
INSERT INTO `Shelter` VALUES (Shelter.site_sid='$userSiteSID',
Shelter.male_bunk_num='$maleBunkCapacity',
Shelter.mixed_bunk_num='$mixedBunkCapacity',
Shelter.female_bunk_num='$femaleBunkCapacity',);
INSERT INTO `ClientServices` VALUES (ClientServices.site_sid='$userSiteSID'
ClientServices.SERVICE_TYPE='Shelter',
ClientServices.HOURS_OF_OP='$operationHours',
ClientServices.CONDITIONS_OF_USE='$serviceConditions',
ClientServices.DESCRPTION='$serviceDescription');
```

- Else If selected option is **Food Bank**

```
INSERT INTO `FoodBank` VALUES (FoodBank.site_sid='$userSiteSID')
INSERT INTO `ClientServices` VALUES (ClientServices.site_sid='$userSiteSID',
```

```
ClientServices.SERVICE_TYPE='Food Bank');
```

- If user hits **Add Service** button
 - If selected option is **Shelter**, **Soup Kitchen**, or **Food Pantry**
 - Service for that site is added to database: Available Service drop-down selection ('\$serviceType'), '\$description', '\$operationHours', '\$serviceConditions'
 - If selected option is **Shelter**
 - Input '\$maleBunkCapacity', '\$femaleBunkCapacity', '\$mixedBunkCapacity' into Shelter table based on site ID
 - Else If selected option is **Soup Kitchen**
 - Input '\$seatingCapacity' into Soup Kitchen database table based on site ID
 - Else If selected option is **Food Bank**, add Food Bank to Services table.
- If user hits **Cancel** button
 - The user will be prompted to confirm cancellation
 - If user selects **Yes** button, they will be returned to the Manage Services screen
 - If user selects **No** button, the user will remain on the Add Service Screen

Edit Service Screen

- Query the database and retrieve all services (service type) based on '\$userSiteSID'

```
SELECT service_type FROM `ClientServices` WHERE ClientServices.site_sid = '$userSiteSID';
```

- Add database results to **Offered Services** drop-down
- Display **Offered Services** drop-down
- If selected option is **Shelter**, **Soup Kitchen**, or **Food Pantry**
 - Description ('\$serviceDescription'), hours of operation ('\$operationHours'), and Conditions for use ('\$serviceConditions') input boxes appear on the screen
 - Query the database and retrieve service description, hours of operation, and conditions for use

```
SELECT description, hours_of_op, conditions_of_use FROM `ClientServices` WHERE ClientServices.site_sid = '$userSiteSID';
```

Populate Description ('\$serviceDescription'), hours of operation ('\$operationHours'), and Conditions for use ('\$serviceConditions') input boxes with retrieved data.

- If selected option is **Shelter**
 - Query the database and retrieve male bunk capacity , female bunk capacity, and mixed bunk capacity.

```
SELECT male_bunk_num, female_bunk_num, mixed_bunk_num FROM `Shelter` WHERE
Shelter.site_sid = '$userSiteSID';
```

- Set storage variables '\$maleBunkCapacity', '\$femaleBunkCapacity', and '\$mixedBunkCapacity'.
- Populate input boxes with '\$maleBunkCapacity', '\$femaleBunkCapacity', and '\$mixedBunkCapacity' and display input boxes on the screen

```
UPDATE `Shelter` SET (Shelter.male_bunk_num='$maleBunkCapacity',
                     Shelter.mixed_bunk_num='$mixedBunkCapacity',
                     Shelter.female_bunk_num='$femaleBunkCapacity');
WHERE Shelter.site_sid = '$USERsite_sid'
INSERT INTO `ClientServices` VALUES (ClientServices.site_sid='$userSiteSID',
                                     ClientServices.SERVICE_TYPE='Shelter',
                                     ClientServices.HOURS_OF_OP='$operationHours',
                                     ClientServices.CONDITIONS_OF_USE='$serviceConditions',
                                     ClientServices.DESRIPTION='$serviceDescription');
```

- Else If selected option is **Soup Kitchen**
 - Query the database and retrieve seat count

```
SELECT seat_count FROM `SoupKitchen` WHERE SoupKitchen.site_sid = '$userSiteSID';
```

- Populate input box with SoupKitchen.seat_count and display input box on the screen
 - User modifies text in Description ('\$serviceDescription'), hours of operation ('\$operationHours'), and Conditions for use ('\$serviceConditions') input boxes
 - If selected option is **Shelter**
 - User modifies male bunk capacity ('\$maleBunkCapacity'), female bunk capacity ('\$femaleBunkCapacity'), and mixed bunk capacity ('\$mixedBunkCapacity') input boxes
 - Else If selected option is **Soup Kitchen**
 - User modifies seat count ('\$seatCount') input box
- If User hits the **Save** button
 - Update database

```
UPDATE 'ClientServices' SET ClientServices.description = '$serviceDescription',
ClientServices.hours_of_op = '$operationHours', ClientServices.conditions_of_use =
'$serviceConditions'
WHERE ClientServices.site_sid = '$userSiteSID' AND ClientServices.service_type =
'$serviceType';
```

- If selected option is **Shelter**

```
UPDATE `Shelter` SET (Shelter.male_bunk_num='$maleBunkCapacity',
                    Shelter.mixed_bunk_num='$mixedBunkCapacity',
                    Shelter.female_bunk_num='$femaleBunkCapacity'
WHERE Shelter.site_sid = '$userSiteSID');
```

- Else If selected option is **Soup Kitchen**
 - Update database table with seat count

```
UPDATE `SoupKitchen`
SET SoupKitchen.seat_count='seatingCapacity'
WHERE SoupKitchen.site_sid = '$userSiteSID';
```

- Else If user hits **Cancel** button
 - The user will be prompted to confirm cancellation
 - If user selects **Yes**, they will be returned to the **Manage Services** screen
 - If user selects **No**, the user will remain on the **Edit Service Screen**

Delete Service Screen

- Query the database and retrieve all Client Services ('\$SERVICE_TYPE') based on site ID (\$userSiteSID')

```
IF EXISTS (SELECT ClientServices.SITE_ID
FROM 'ClientServices'
WHERE ClientServices.site_sid='$userSiteSID')
```

- Display a drop-down (**Offered Services**) with offered services
- User selects service from drop-down list
- If user hits **Delete Service** button
 - The user will be prompted to confirm deletion
 - If user selects **Yes** button
 - If there is only one service left in that site
 - Confirmation screen: the user will be warned that the site will be 'marked for deletion', to be deleted by a DBA
 - If user selects **Yes** button, that site will be marked for deletion

```
UPDATE 'Site'
SET MARKED_FOR_DELETION = 'TRUE'
WHERE site_sid='$userSiteSID'
```

- Else If user selects **No** button, the user will remain on the **Delete Service Screen**
 - If there are more than one service

- The service will be deleted from the database based on service_type

```
DELETE FROM 'Client Services.SITE_ID'
WHERE site_sid='$userSiteSID' AND SERVICE_TYPE='$SERVICE_TYPE';
```

- The drop-down list of offered services (**Offered Services**) is updated
- Else If user selects **No** button, the user will remain on the **Delete Service Screen**.
- Else If user hits **Cancel** button
 - The user will be prompted to confirm cancellation
 - If user selects **Yes** button, they will be returned to the **Manage Services** screen
 - If user selects **No** button, the user will remain on the **Delete Service Screen**

Item Request

Sub-tasks: **item_request**, **cancel_request**, **request_from_food_bank**, **generate_request_status_report**

If **Item Request** is selected, display a search form that contains input fields: expiration date ('\$expirationDate'), storage type ('\$storageType'), a Food/Supply drop-down ('\$foodOrSupply'), category ('\$category') and item name/description ('\$itemName')

- If user hits **Search** button
 - If user didn't type anything in input fields or select anything
 - Display all items on screen ('\$itemName', \$expirationDate, \$storageType, \$foodOrSupply, \$category), along with an input box to request number of items.
 - Else if user inputted into search fields and drop-down
 - Query Food Bank/Item database tables for the item searched for by user's input (\$expirationDate, \$storageType, \$foodOrSupply, \$category, and \$name)
 - If found, display item information on screen (\$itemName, \$expirationDate, \$storageType, \$foodOrSupply, \$category), along with an input box to request number of items.
 - Query Food Bank based on siteID to see if the associated items are owned by that foodBank
 - If Yes, display 'owned' on that item's row.
 - Else display a message stating the item was not found
- If user hits **Request Item** button
 - Loop through all item rows
 - If item has a number in the request input field
 - Get item information from the row '\$itemName' and '\$requestQuantity', and input a request into the Request database table

```
INSERT INTO `Request` VALUES (Request.site_sid='$userSiteSID',
                                Request.item_name='$itemName',
                                Request.item_qty='$requestQuantity',
                                Request.status='Pending');
```

- If user hits **Cancel** button
 - Go back to **Manage Items** task, displaying to-do options
- Query database for the item searched for requests made by user.
- If requests are found
 - Populate a list of requests that contain the request details and the request status.

```
IF EXISTS (Select Request.ITEM_NAME, Request.ITEM_QTY, Request.STATUS
FROM 'Request'
WHERE Request.site_sid='$userSiteSID');
```

Lastly, every request that is outstanding will have a button (**Cancel Request**) next to it to delete the request from the database

- If the request's status shows it to be a closed request
 - Display number of items provided
- If user hits **Cancel Request** button
 - Prompt the user to cancel the request
 - If user hits **Yes** button
 - Delete the request from the request table based on the request ID
 - Display a message stating 'The request was successfully cancelled'
 - Remove the request from the **Request Status Report** view
 - Else If user selects **No** button
 - Return to **Request Status Report**
- If user hits **Close** button
 - Return to the **Main** screen