# EVALUATION AND PRESENTATION GUIDE

The AltHealth Database System

JUNE 7, 2021

ICT3715

Student: 5705-033-3

# Table of Contents

# System Project Information

## Overview

In this module I have been approached to design and implement a working solution for a community business AltHealth, which is a provider for alternative healthcare solutions who sells supplements and offers a consultation service with the health care practitioner.

The purpose of system is to replace an older and outdated system with a new and convenient database driven system. This database system focusses in digitalising all patient records and to keep better track of supplements in stock, client orders and client appointments. From the owner's (the Health Care Practitioner – HCP) side of the system, more information about the various business activities within the business (namely the amount of consultations held with clients, client demographics and supplements sold) is provided enabling the owner to take more calculated decisions regarding the business offerings and position in the future.

With the data provided by AltHealth, the data has been normalised accordingly and integrated into a structured relational database management system (in this project MySQL was utilised) with the appropriate data relationships discovered and assigned. The functionality of the system contains a smooth and effective interface for both users, namely the HCP and General Administrative (GA).

The hardware and software requirements of the system is in line with the budget set by the business at the start of the project. AltHealth has exclusive rights of the system as this solution is exclusively developed and designed for their use. Intellectual Property Ownership remains at AltHealth, but the owner of the coding remains with me.

After the installation, implementation and training of the system, continued support would be provided if and when required by the business. I would be more than willing to assist the business wherever possible.

The name of my system is **The AltHealth Database System**.

## Phase Selected: Phase 2

In this project I have selected to design and implement Phase 2 of the two development phases for this project. This includes the graphical user interface (GUI) for adding clients to the database, as well as a booking system where the HCP or GA can schedule appointments.

## Type of System

The system is web based, as the front-end coding consists of web programming code, such as HTML, JavaScript and CSS. The system is hosted over the Local Area Network. On the main computer, where all the coding is installed for utilisation, the URL for the system is http://localhost.

The main computer (i.e. localhost) is connected to a router, which is connected to the Internet connection. Any other authorised computer or device (such as the HCP's tablet computer) connected to the same network may access the system (locally) by entering the IP-address of the localhost computer, for example 192.168.8.107, to access the system. The system can indeed

operate as a stand-alone, but provided that an Internet connection is present due to the API's I have utilised for the creation of the MIS reports.

Currently, the system is dependant on a Microsoft Windows Operating System only (Windows 7 or higher). However, if the system is accessed through the network on a different device, the system is not operating system dependant. The system can, however, be installed on other operating systems (such as Linux or MacOS) but the appropriate software versions used for this project would need to be sourced that would be compatible for the relevant operating system.

Even though that this system is not designed as a mobile application, the GUI is still accessible through the device's web browser provided the device is granted access onto the network and a web browser is utilised.

## Programming and Coding

The coding I have used are as follows:

- **Front-End Coding**: **HTML** (to arrange the user interface to display neatly with styling), **JavaScript** (including jQuery for the Google Charts function) and **CSS** (for styling the GUI)

- **Back-End Coding**: **PHP** (server-side scripting language working alongside with HTML) and SQL (for database functionality). JavaScript is used to assist in the creation of the charts for the MIS reports.

For this system, I have made use of a MySQL relational database management system.

# Documentation and Resources

For this project, I have created documentation files that is associated with the system. All relevant documentation, including this guide, has been added to my myUnisa Dropbox. The files on the server are as follows:

| Document Name | Version | Last Modified/Uploaded |
|---|---|---|
| Installation and Administration Manual | Version 1.4 | 07 June 2021 |
| User Manual | Version 1.2 | 07 June 2021 |
| Database Guide | Version 1.0 | 25 November 2019 |
| Evaluation and Presentation Guide (This current document) | Version 11.2019 | 07 June 2021 |
| AltHealth.zip | N/A (This zipped file contains my code of my system within the htdocs folder) | 07 June 2021 |

With regards to resources, I wish to declare in this section the software I have utilised for assisting me in implementing the system. These software packages are also mentioned in the installation and administration manual which elaborates the installation that needs to be conducted to install and initialise the system.

The software packages I have used on a Windows machine are as follows:

- XAMPP
- Google Chrome
- Backup and Sync from Google (for Google Drive)

I also wish to state that I have used external scripts within my code to assist me in displaying my code correctly in the GUI. These resources are as follows:

- The Login Page – A style script from Bootstrap (in the link below) has been used to style part of the login page (such as user inputs and error displays): https://maxcdn.bootstrapcdn.com/bootstrap/3.3.7/css/bootstrap.css

- The Navigation Bar – A style script for font-awesome has been used to display the arrows located in the navigation bar. The link is as follows: https://cdnjs.cloudflare.com/ajax/libs/font-awesome/4.7.0/css/font-awesome.min.css

- The Google Charts Library – The JavaScript located at the link below has been used to load the chart libraries from Google in order to assist me in creating the interactive MIS charts https://www.gstatic.com/charts/loader.js

# The Reports

In this system, I have created the functionality to display reports in an interactive dashboard. I have created three day to day reports and four MIS reports, as explained below.

## Day to Day Reports

### Appointments for Today

In this report, viewable for both HCP and GA users, a database query is executed to display the patients who are scheduled to see the HCP today. The patient's name and surname is displayed and the timeslot he/she is booked for. The HCP has the option to initiate the consultation by pressing on the button allocated next to each appointment.

The SQL query used is as follows (note that parameters have been used to execute the query with PDO in PHP to assign the current date to the :app_date parameter):

```
SELECT c.client_id, c.c_name, c.c_surname, b.app_time_start, b.app_time_end
FROM tblclientdata c JOIN tblbookingsinfo b
ON b.CLIENT_ID = c.CLIENT_ID
WHERE app_date = :app_date';
```

### Client Birthdays Today

In this report, viewable for both HCP and GA users, a database query is executed to display the clients who is celebrating a birthday today. The patient's ID number is displayed (only on the HCP side) and the names of the clients (for both users). At the bottom of the table a button may be pressed to send a personalised email to the clients displayed.

The SQL query used is as follows:

```
SELECT client_id, c_name, c_surname, c_email
FROM tblclientdata
WHERE client_id LIKE CONCAT('%', (SELECT DATE_FORMAT(SYSDATE(), '%m%d') LIMIT 1), '%');
```

### Low Stock Levels for Supplements

In this report, viewable for both HCP and GA users, a database query is executed to display the supplements who are in need to be ordered as stock levels have dropped below the set minimum limit of stock to be on-hand. The results are displayed in a table, with the option to view the supplier of the supplement and to update the supplement when the levels or prices have changed. The action can be performed by both the HCP and the GA.

The SQL query used is as follows:

```
SELECT supplement_id, min_levels, current_level, supplier_id
FROM tblsupplements
WHERE min_levels > current_level;
```

## Management Information System Reports

All MIS reports are exclusively viewed to the HCP in order to keep track on how the business performs. With the statistics provided graphically, the HCP can make informative decisions in the near future. Also, the HCP can even keep track of the health trend, for example the duration of the flu season or any other possible illnesses that requires a consultation.

### Referrals of Clients

This report displays a graphical representation of the references of clients and the percentage (segment) of the type of referral. This can assist the HCP in making future decisions to improve a certain reference segment, for example improving the business' website in order to reduce the amount of no referrals. The graphical representation is in the form of a pie chart.

The SQL query used is as follows:

```
SELECT c_reference, COUNT(*) AS count_of_reference
FROM tblclientdata GROUP BY C_REFERENCE
ORDER BY count_of_reference;
```

### Representation for the Top 10 Areas most clients reside in

This report displays a graphical representation (in the form of a column chart) of the ten areas (postal codes) where most clients reside in. The HCP can keep track where most of his clients reside and take future decisions in focusing on other areas to increase revenue, such as having consultation rooms closer to clients.

The SQL query used is as follows:

```
SELECT DISTINCT POSTAL_CODE, COUNT(*) AS count_of_postal
FROM tblclientdata
GROUP BY POSTAL_CODE
ORDER BY `count_of_postal` DESC LIMIT 11;
```

Notice that the DESC LIMIT is set to 11, due to the fact that the results contains a UNK value (i.e. unknown) which indicates the number of patients whose postal codes are still unknown to the business. The DESC LIMIT amount will be changed once the postal codes are provided.

### Number of Supplements Sold in a Specific Period

This report displays a graphical representation of the supplements that were sold in a given timeframe. This effectively allows the HCP to see which supplements are the most popular and less popular in a given period. This assists the HCP in taking future decisions to increase sales of other supplements, to take decisions in replacing some supplements with new or improved supplements, or to cease in selling specific supplements that aren't selling at all.

The SQL query used is as follows (notice that variable parameters $date_start and $date_end is used during the execution of the query by making use of mysqli in order to determine the date period):

```
SELECT DISTINCT supplement_id, SUM(quantity) as calc_quantity
FROM tblinvoicesupplements s JOIN tblinvoiceinfo i ON s.INV_NUM = i.INV_NUM
WHERE i.INV_DATE BETWEEN '$date_start' AND '$date_end'
GROUP BY supplement_id;
```

## Area Representation of Patients Seen during a Specific Period

This report displays a graphical representation of the postal codes of clients who has seen the HCP in a given timeframe. This allows the HCP to keep track of the spread of potential illnesses, such as the flu, that might have led to a consultation with the HCP. This data may assist in future health studies in determining the root source or cause of an acute illness and any potential steps that may be taken in the future to minimize or eliminate the cause. The HCP has the option to query the frequency by providing two dates of which he would like to display the data. The graphical representation of the data is in the form of a column chart.

The SQL query used is as follows (notice that variable parameters $date_start and $date_end is used during the execution of the query by making use of mysqli in order to determine the date period):

```
SELECT postal_code, COUNT(*) AS count_of_postal
FROM tblclientdata c JOIN tblbookingsinfo b ON c.CLIENT_ID = b.CLIENT_ID
WHERE APP_DATE BETWEEN '$date_start' AND '$date_end'
GROUP BY POSTAL_CODE;
```

Each MIS report can be accessed individually to view, and the timeframe can also be edited on the Area Representation and Number of Supplements Sold reports on the individual views. In addition, all reports has the functionality to be viewed in a table that can be printed by the HCP for his use or records.

## System Functionality and GUI

In my presentation I will be demonstrating the functionality of the system through the Graphical User Interface provided for the system. I will also be describing some of the aspects mentioned in the user manual I have created for the system, which include the users of the system.

It may be noticed that for the business' client side (i.e. the general public) the website may appear empty (as in not enough information is displayed). In this phase of the system development, I have mainly focussed on the HCP and the GA users of the application as these two users are the key players in the system.

I will be demonstrating the forms I have created, and the validation measures I have implemented. Appropriate error messages are displayed if the form detects that invalid data has been provided by the user. This not only secures data integrity for the system, but also acts as a preventative measure to avoid erroneous data being entered into the database which would affect its integrity.

I have also implemented automatic email functionality where some actions would result in an email sent out to the client, such as new invoice after consulting with the HCP, a personalised birthday message for the client, booking confirmation and updated invoice reflecting the amount that has been paid and received.

As mentioned earlier, the key roles for the system is the HCP and the GA. It has been noted that these two users are not fully computer literate. I have therefore designed the system in such a way that the users would understand the forms, the actions performed and/or being requested.

## The Database (MySQL)

The source of the data came from AltHealth, which was in a Microsoft Excel workbook. The data has been refined, cleaned and imported into the AltHealth database.

In my demonstration, I will be logging into the database I have created for AltHealth. The relational database management system I have used is MySQL. MySQL is supported in XAMPP and a graphical user interface is made available to assist in managing the database. I will be displaying the relations I have created, the enforced referential integrities and the datatypes I have assigned to each field.

Furthermore, I will be demonstrating that my database is interacting with the front end I have designed by adding a patient, supplement, supplier and creating a booking for the newly created client. I will also be updating a patient, supplier and supplement and will show that the data has been successfully updated in the database.

# Backup and Recovery

Backup and recovery of the AltHealth Database System is of utmost importance as it is vital to prevent the risk of any data loss, should any error occur with the system. The backup of the system does not include my code (for example PHP and HTML code) as to protect the integrity of the code. The backup of my code is securely backed up on my storage mediums (i.e. local, external and secure cloud storage away from the database). The backup of the system composes of the whole database which consists of the tables, data structures and the data itself. Backups are stored both locally (on the hard drive) and externally (on cloud storage automatically and on external devices manually). Should it be necessary to perform a reinstallation of the system at the client's premises, I will be accompanied with my code and the latest backup to perform the reinstallation process.

The backup system has been designed in such a way that no action needs to be warranted by the HCP or GA. Through the use of a Windows Batch File I have coded, I instruct the system through the Task Scheduler of the Windows Operating System that the executional file of the database (i.e. mysqldump.exe) be utilised to create a backup of the entire database to an SQL file which would be stored on the hard drive of the host machine. This therefore creates a local backup on the computer's hard drive. The task is named as "AltHealth" on the Task Schedule. This task has been set to run automatically every Friday at 15:30 (03:30 PM). Should the system (the computer) not be powered on at this time or the process cannot be initiated at the specified time, the task has been instructed to run as soon as possible. The time of the backup can be adjusted to a different time slot or a different frequency, such as daily or monthly. I do recommend that the backup be run on a weekly basis, during office hours and on a workday due to the possibility that the system would not be in operational after hours, such as over a weekend. This would also ensure that the system would retain as much data as possible should a backup be necessary to perform. The name of the file is althealth_backup_yyyy-mm-dd_hh-mm.sql, where yyyy-mm-dd represents the creation date of the file and hh-mm represents the time at which the file has been created.

To ensure that a copy of the backup is stored on an additional medium, I have installed "Backup and Sync" by Google Drive to ensure that the backup is automatically uploaded to the business' cloud storage medium. For the purpose of this project I have created a Google account for the business which contains 15GB of online storage free of charge. With the software installed, the local folder on the system which contains the automated SQL backups of the system is synchronised with the cloud. The exact image of the folder is available on Google Cloud which means that folder is available both locally and on the cloud. Should there be no Internet connectivity available to store the backup file immediately, the program is set to automatically synchronise with the folder once Internet connection has been established. I will also remove older backups (older than 6 months) from the synchronised cloud folder when performing maintenance at the business.

The backups of the database would also be stored on an additional medium, such as the business' portable hard drive. As automatic local and cloud backups are in effect, backups to the additional medium will only occur once I perform routine maintenance or end-user support on the premises. The portable hard drive will not be in my possession. It is the responsibility of AltHealth to keep the portable hard drive stowed away in the business' safe. I will also remove older backups (older than 6 months) from the synchronised cloud folder when performing maintenance at the business.

The system is equipped with a recovery solution in the case of an error with the database. With the version of MySQL, I have used for the database, it includes the use of the InnoDB storage engine for the database management system. InnoDB has an auto recovery solution active to recover from a potential server crash. The only requirement is to restart the database server (i.e. the system the database is hosted from) as InnoDB automatically checks the logs and performs a roll-forward of the database to the present. InnoDB automatically rolls back actions that were present at the time of the error.

For recoveries, I will perform recoveries myself through the phpMyAdmin GUI of XAMPP. Should it be deemed necessary, I would drop the entire database and use the latest backup that has been made (whether it was performed manually or automatically) to import back into a newly created “AltHealth” database.

In any instance where the system is slow, the system does not perform correctly, or the system is erroneous in data output, it is highly recommended to request for support in order to prevent any damage in the form of data corruption to the system.

Backup and recovery implementations has also been explained in the installation and administration manual of The AltHealth Database System.

# Technical Development

Within my code, I have commented throughout my code the changes and implementations I have performed. I have explained, in short, what the purpose of the section of code fulfils.

I wish to state that with regards to emails I have implemented the functionality to instead send any email that is generated to myself and not to the clients' stored email addresses. The reason for this setup is to demonstrate that the emails are being populated and being sent by making use of PHPMailer.

The system has been structured using a commonly known programming pattern known as the Model View Controller (MVC). I have implemented efficiency in my code, for example, database functions are stored in the functions.php file located in the util directory of the user. Also, the validation scripts I have created for the various forms (i.e. user inputs and actions) are also located in the util folder.

In the validation scripts mentioned in the previous paragraph, I have implemented validation algorithms with PHP in order to check for the field and to ensure that the input provided is true and correct. For example, I have implemented an algorithm for the South African ID Number which will validate if an inserted ID number contains 13 digits, numerical digits only and if the first 6 digits comply to a birthdate. Other validations include name fields, whether they contain any digits or not, and the booking slots (whether the booking slot is open or occupied). These validation algorithms and functions will be demonstrated when I attempt to insert data into the database through the designed GUI.

# Appendix A: Test Data

The following data I will use to insert and update the database.

**Client Information:**

| | |
|---|---|
| ID Number: | 9402105008085 |
| Name: | Petrus |
| Surname: | Olivier |
| Address: | 19 Lake Avenue, Sandown |
| Postal Code: | 2196 |
| Home Ph.: | 0115680478 |
| Work Ph.: | N/A |
| Cell: | 0812547578 |
| Email: | peto@sasol.co.za |
| Reference: | Website |

**Supplement Information:**

| | |
|---|---|
| Supplement ID: | VitaForce |
| Description: | 30 Capsules |
| Cost_Excl: | 65.57 |
| Cost_Incl: | 75.41 |
| Profit: | 25.00 |
| Cost_Client: | 100.41 |
| Supplier ID: | SUPPLIER J |
| Min Level: | 5 |
| Current Level: | 6 |
| NAPPI: | N/A *or* 547822 |

**Supplier Information:**

| | |
|---|---|
| Supplier ID: | SUPPLIER J |
| Contact Name: | Lisa |
| Tel: | n/a |
| Cell: | 0826547157 |
| Fax: | n/a |
| Email: | lisa@vitalagents.co.za |
| Bank: | ABSA |
| Branch Code: | 632005 |
| Acc Num: | 46258765113 |
| Acc Type: | Business |
| Comments: | This is the VitalForce representative |

**Note**: A consultation has been made for a patient today at 11:00 till 12:00, with ID Number 9506125412045

# Appendix B: Usernames and Passwords

The following users have been set up in order to access the system through the login page:

Username: hcp
Password: **@ltH_hcp01**

Username: ga
Password: **AltH@ga**

Usernames for the Database (phpMyAdmin – MySQL):

Username: root
Password: myPHP333

Username: hcp
Password: althealth@hcp

Username: ga
Password: ga@althealth