University of Maryland University College (UMUC)

Software Development Plan and Requirement Specifications

Employee's Time management System Version 0.4

**Team Members:**

Justin Mullins

Elvin Petrosy

Ian Spooner

Wendy Velasquez Ebanks

**Professor:** Christopher Howard

**CMSC 495:** Current Trends in Computer Science

March 26, 2017

Revision History

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Date | Version | Description | Author | Signature |
| 03/14/2017 | 0.1 | Document Creation, and drafted initial outline | Ianne Spooner | Ianne Spooner |
| 03/22/2017 | 0.2 | Reformatting of the entire document, add all content to the outline previously drafted | Wendy Velasquez Ebanks | WCVE |
| 03/25/2017 | 0.2 | Added Dataflow Diagrams | Ianne Spponer | Ianne Spooner |
| 03/26/2017 | 0.3 | Added Use Cases and use case scenarios | Justin Mullins | Justin Mullins |
| 03/26/2017 | 0.4 | Added attachments, Gantt chart and made final retouch/edits | Wendy Velasquez Ebanks | WCVE |

Document Approval

|  |  |  |  |
| --- | --- | --- | --- |
| Name | Date | Version Signed | Signature |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |

**Table of Contents**

[1. Introduction 4](#_Toc478309809)

[1.1 Purpose 4](#_Toc478309810)

[1.2 Scope 4](#_Toc478309811)

[1.3 Objectives 5](#_Toc478309812)

[1.3.1 Assumptions 5](#_Toc478309813)

[1.3.2 Constraints 5](#_Toc478309814)

[1.4 Project Deliverables (Milestones) 5](#_Toc478309815)

[2. Definitions 7](#_Toc478309816)

[3. References 7](#_Toc478309817)

[4. Project Organization 8](#_Toc478309818)

[4.1 Internal Interfaces 8](#_Toc478309819)

[4.2 External Interfaces 8](#_Toc478309820)

[4.3 Roles and Responsibilities 8](#_Toc478309821)

[4.5 Communication Methodology 9](#_Toc478309822)

[4.6 Audience 9](#_Toc478309823)

[4.7 Communication Objectives 9](#_Toc478309824)

[5. Managerial Process Plan 10](#_Toc478309825)

[5.1 Work Plan 10](#_Toc478309826)

[5.2 Work Tasks Activities 10](#_Toc478309827)

[5.3 Resource Allocation 10](#_Toc478309828)

[5.4 Decision Making Process 12](#_Toc478309829)

[6. Risk Management Plan 12](#_Toc478309830)

[7. Technical process Plan 13](#_Toc478309831)

[7.1 Process Model 13](#_Toc478309832)

[7.2 Methods, Tools and Techniques 13](#_Toc478309833)

[8. General Description 14](#_Toc478309834)

[8.1 Product Perspective 14](#_Toc478309835)

[8.2 Product Functions 15](#_Toc478309836)

[8.3 User Characteristics 15](#_Toc478309837)

[9. Specific System Requirements 15](#_Toc478309838)

[9.1 External Interface Requirements 15](#_Toc478309839)

[9.1.1 User Interfaces 15](#_Toc478309840)

[9.1.2 Hardware Interfaces 16](#_Toc478309841)

[9.1.3 Software Interfaces 16](#_Toc478309842)

[9.1.4 Communications Interfaces 16](#_Toc478309843)

[9.2 Functional Requirements 16](#_Toc478309844)

[9.2.1 Input Requirements 16](#_Toc478309845)

[9.2.2 Output Requirements 16](#_Toc478309846)

[9.3 Non-Functional Requirements 17](#_Toc478309847)

[9.3.1 Security Requirements 17](#_Toc478309848)

[9.3.2 System Control Requirements 17](#_Toc478309849)

[9.3.3 Performance Requirements 18](#_Toc478309850)

[9.3.4 Business Continuity Requirements 18](#_Toc478309851)

[10. Use Cases 18](#_Toc478309852)

[10.1 Use Case #1 – Employee Punches-In 18](#_Toc478309853)

[10.2 Use Case #2 – Employee Punches-Out 18](#_Toc478309854)

[10.2 Use Case #3 – Employee Submits timesheet for supervisor approval 19](#_Toc478309855)

[10.2 Use Case #4 – Record Review 19](#_Toc478309856)

[10.2 Use Case #5 – Supervisor approves employee's timesheet 19](#_Toc478309857)

[10.2 Use Case #6 – Employee is out on sick or emergency leave 20](#_Toc478309858)

[11. Data Flow Diagrams (DFD) 20](#_Toc478309859)

[11.1 DFD Level 0 20](#_Toc478309860)

[11.2 DFD Level 1 20](#_Toc478309861)

# Introduction

The following provide detailed explanation of the planning, development and implementation process for the **Employee's Time Management System**. Furthermore, it explains the different tasks that will be accomplished through the following six weeks: the deliverables for each week and the modeling sectioned of the requirements for the system to exist.

## 1.1 Purpose

The main purpose of the Employee's Time Management Systemis to provide an interface for a user to be able to set/program their times according to their schedules so a user can register its “clock-in” and ‘clock-out” in order to record the time, date, and hours worked, and for the management to audit the times against a preset schedule. The system will provide an all-in one approach to employee’s time management mitigating the need for fragmentation or ad-hoc integration of multiple systems. This document will outline the specifications guidelines utilized in future designing of the system as well as in its implementation.

## 1.2 Scope

The Scope of the software project plan management is entailed to cover all phases of the software development life cycle for the Employee's Time Management System; this include planning, design, development, integration, testing and deployment with the specified requirements for each phase having as based functionality the following features:

* Input and authenticate user’s information
* Being able to record user’s start and end time
* Being able to calculate the estimated work hours
* Output detail work hours report
* Being able to manage different user types and privileges
* Output historical data from user’s time recorded in a maximum of two previous years.

The following items **will not** be considered as part of the scope, however they could be added in future release or an upgrade of the system:

* After implementation technical support.

## 1.3 Objectives

The main objective of the software project plan is to:

* Ensure that end users have an input into the design process, for the best unfolding of this project.

### Assumptions

The following assumptions were made during the elaboration of this document.

* The project plan may change and evolve as new information and issues are identified.
* The management will ensure that project team members are available as needed to complete tasks and objectives related to it.
* Failure to identify possible changes/issues to draft deliverables within the specified time in the timeframe previously established will result in project delays.

### Constraints

The following constrains were contemplated during the elaboration of this document.

* Resources may be limited, with no eventuality.
* Due to rules outside and probably not contemplated may cause resource availability to become inconsistent.
* The software does not work properly causing delays on implementation.

## 1.4 Project Deliverables (Milestones)

The following represent key project milestones, with estimated completion time

**Milestone Estimated Completion Date**

**Phase I:**

Project Plan Development 03/26/2017

**Phase II:**

Project Design 04/09/2017

**Phase III:**

User guide and Test Development Plan 04/02/2017

Implementation Phase I 04/16/2017

Implementation Phase II 04/23/2017

Implementation Phase III 04/30/2017

**Phase IV:**

Deployment Plan 05/07/2017

* 1. Project Timeline

The Gantt chart below displays the entire timeline for the development of the employee management system

|  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **Employee Time Management System** | | | | | | | | |
| **(Estimated Work Schedule)** | | | | | | | | |
|  |  |  |  |  |  |  |  |  |
|  | Week1 | Week 2 | Week 3 | Week 4 | Week 5 | Week 6 | Week 7 | Week 8 |
| Initiation/Planning Project |  |  |  |  |  |  |  |  |
| Plan the Project |  |  |  |  |  |  |  |  |
| Develop Project Plan |  |  |  |  |  |  |  |  |
| Design Framework |  |  |  |  |  |  |  |  |
| Design Use Case Scenarios |  |  |  |  |  |  |  |  |
| Design Database |  |  |  |  |  |  |  |  |
| Present Project Design Plan |  |  |  |  |  |  |  |  |
| Develop Test Plan |  |  |  |  |  |  |  |  |
| Develop User Guide |  |  |  |  |  |  |  |  |
| Present Test and User Guide Plan |  |  |  |  |  |  |  |  |
| Implement Framework |  |  |  |  |  |  |  |  |
| Phase I |  |  |  |  |  |  |  |  |
| Phase II |  |  |  |  |  |  |  |  |
| Phase III |  |  |  |  |  |  |  |  |
| Deployment |  |  |  |  |  |  |  |  |
| Present Consolidated Plan |  |  |  |  |  |  |  |  |
| Close the Project |  |  |  |  |  |  |  |  |

# Definitions

## 

|  |  |
| --- | --- |
| **Term** | **Definition** |
| **Punch In** | Refers to the user entering credentials in order to start a shift |
| **Punch Out** | Refers to the user entering in order to stop time tracking for the current shift. |
| **Shift** | Refers to a pre-established timed schedule that can be tracked. |
| **Super User** | Refers to the administrative user performing supervisory functions. |
| **User** | Refers to the hourly employee needing to track his or her time. |
| **Admin** | Refers to a user with privileges such that it can oversee and manipulate user and super user accounts. |

# 3. References

[1] IEEE Standard 1233-98: Guide for Developing System Requirements Specification

[2] IEEE Standard 1058-1998: Standard for Software Project Management Plans

[3] R. Darnall and J. Preston, Project management from simple to complex, 1st ed. The Saylor Academy, 2010.

[4] D. Levy, "Use Case Examples -- Effective Samples and Tips", Gatherspace.com, 2014. [Online]. Available: http://www.gatherspace.com/static/use\_case\_example.html

# 4. Project Organization

This section delineates how the project team is organized for both the development of the internal and external interfaces of the system. The project team is designed to be in a small nature to enforce collaboration and commitment among team members as well as ensure effective communication between team members

## 4.1 Internal Interfaces

The internal interfaces for the elaboration of the system will be modeled and specified in further in later sections of this document.

## 4.2 External Interfaces

Section 9.1 External Interface Requirements provide some of the specifics that will be address with the external interfaces.

## 4.3 Roles and Responsibilities

The responsibilities for each team member and his/her role are described in the following table:

|  |  |  |  |
| --- | --- | --- | --- |
| **Role** | **Responsibilities** | | **Participant(s)** |
| \*\*Project Manager | 1. Manages project in accordance to the project plan 2. Serves as liaison to the Steering Committee 3. Provide overall project direction 4. Direct/lead team members toward project objectives 5. Handle problem resolution | | WCVE |
| **Project Participants:** ensure they understand user needs and business processes; will act as consumer advocate in representing their area; they will communicate project goals, status and progress throughout the project to personnel, and will review and approve project deliverables; they are responsible to coordinate participation of work groups, individuals and stakeholders. They provide knowledge and recommendations, and Help to identify and remove project barriers as well as identifying risks and issues and help in the resolution. They are responsible for the quality of products and that meets project goals and objectives. | | | |
| Software Dev. | | Assists with back-end dev and DB | Justin Mullins  Elvin Petrosi  Ian Spooner  Wendy Velasquez Ebanks |
| Web App. Dev. | | Assists with front-end development of the application. Also, testing and editing. | Elvin Petrosi (Main)  [epetrosy@gmail.com](mailto:epetrosy@gmail.com) |
| Document main editor | | Manages all the team's documents, and is in charge of consolidating and finalizing each deliverable.  Assist with DB, dev testing, back-end dev, and document management. | Wendy Velasquez Ebanks (Main)  [wendy.velasquezebanks@gmail.com](mailto:wendy.velasquezebanks@gmail.com) |
| Editors | | Performs a final quality review and assure that the consolidated document reads as if one person prepared it. | Ian Spooner  Elvin Petrosi  Justin Mullins |
| Software Developer/ modeling designer | | Assists with back-end development of the application, logistics, testing, and editing. | Ian Spooner (Main)  [cawwot@gmail.com](mailto:cawwot@gmail.com) |
| DBA | | Lend expertise and guidance as needed in regards of the information required on each deliverable of the project. | Justin Mullins (main)  [shadowx787@gmail.com](mailto:shadowx787@gmail.com)  Wendy Velasquez Ebanks |
| Software Tester | | Ensure deliverable products meet the established requirements initially specified. | Justin Mullins  Elvin Petrosi  Ian Spooner  Wendy Velasquez Ebanks |

*\*\* The role may change based on expertise and deliverable*

## 4.5 Communication Methodology

During the project, it is important to that the IT project team provides accurate, relevant, and timely project information to the stakeholders. With effective communication, the project can accomplish the requirements of the project with support from all stakeholders. Per Project Management from Simple to Complex, “Effective communication on a project is critical to project success.”[3].

## 4.6 Audience

The intended audience is the project manager, project team, project sponsor, leaders, and stakeholders who will be relied upon for the successful completion of this project.

## 4.7 Communication Objectives

Our objectives for communication are open and effective communications to ensure the success of the project. The project team will:

* Give timely information about the project
* Give accurate and details information about the project
* Strive for constant and consistent communication

# 5. Managerial Process Plan

## 5.1 Work Plan

The expected work plan for the development of the Time Management System Log follows the work break down structure below from the planning phase up to its deployment:



## 5.2 Work Tasks Activities

The tasks activities for the team performance throughout the entire Software Development Life Cycle are detailed on daily basis schedule by week period in the excel spreadsheet attached below:



## 5.3 Resource Allocation

The resource allocation describes the amount of work that will need to be performed by the team in order to cover all phases of development and test of the software application created for the Employee's Time Management Log System, aligning with the major work activities identified depending on role and the necessary skills of the personnel involve on each activity.

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| COCOMO Mode | Project Type | Team Size | Development Environment | Constraints and Deadlines |
| Organic | Transactional | Small | Stable (in-house) | Tight |

|  |  |  |
| --- | --- | --- |
| **Functions** |  | **Estimated LOC** |
| User Interface | UI | 500 |
| Database | DB | 350 |
| Database Admin. Interface | DBI | 300 |
| Existing Bug Fixing | TEST | 250 |
| Total Estimated Lines of code |  | 1400 |

By the COCOMO model, the use of organic mode can be used for further analysis. This process provides an effort equation in the form of:

Where under the ‘Organic’ COCOMO mode a = 2.4 and b = 1.05. A preliminary estimation of the source lines of code (SLOC) in thousands of units (KLOC) is determined to be 1.4k, or ~1400 lines of code. Through this equation the following result is obtain for estimation on human effort:

After the estimated human effort is being calculated, the project must correlate with an an approximate estimate in time that will take staff. The COCOMO method for the ‘Organic’ mode provides the equation as:

The estimated time on human effort becomes:

The formulas provide an idea and estimation of the complexity that could take the development based on the ratio of estimated human effort versus project staff time in months through the following equation:

This equation provides an estimate of how many developers should be needed

As a result, the calculation gives an estimate that at least two full time software developers should be in charge, in addition to staff providing customer liaison and project oversight functions, as well as user interface design; this matches precisely with mechanics required for full implementation of the Time Management Log System. In addition, the approximation of 6.06 months for development is roughly equivalent.

## 5.4 Decision Making Process

The decisions related to the each project deliverable will be done democratically by vote as all projects participants have previously discussed and agreed with this type of system. If the vote results in a tie, the project leader will make the final decision.

# 6. Risk Management Plan

The plan attempts to prioritize and document an approach to risks that have been identified and acknowledged prior to the start of the project. This plan will monitor and updated throughout the life of this project; modification approaches must be agreed by the entire team group. The table below depicts some of the risks contemplated with anticipation.

|  |  |  |
| --- | --- | --- |
| **Risk Ranking (High, Medium, Low)** | **Risk Description and Impact** | **Mitigation Strategy and/or Contingency Plan** |
| Medium | *Unclear understanding of requirements* | In order to address it the team will meet on a daily basis to discuss and debrief on the approach and how the functions target specific requirements of the Employee's Time Management Log System are being developed. |
| Medium | *Late Delivery of Software* | To minimize this risk a detailed schedule was elaborated with respective dates, and specifics as to who/whom will have to present that portion as well as having a backup person for each role, and in the case that someone cannot meet the specified deadline, the issue will be discuss and re-assigned during our daily meeting. |
| Medium | *Loss of project related documents* | Centralized document maintenance and version control software, GitHub repository is used to ensure proper document management. GitHub is Web based which ease accessibility and use to all team members to check out, modify, or upload documents. GitHub repository is also utilized as the team’s bug tracking repository, where any observed bug or change request was tracked. This permitted the testers and developers that are located in different time zone locations a collaborate environment in real-time. |
| High | *Equipment Failure* | In order to reduce this risk the team will have a person designated to make sure all equipment is working properly. |
| High | *Changes in requirements* | Changing the requirements in the middle of implementation may cause major issues in the entire system, and in order to approach this situation the meetings will be hold daily, to do clarifications and review feedback ensuring that the changes are within the scope of what was previously discussed when agreeing on the requirements. |
| Low | *Poor Commenting/ documentation of source code* | To reduce this inconvenient, it is required that developers document their source code, so anyone that come later and modifies a function or class related to the system will know the properties and the purpose. Also, it will help tester to prepare the different types of tests. |

# 7. Technical process Plan

## Process Model

This section describes methods, tools and techniques that will be used in the development and implementation of the Employee's Time Management Log System. The organization of this process is to ensure that through an audit process the software development process outlined in this plan is followed through.

## Methods, Tools and Techniques

The development methodology is based on a combination of use cases. The Software configuration summarizes the approach ensuring that all software is developed and controlled by established configuration best practices, standards and procedures. The following tools are used by the development team of the Employee's Time Management Log System:

* JDK for Eclipse: The Java programing language Development kit for the "back end" of software development, compilation, and testing.
* Mozilla Firefox browser Version 29.0 and IE version 9.0 or greater
* Notepad++ for HTML as part of the front-end development
* JavaScript for front-end development
* MySQL for the development of the Database as part of the back-end development with Open SSL to certify security of the information the database will hold in it.
* Adobe Acrobat 9.8 for the reports generated

# 8. General Description

The requirements are organized and presented based on functionalities illustrated in the activity context diagram.

## 8.1 Product Perspective

C:\Users\Wendy\Downloads\CMSC495 Diagrams(1).png

Figure 1 ACD for Employee Time Management System

These specifications will then be refined to be presented as activities performed portraying the intended actions or processes through different types of use case scenarios to illustrate the functionalities of the system. The requirements are presented by categories facilitating the process of mapping design and test for each feature contained in the system.

## 8.2 Product Functions

Product functionality for can be divided in the following sections:

**8.2.1** User verification and authentication into the system

**8.2.2** User’s time recording for hours worked

**8.2.3** User’s submission of hours worked for a specific timeframe

**8.2.4** User’s ability to retrieve historical data

**8.2.4** Super User’s ability to approve timesheets for multiple users

**8.2.5** Admin ability to create, modifies rights, or removes a user account from the system

## 8.3 User Characteristics

Users can only view their own time sheet data, super users can view their own data as well as data of users under their rights of privilege to approve time sheets, Admins can oversee all accounts for users, and super users besides their own.

# 9. Specific System Requirements

## 9.1 External Interface Requirements

This section is intended to specify the requirements for interaction between hardware, software, and the user, which will be detailed during the design implementation of the system

### 9.1.1 User Interfaces

The system should have the following interfaces:

1. Log in screen
2. Main Menu with different options such as: Account settings, report retrieval on past timesheets.
3. Time sheet setting and input
4. Details of the account screen

### 9.1.2 Hardware Interfaces

Hardware interface requirements for the Employee Time Management System are not applicable since the requirements once is implemented are very universal due to its web interface.

### 9.1.3 Software Interfaces

Software interfaces for the Employee Time Management System are mainly through the use of a browser.

### 9.1.4 Communications Interfaces

The communication interface for the Employee Time Management System is mainly for internal communication between the database and the web application from where the input is going to be retrieve and stored in the Database.

## 9.2 Functional Requirements

### 9.2.1 Input Requirements

**IR-1** The user shall be authenticated by inputting its credentials when log-in to the system.

**IR-2** The user shall be able to input his/her hours according to the schedule previously set and submit them for review.

**IR-3** User data shall be validated when entered in order to avoid duplications and reinforce integrity.

**IR-4** The system shall request the necessary information from the user in order to generate historical and current reports for the user.

### 9.2.2 Output Requirements

**OR -1** The System shall be able to generate soft and copy reports to the user.

**OR-2** The system shall be able to provide a confirmation of work hours submission.

**OR-3** The system shall be able to generate a report based on current hours marked, and remaining hours to mark.

**OR-4** The system shall provide a summary of hours worked by day and throughout the week.

**OR-5** The system shall be able to retrieve historical data from the user for up to two years and provide it in a form of report.

## 9.3 Non-Functional Requirements

### 9.3.1 Security Requirements

**SR-1** The system shall be able to authenticate a user by verifying if his/her account exists in the system.

**SR-2** The system shall use secure sockets in all transactions that include any confidential employee information stored in the database.

**SR-3** The system shall be able to provide different accesses to it such as user, super-user and admin.

**SR-4** The database shall use a Transport Security Layer (TSL) to ensure connections between the client and server so that no user data transfers are unencrypted.

**SR-5** Password Policies shall be in place and enforced through the system with the use of strong passwords.

**SR-6** Database access shall follow the principal of least privilege by providing access to only the data users are required in order to perform their work.

**SR-7** Database access will follow the principal of least privilege by providing access to only the data users are required in order to perform their work.

### 9.3.2 System Control Requirements

**SCR-1** A user shall be added into System’s log in order to use the system.

**SCR-2** Accessibility shall be granted based on the user’s role.

**SCR-3** The system shall restrict access to the server to only hosts that require actual access.

**SCR-4** The system shall create an error log file that includes the error type, description, and time it happened.

### 9.3.3 Performance Requirements

**PR-1** The system shall provide options to super-user in order to set flexible shifts to the user.

**PR-2** The system shall provide for replication of the database.

**PR-3** The system shall be able to provide portability due to its design and web interface.

### 9.3.4 Business Continuity Requirements

**BCR-1** Any change to the system design shall be documented for reference in case an unexpected failure.

**BCR-2** Testing shall be performed every time a change or upgrade in the system occurs.

# 10. Use Cases

## 10.1 Use Case #1 – Employee Punches-In

* + - Employee logs into the system
    - Employee clicks the punch-in button -> arrival time is recorded in the database.

## 10.2 Use Case #2 – Employee Punches-Out

**-** Employee logs into the system

**-** When leaving for the day the employee clicks the punch-out button -> departure time is recorded in the database.

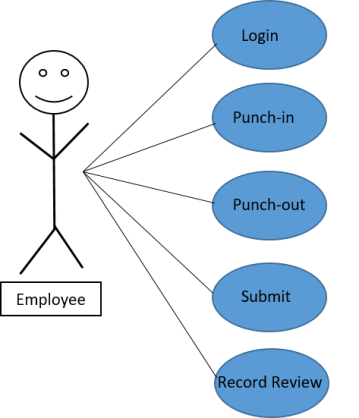


Figure 2 displays Use Case 1 and use case 2

## 10.2 Use Case #3 – Employee Submits timesheet for supervisor approval

* Employee logs into the system
* Employee verifies data in their timesheet is correct.
* Employee submits their timesheet for supervisor approval by clicking the submit timesheet button. -> Employee's timesheet is marked for supervisor review in the database.

## 10.2 Use Case #4 – Record Review

* Employee or Supervisor logs into the system (Supervisors can view employees historical timesheets)
* Select the pay period from which you’d like to view the record.
* Click the Fetch Timesheet button to view the timesheet -> load the timesheet from the database for the specified pay period.

## 10.2 Use Case #5 – Supervisor approves employee's timesheet

* Supervisor Logs into the system
* Supervisor will have notifications alerting them that employee timesheets are awaiting approval
* Supervisor will click on an employee's data to view their timesheet
* If timesheet is correct they click the approve button -> the hours worked are then sent to the finance office for payment distribution.
* If timesheet is wrong the supervisor will click the return to sender button as well as submit clarification to the issue with the timesheet -> The timesheet will be kicked back to the employee for corrections.

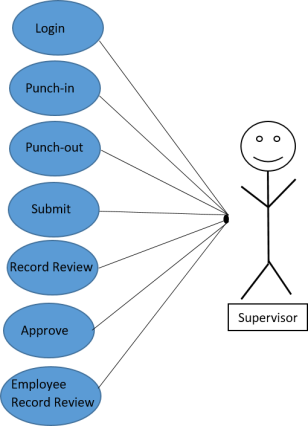


Figure 3 Use Case 5 representation

## 10.2 Use Case #6 – Employee is out on sick or emergency leave

* Supervisor logs into the system
* The supervisor will enter the appropriate leave hours in the system on the employee's behalf -> leave hours are recorded in the database.

# 11. Data Flow Diagrams (DFD)

## 11.1 DFD Level 0

C:\Users\Wendy\Downloads\DFD(1).png

Figure 4 DFD Level 0 for Employee Time Management System

## 11.2 DFD Level 1

C:\Users\Wendy\Downloads\DFD(3).png

Figure 5 DFD Level 1 for Employee Time Management System