Application Name: Clock In-Out System (CIOS)

Software Method: Agile

Developer: Brandon Lu

Client: Leading Technologies Micro, Inc.

**Section 1: Clock In-Out System Information**

1.1 Overview

The CIOS is a simple, user-friendly desktop application that shall serve as a shift management system for the employees of Leading Technologies Micro, Inc. CIOS shall run on a relational cloud database. CIOS aims to be a simple, easy to use program.

1.2 Functional Requirements

In order to keep the program as simple as possible, only the most basic functions of a shift management system will be incorporated within CIOS.

1.2.1 Administrators and Employee Accounts

CIOS shall assign each employee an ID number. Each administrator will be assigned an ID number as well. Both account types will store the first and last names of the employee/admin, their ID number, a month-long record of their clock in and clock out dates and times, the amount of time that they have worked for one day, and the accumulated total amount of time that they have worked for the month. Employees shall be able to view only their own account information while Administrators shall be able to access any account.

1.2.1 Basic Functions

CIOS shall allow users to both clock in and out of their shifts. CIOS shall keep a record of the dates and times of each clock in and clock out. CIOS shall also keep track of how long an employee has worked during the current day, and also the total accumulated time worked throughout that current month. CIOS shall also reset the total time worked during the day with the end of each day.

1.2.2 Appearance

CIOS shall run on a main window. The color scheme shall be mainly neutral with warm colors. The main window shall have three buttons: Clock In, Clock Out, and Look Up and Employee. Upon selecting “Clock In” or “Clock Out”, a smaller window shall appear asking for the employee’s ID number. Once a valid ID number has been provided, the current window shall display a message confirming a successful clock in or clock out before returning to the main window. Selecting “Look Up an Employee” will result in another smaller pop up window requesting the ID number of an administrator. Upon providing a valid ID number, the current window closes to return to the main window where the three buttons shall be replaced with a list of all employee accounts. From there, the administrator shall be able to select any account and view the details of that account.

1.2.3 Interface

Users shall be offered an application in which navigation is least complicated. Icons, text, and buttons shall be appointed to an appropriate size that is accessible and legible to all users.

1.3 Specifications

1.3.1 User shall select one of three buttons from the main window.

1.3.2 Selecting “Clock In” shall redirect the user to another pop up window.

2.1 The system shall prompt the user for an ID number.

2.2 The user shall enter an ID number

2.3 The system shall search the database for an employee with the same ID number as provided.

2.4 If no such employee is found, a message alerting the user that an invalid ID number has entered is displayed

2.5 If an employee is found, the date and time are stored in their proper fields in the database and the system displays a message confirming that the user that they have been clocked in.

1.3.3 Selecting “Clock Out” shall redirect the user to another pop up window.

3.1 The system shall prompt the user for an ID number.

3.2 The user shall enter an ID number

3.3 The system shall search the database for an employee with the same ID number as provided.

3.4 If no such employee is found, a message alerting the user that an invalid ID number has entered is displayed

3.5 If an employee is found, the date and time are stored in their proper fields in the database and the system displays a message confirming that the user that they have been clocked out.

3.6 The system calculates the difference between the clock in and clock out times, adds it to the time worked that day so far and also the total time worked during that month so far, and then stores them in the proper fields in the database.

1.3.4 Selecting “Look Up an Employee” shall redirect the user to another pop up window.

4.1 The system shall prompt the use for the ID number of an administrator.

The user shall enter an ID number

4.2 The system shall search the database for an administrator with the same ID number as provided.

4.2 If no such administrator is found, a message alerting the user that an invalid ID number has entered is displayed

4.4 If an administrator is found, the system shall return the user to the main window which shall show a new layout with two textboxes prompting the user to enter the name or ID number of an employee, a button below them labeled “Search”, and a button at the bottom labeled “Display All Accounts”

4.4.1 If the user selects “Search”, the system searches the database for an account with a name or ID number matching what was entered into the text boxes.

1 If no such account is found, the system shall redirect the user to a pop up window alerting the user that no such account was found with a button below labeled “Return”.

2 If such an account is found, the system shall redirect the user to a pop up window with all information associated with that account displayed and a button at the bottom labeled “Return”

3 The user shall be redirected to the main window after selecting “Return”

4.4.2 If the user selects “Display All Accounts”, the main window will clear before displaying a list of all employee accounts. At the bottom will be a button labeled “Return”

1 Selecting an employee from the list shall redirect the user to another pop up window that shall display the information of the selected employee followed by a button labeled “Return”.

2 Selecting “Return” shall redirect the user to the main menu.

4.4.3 Selecting “Return” shall redirect the user the home window.

**Section 2: Sprint 1 Story Line**

The initial sprint of this project shall focus on the basic functions of CIOS that shall be implemented in Java code. Each account shall be represented by the Employee object. Users shall be able to clock in and out of their accounts and access any information associated with their accounts. Administrators shall be able to view any information from any account in the system. System shall be able to convert time from a String object to an int object represented in minutes. System shall be able to store each clock in and out date and time within their corresponding fields in the Employee objects. System shall be able to calculate the amount of time each employee has spent working during any individual day an individual month.

**2.1 User arrives at homepage**

All users will be greeted by the home page in the main window upon launching the application. The main window shall display the company name at the top above three vertical buttons labeled respectively “Clock In”, “Clock Out”, and “Look Up An Employee”.

Acceptance Criteria:

· User selects the program on their PC

**2.2 User clocks in**

Employees shall be able to clock into their shifts from the main menu by selecting “Clock In” which shall redirect them to a smaller popup window wish will prompt them to enter an Employee ID Number before selecting the button below it labeled “Enter”. Once clocked in, the popup window shall clear to display the message “\*Employee name\* has clocked in on \*date\* at \*time\*” above a button labeled “Return” that shall close the popup window and redirect the user to the home page on the main window once selected.

Acceptance Criteria:

· User selects “Clock In” from main menu

· User is currently clocked out

· User enters a valid Employee ID Number

· User selects “Enter”

· User selects “Return”

**2.3 User clocks out**

Employees shall be able to clock out of their shifts from the main menu by selecting “Clock Out” which shall redirect them to a smaller popup window wish will prompt them to enter an Employee ID Number before selecting the button below it labeled “Enter”. Once clocked in, the popup window shall clear to display the message “\*Employee name\* has clocked out on \*date\* at \*time\* and has worked \*dayWork\* today” above a button labeled “Return” that shall close the popup window and redirect the user to the home page on the main window once selected.

Acceptance Criteria:

· User selects “Clock Out” from main menu

· User is currently clocked in

· User enters a valid Employee ID Number

· User selects “Enter”

· User selects “Return”

**2.4 User looks up an account**

Employees shall be able to access all information associated with their own accounts from the main menu by selecting “Look Up An Account” which shall clear the main window before displaying a scrollable list of all accounts above a button labeled “Return” that shall return the main window to the homepage once selected. Before returning, the user may select any account. Upon selecting an account, the user is prompted to enter the ID number of either the account selected or any administrator’s ID number. If the user enters the correct ID Number, a popup window will appear and display all information associated with the account above a button labeled “Return” which shall close the popup window and redirect the user the main window.

Acceptance Criteria:

· User selects “Look Up An Employee” from the home page

· User selects an account from list

· User enters valid ID Number associated with selected account

· User selects “Return” to return to main window

· Use selects “Return” again to return to the home page

**Section 3: Requirements Traceability Matrix**

|  |  |  |
| --- | --- | --- |
| **SW: Software DB: Database** | | |
| **Entry #** | Requirement | Type |
| **1** | User shall be able to clock into their shift | SW |
| **2** | User shall be able to clock out of their shift | SW |
| **3** | User shall be able to view all information associated with their account | SW |
| **4** | Administrator shall be able to access any information from any account | SW |
| **5** | System shall be able to calculate time spent working during the day and during the month | SW |

**Section 4: Sprint 2 Storyline**

This sprint shall focus on the back end of the project. CIOS shall utilize Google’s Firebase database for storage of all information regarding user accounts. Utilizing Firebase as a database shall allow administrators to view any activity in the database in real-time.

**Section 5: Sprint 3 Story Line**

The third sprint of this project shall focus on programming the graphic user interface of CIOS. The entire GUI shall consist of one main window and a smaller popup window. The main window shall consist of two loadable pages: The home page including the company name at the top above three vertical buttons respectively labeled “Clock In”, “Clock Out”, and “Look Up An Employee”, and a page that displays a scrollable list of all accounts above a button labeled “Return”. The popup window shall consist of a page that displays a message prompting the user to enter the ID Number of their account above a button labeled “Enter” above another button labeled “Return”.

**6.1 Main Window**

· **6.1.1 Home page**

o Displays company name over three buttons respectively labeled “Clock In”, “Clock Out”, and “Look Up An Employee”

o Clock In

- Selecting this button shall open a popup window prompting the user to enter their ID Number. The program shall verify that it matches an ID Number in the database before displaying a message that confirms that they are clocked in above a button labeled “Return” that shall close the popup window.

o Clock Out

- Selecting this button shall open a popup window prompting the user to enter their ID Number. The program shall verify that it matches an ID Number in the database before displaying a message that confirms that they are clocked out above a button labeled “Return” that shall close the popup window.

o Look Up An Employee

- Selecting this button shall clear the homepage and load another page that shall display a scrollable list of all accounts in the database above a button labeled “Return” that upon selecting shall clear the page and load the home page. Selecting an account before returning shall open up a popup page that prompts the user to enter the associated ID Number or an administrator ID Number above two vertical buttons respectively labeled “Enter” and “Return”. If “Enter” is selected, the program shall verify that the ID Number entered matches either that of the account or of an Administrator before clearing and displaying a scrollable list of all information associated with that account. Selecting “Return” shall close the popup window.

**6.2 Popup Window**

· **Input page**

o Displays the message prompting the user to enter an ID Number above two vertical buttons respectively labeled “Enter” and “Return”

**Section 7: Requirements Traceability Matrix**

|  |  |  |
| --- | --- | --- |
| **SW: Software DB: Database** | | |
| **Entry #** | Requirement | Type |
| **1** | System shall launch homepage on main window upon being launched | SW |
| **2** | System shall redirect user to appropriate popup window after “Clock In” or “Clock Out” has been selected | SW |
| **3** | System shall redirect user to previous page after “Return” has been selected | SW |
| **4** | System shall load list page on main window after “Look Up An Employee” has been selected | SW |
| **5** | System shall redirect user to popup window once account is selected | SW |
| **6** | System shall load information page onto popup window upon verifying valid authority | SW |

**Section 8: Use Cases**

|  |  |
| --- | --- |
| **Use Case ID** | UC1 |
| **Use Case Name** | Clock In |
| **Description** | User clocks into their shift |
| **Pre-Conditions** | User is clocked out prior to clock-in |
| **Standard Flow** | 1. User selects “Clock In”  2. System prompts user for ID Number  3. User enters ID Number  4. System verifies ID Number entered matches ID Number in database  5. System retrieves current system date and time  6. System converts time into string in minutes from 00:00  7. System stores information in appropriate fields in database |
| **Post-Conditions** | User is clocked into their shift and is noted in the database under their account |
| **Open Issues** | System adds “1” digit to beginning of date retrieved |

|  |  |  |
| --- | --- | --- |
| **Use Case ID** | UC2 |  |
| **Use Case Name** | Clock Out |  |
| **Description** | User clocks out of their shift |  |
| **Pre-Conditions** | User is clocked in prior to clock-out |  |
| **Standard Flow** | 8. User selects “Clock Out”  9. System prompts user for ID Number  10. User enters ID Number  11. System verifies ID Number entered matches ID Number in database  12. System retrieves current system date and time  13. System converts time into string in minutes from 00:00  14. System retrieves clock-in time from database  15. System subtracts clock-in time from clock-out time  16. System stores information in appropriate fields in database |  |
| **Post-Conditions** | User is clocked out of their shift and is noted in the database under their account |  |
| **Open Issues** | System adds “1” digit to beginning of date retrieved |  |

**Section 9: User Stories**

Clock In

· Prompt user for ID Number

· Search database for matching ID number

- If none found, display “Invalid ID Number”

· Verify that user is currently clocked out

· Retrieve date and time from system

· Convert time from string to int in minutes from 00:00

· Store into appropriate fields in Employee object

Clock Out

· Prompt user for ID Number

· Search database for matching ID number

- If none found, display “Invalid ID Number”

· Verify that user is currently clocked in

· Retrieve date and time from system

· Convert time from string to int in minutes from 00:00

· Retrieve clock-in time from database

· Subtract clock-in time from clock-out time

· Store into appropriate fields in Employee object

Look Up An Employee

· Display list of all accounts

· After account is selected, prompt user for ID Number

- If invalid, display “Invalid ID Number”

· If ID number matches account ID Number or Administrator ID Number, display all account information