# Description of Scenario

Mr. Samuel is a general manager of a cable manufacturing company *(Refer to Appendix A: Client Interview Transcript)*. Mr. Samuel, along with other company superiors, is assigned to calculating the monthly extra wages of some employees, which may be calculated from their overtime hours.

The client explains that, everyday, employees who enter or leave the company’s premises will use a punch card machine to log their work time. At the end of every month, Mr. Samuel explains that he would gather these timecards and set aside 5 hours of work using pen-and-paper to calculate overtime hours of each employee he is assigned to supervise. Mr. Samuel also has to take time dealing with extra wages exemptions for some employees, such as those receiving a diligence pay based on absences instead of overtime, or those with fixed extra wages, and etc (*Refer to Appendix B: Summary of Client Further Consultation*). Employees who want to view their attendance hours before the month is over would have to do the same manual pen-and-paper calculations (*Refer to Appendix B*).

Mr. Samuel explains that the pen-and-paper process is “time-consuming” for both the client and employees. The client further explains that the system is prone to human error because it is manually calculated by both Mr. Samuel and individual employees. Mr. Samuel tries to minimize the errors by double-checking and relying on his memory and skills. He hopes to find a way to make the process more efficient and less error prone.

WORD COUNT: 226 words (excluding citations)

# Rationale for Proposed Product

The proposed product is a mobile application for iOS devices– iPad and iPhone– because almost all the employees assigned to Mr. Samuel has iOS devices (*Refer to Appendix A– interview transcript*). Functionality on a tablet is needed by the client because Mr. Samuel prefers to view information on his tablet (*Refer to Appendix A*). Extending functionality to phones allows employees to check their overtime hours anywhere and anytime. The program is scalable as new employees can install the app into their phones.

The proposed product will replace timecards by allowing work time to be recorded in the system and to automatically handle calculations that the client would otherwise do (*refer to Appendix B– Summary of Client Interview*). A mobile app is also suitable for recording work time because it allows a portable tablet to be stationed in the entrance/exit of the workplace for efficient checking-in/out.

The program will be developed using the Swift programming language as it allows development in iOS devices. Swift also allows Object-Oriented programming, which I will use to break problems into smaller subproblems. The integrated development environment used for developing the program will be Xcode because there are built-in features to simplify the GUI development and testing process.

A database will be needed to keep a record for Mr. Samuel to keep track of employee’s attendance and keep a history of the attendance from previous months for future reference. The Core Data persistence framework will be used because it is easily accessible on Xcode. Xcode also has a GUI feature that allows the simple creation of Core Data entities.

WORD COUNT: 247 words (excluding citations)

**TOTAL WORD COUNT (Rationale + Description): 473**

(*refer to Appendix F- Client Agreement for evidence of client interaction and consent*)

# Success Criteria For The Product

1. As the client is not the only user of the program, a secure login and logout feature is provided for all users including employees. This will allow personalized access to the system.
2. Basic client and employee information (including name, unique username, password) should be displayed to the user and stored in the database. Client information should be able to be edited by the client as desired.
3. The client should be able to view all and search for any employee’s information. This allows the client to access all employees as an administrator and allows the client to easily navigate between employees.
4. The client should be able to and should be the only one who can add a new employee into the database with their basic information and delete existing employees. This ensures that the program can process worktimes for new employees.
5. Clients should be given write access to an employee’s account information, while employees have largely read-only data. To ensure security, employees should only be able to change their passwords, while clients should be able to change an employee’s name, password, and the extra wage type of the employee (e.g. Normal pay type, Diligence pay, etc).
6. A check-in/out feature that will verify an employee’s credentials and store a timestamp belonging to the employee in the database. A client verification should be required to allow this feature to ensure that employees cannot check-in/out for work outside the vicinity of the workplace.
7. A client should be able to and should be the only one who can change, add, and delete logs for a day when an employee is present, and monthly/daily processing should be updated accordingly. This is necessary in case employees make duplicate logs for a day or forget to check-in/out.
8. The program should be able to store, process, and display the accurate work times of employees on any given day, the daily and monthly overtime hours if applicable, and/or the number of absences depending on the employee’s extra wage types. This information makes up the integral purpose of the program.
   1. Worktime for a day is the amount of worked time between a check-in and check-out log.
   2. Quota of a weekday worktime is 8 hours and a weekend is 5 hours. Hours beyond are counted as overtime hours for normal pay workers.
   3. Monthly overtime is the addition of all weekly overtimes of the month for normal pay employees. Absences count as -2 daily overtime.
9. The program should be able to cap the maximum negative number of overtime daily, weekly, and monthly hours for normal pay employees according to the company’s rules (*refer to Appendix B for more details*). This includes keeping track of absences.
   1. For normal pay employees, daily overtime is capped at -2 overtime.
   2. For normal pay employees, weekly overtime is capped at -4 hours minimum.
   3. For normal pay employees, monthly overtime is capped at 0 hours minimum.
   4. For diligence pay employees, diligence absences are capped at 3 max.
10. The client should be able to send, to his own email or another, the important information (*refer to Appendix B for details*) of employees for any month as a JSON format and file. This allows the client to cross-reference the information on his JSON viewer app with his business database’s data (*refer to Appendix B*) to determine pay.