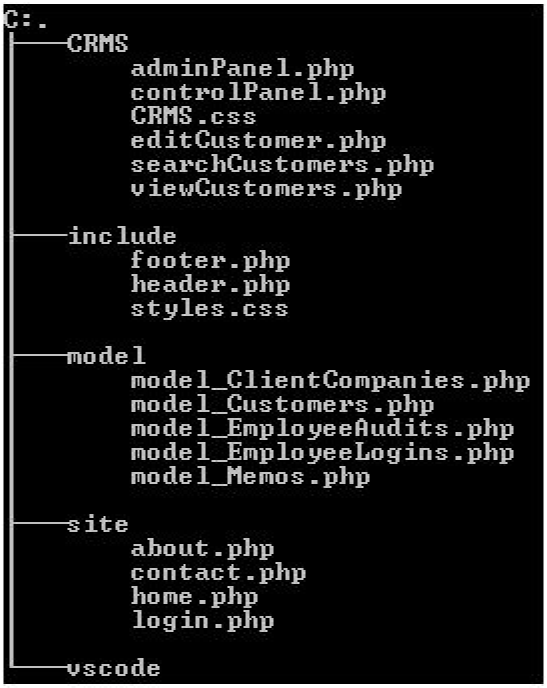
Saimer Nieves & George Saban Jr.
Professor Renselaar
SE266.05 Web Dev Using PHP & MySQL
24 November 2020

Web Dev Using PHP & MySQL W7 Final Project Tech Spec

Directory Structure Diagram



```
C:.
├───CRMS
│       adminPanel.php
│       controlPanel.php
│       CRMS.css
│       editCustomer.php
│       searchCustomers.php
│       viewCustomers.php
│
├───include
│       footer.php
│       header.php
│       styles.css
│
├───model
│       model_ClientCompanies.php
│       model_Customers.php
│       model_EmployeeAudits.php
│       model_EmployeeLogins.php
│       model_Memos.php
│
├───site
│       about.php
│       contact.php
│       home.php
│       login.php
│
└───vscode
```

Model Folder

model/model_EmployeeLogins.php: Contains all of the database related functions for the EmployeeLogins table. The maintenance and addition of records for this table is manually completed by the NIAN business administrators (i.e. Saimer & George). Functions include:

- checkEmployeeLogin($name, $pass) - Evaluates whether the provided credentials exist in the employeeLogins table. Returns 2 if successful and is a manager, 1 if successful and is an employee, and 0 otherwise.

model/model_ClientCompanies.php: Contains all of the database related functions for the ClientCompanies table. The contents interconnect the employeeLogins' and Customers' related table records together (via a foreign key field). The maintenance and addition of records for this table is manually completed by the NIAN business administrators (i.e. Saimer & George).

model/model_EmployeeAudits.php: Contains all of the database related functions for the EmployeeAudits table. Moreover, no audit log entries can be deleted. Functionality includes:

- getEmpAudits() - Retrieves all of a company's employees CRUD audits pertaining to their customers' data. Returns an array of associative arrays. Each associative array contains field names $auditID, $userID, $aDate, $aAction, and $customerID.

- getEmpAudit($eID) - Retrieves all audits of a single employee. Returns an array of associative arrays. Each associative array contains field names: $auditID, $aDate, $aAction, and $customerID.
- addEmpAudit($eID, $date, $action, $cusID) - Inserts a record of an employee's manipulation of customer information (e.g. creating, reading, updating, and deleting a customer) into the employeeAudit table. Returns a 1 for successful entry into the database and 0 otherwise.

model/model_Customers.php: Contains all of the database related functions for the Customers table. Functionality includes:

- getCustomers() - Retrieves all existing customers for the client company. Returns an array of associative arrays. Each associative array contains field names: $customerID, $firstName, $lastName, $address1, $phone.
- getCustomer($id) - Returns a customer selected by the user to inspect. Returns an associative array that contains field names: $profileImg, $firstName, $middleName, $lastName, $address1, $address2, $city, $state, $zip, $phone, $email, $purchases, and $birthday.
- searchCustomers($f, $l, $bd, $city, $state) - Searches for customers recorded by a client company in the customer table that matches the provided values from the user. Returns an array of associative arrays that contains field names: $profileImg, $firstName, $middleName, $lastName, $address1, $address2, $city, $state, $zip, $phone, $email, $purchases, and $birthday.
- addCustomers($companyID, $pImg, $fName, $mName, $lName, $addr1, $addr2, $city, $state, $zip, $phone, $email, $purch, and $bd) - Inserts a client company's customer to the customers table. Moreover, the customer would only be viewable by their respective client companies. Returns a 1 for a successful entry and 0 otherwise.
- updateCustomers($companyID, $pImg, $fName, $mName, $lName, $addr1, $addr2, $city, $state, $zip, $phone, $email, $purch, and $bd) - Updates a client company's customer to the customers table. Moreover, the customer would only be viewable by their respective client companies. Returns a 1 for a successful entry and 0 otherwise.
- deleteCustomers($id) - Deletes a client company's customer from the customers table. Returns a 1 for a successful deletion and 0 otherwise.

model/model_Memos.php: Contains all of the database related functions for the Memos table. Functionality includes:

- getMemos($custID) - Retrieves all memos regarding the selected customer. Returns an array of associative arrays with field values: $memoID, $customerID, $mDate, $memo.
- addMemos($custID, $memo) - Adds a memo entry or entries into the Memos table. The dates are auto-generated by SQL when inserted into the Memos table. Returns a 1 for successful insertion or 0 otherwise.
- deleteMemos($custID) - Deletes a memo entry from the Memos table. Returns a 1 for successful deletion or 0 otherwise.

Page Layouts

home.php: Represents the static home page of Saimer & George's Customer Relationship Management System application (NIAN CRMS).

about.php: Represents the static about page of NIAN CRMS that contains the purpose of the program, company rules, origin, etc.

contact.php: Represents the static contact page of NIAN CRMS that contains the contact information of the business owners (e.g. Saimer and George). Afterwards, when a purchase between a client company and the aforementioned is completed, accounts would be created and leased to the companies manually.

login.php: Static login page that allows entry to the NIAN CRMS application.

- Checks users' credentials to determine that they are a valid user and their user group (Employee | Manager).
    - Functions called:
        - **checkEmployeeLogin(**$name, $pass**)**

controlPanel.php: Dynamic summary page that displays the client company's customer statistics. Examples include a line chart reporting customer growth over time, a pie chart that displays the total population of customers per state.

adminPanel.php: Contains an employee audit log that only client company manager accounts can access. All of the employee's actions regarding customer information are recorded and viewable by manager accounts.

- For each loop to view employee audits.
    - Functions called:
        - **getEmpAudits()** | Displays the employee audit for all employees
        - **getEmpAudit($eID)** | Displays audits on specific employee

editCustomer.php: Represents the add and update customer page for client companies. The page allows inspection of existing customers, entry of a new customer, updates for current data, and deletion of a customer.

- For each loop to populate existing field data in the selected customer.
- Functions called:
    - **getCustomer($id)** | Displays information about specific customer
    - **getMemos($custID)** | Displays all existing memos
    - **updateCustomers($companyID, $pImg, $fName, $mName, $lName, $addr1, $addr2, $city, $state, $zip, $phone, $email, $purch, and $bd) |** Update customer information
    - **addCustomers($companyID, $pImg, $fName, $mName, $lName, $addr1, $addr2, $city, $state, $zip, $phone, $email, $purch, and $bd) |** Adds a brand new customer to database
    - **addMemos($custID, $memo)**
    - **deleteCustomers($id)** | Deletes customer from records based on their ID

  - **deleteMemos($custID)** | Deletes a memo entry regarding the selected customer
  - **addEmpAudit($eID, $date, $action, $cusID)** | Records action done by employee and sends it to the EmployeeAudits table

viewCustomers.php: Contains a table grid that lists all customers for a client company.

- For each loop to view an associative array.
- Functions called:
  - **getCustomers()** | This function will display all customers
  - **deleteCustomers($id)** | Deletes customer from records based in ID
  - **addEmpAudit($eID, $date, $action, $cusID)** | Records action done by employee and sends it to the EmployeeAudits table

searchCustomers.php: Allows client company users to search for an existing customer(s) in the database based on their text field inquiries. Matching results would reuse the viewCustomers.php page and be listed in a table grid.

- Function that passes an associative array for matching records to the viewCustomers.php
- Functions called:
  - **searchCustomers($f, $l, $bd, $city, $state)** | Returns matching customers from the database