College of Computer Science & Technology

2017 Software Engineering Practice Project

2017.5.23 - 2017.6.13

Payroll Requirements

Problem Statement

As the head of Information Technology at Acme, Inc., you are tasked with building a new payroll system to replace the existing system, which is hopelessly out of date. Acme needs a new system to allow employees to record timecard information electronically and automatically generate paychecks based on the number of hours worked and total amount of sales (for commissioned employees).

The new system will be state of the art and will have a Windows-based desktop interface to allow employees to enter timecard information, enter purchase orders, change employee preferences (such as payment method), and create various reports. The system will run on individual employee desktops throughout the entire company. For reasons of security and auditing, employees can only access and edit their own timecards and purchase orders.

The system will retain information on all employees in the company (Acme currently has around 5,000 employees world-wide). The system must pay each employee the correct amount, on time, by the method that they specify (see possible payment methods described later). Acme, for cost reasons, does not want to replace one of their legacy databases, the Project Management Database, which contains all information regarding projects and charge numbers. The new system must work with the existing Project Management Database, which is a DB2 database running on an IBM mainframe. The Payroll System will access, but not update, information stored in the Project Management Database.

Some employees work by the hour, and they are paid an hourly rate. They submit timecards that record the date and number of hours worked for a particular charge number. If someone works for more than 8 hours, Acme pays them 1.5 times their normal rate for those extra hours. Hourly workers are paid every Friday.

Some employees are paid a flat salary. Even though they are paid a flat salary, they submit timecards that record the date and hours worked. This is so the system can keep track of the hours worked against particular charge numbers. They are paid on the last working day of the month.

Some of the salaried employees also receive a commission based on their sales. They submit purchase orders that reflect the date and amount of the sale. The commission rate is determined for each employee, and is one of 10%, 15%, 25%, or 35%.

One of the most requested features of the new system is employee reporting. Employees will be able to query the system for number of hours worked, totals of all hours billed to a project (i.e., charge number), total pay received year-to-date, remaining vacation time, etc.

Employees can choose their method of payment. They can have their paychecks mailed to the postal address of their choice, or they can request direct deposit and have their paycheck deposited into a bank account of their choosing. The employee may also choose to pick their paychecks up at the office.

The Payroll Administrator maintains employee information. The Payroll Administrator is responsible for adding new employees, deleting employees and changing all employee information such as name, address, and payment classification (hourly, salaried, commissioned), as well as running administrative reports.

The payroll application will run automatically every Friday and on the last working day of the month. It will pay the appropriate employees on those days. The system will be told what date the employees are to be paid, so it will generate payments for records from the last time the employee was paid to the specified date. The new system is being designed so that the payroll will always be generated automatically, and there will be no need for any manual intervention.

Glossary

Introduction

This document is used to define terminology specific to the problem domain, explaining terms, which may be unfamiliar to the reader of the use-case descriptions or other project documents. Often, this document can be used as an informal *data dictionary*, capturing data definitions so that use-case descriptions and other project documents can focus on what the system must do with the information.

Definitions

The glossary contains the working definitions for the key concepts in the Payroll System.

Bank System

Any bank(s) to which direct deposit transactions are sent.

Employee

A person that works for the company that owns and operates the payroll system (Acme, Inc.)

Payroll Administrator

The person responsible for maintaining employees and employee information in the system.

Project Management Database

The legacy database that contains all information regarding projects and charge numbers.

System Clock

The internal system clock that keeps track of time. The internal clock will automatically run the payroll at the appropriate times.

Pay Period

The amount of time over which an employee is paid.

Paycheck

A record of how much an employee was paid during a specified Pay Period.

Payment Method

How the employee is paid, either pick-up, mail, or direct deposit.

Timecard

A record of hours worked by the employee during a specified pay period.

Purchase Order

A record of a sale made by an employee.

Salaried Employee

An employee that receives a salary.

Commissioned Employee

An employee that receives a salary plus commissions.

Hourly Employee

An employee that is paid by the hour.

Supplementary Specification

Objectives

The purpose of this document is to define requirements of the Payroll System. This Supplementary Specification lists the requirements that are not readily captured in the use cases of the use-case model. The Supplementary Specifications and the use-case model together capture a complete set of requirements on the system.

Scope

This Supplementary Specification applies to the Payroll System, which will be developed by the OOAD students.

This specification defines the non-functional requirements of the system; such as reliability, usability, performance, and supportability as well as functional requirements that are common across a number of use cases. (The functional requirements are defined in the Use Case Specifications.).

References

None.

Functionality

None.

Usability

None.

Reliability

The main system must be running 98% of the time. It is imperative that the system be up and running during the times the payroll is run (every Friday and the last working day of the month).

Performance

The system shall support up to 2000 simultaneous users against the central database at any given time, and up to 500 simultaneous users against the local servers at any one time.

Supportability

None.

Security

The system should prevent employees from changing any timecards other than their own. Additionally, for security reasons, only the Payroll Administrator is allowed to change any employee information with the exception of the payment delivery method.

Design Constraints

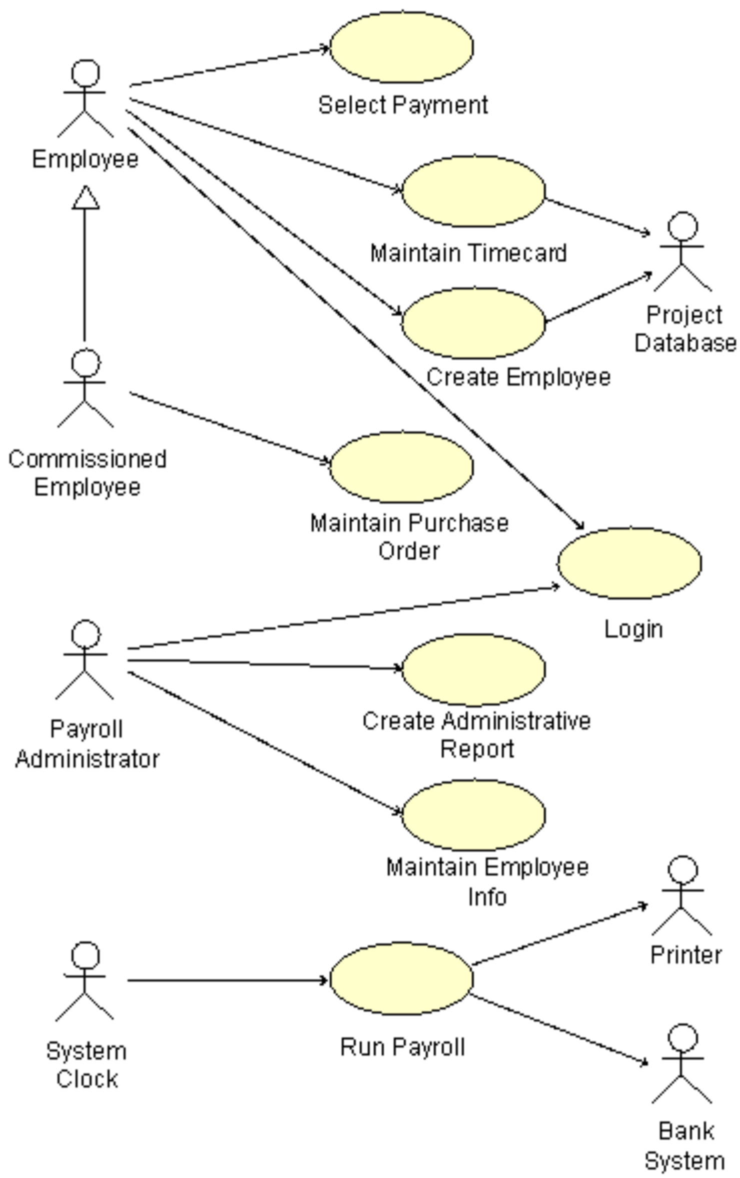
The system shall integrate with an existing legacy system, the Project Management Database, which is a DB2 database running on an IBM mainframe.

The system shall interface with existing bank systems via an electronic transaction interface (NOTE: THE FORMAL INTERFACES WITH THE EXTERNAL BANK SYSTEM WOULD NEED TO BE DEFINED EARLY IN THE PROCESS AND DEFINED HERE OR IN A SEPARATE SUPPORTING DOCUMENT. SUCH A DEFINITION IS OUT OF THE SCOPE OF THIS COURSE.)

The system shall provide a Windows-based desktop interface.

Use-Case Model

Payroll System Use-Case Model Main Diagram



Create Administrative Report

Brief Description

The use case allows the Payroll Administrator to create either a "Total Hours Worked" or "Pay Year-to-Date" report.

Flow of Events

Basic Flow

The use case begins when the Payroll Administrator requests that the system create an administrative report.

1. The system requests that the Payroll Administrator specify the following report criteria:

   - Report Type (either total hours worked or pay year-to-date),
   - Begin and end dates for the report,
   - Employee name(s)

2. Once the Payroll Administrator provides the requested information, the system provides the Payroll Administrator with a report satisfying the report criteria.
3. The Payroll Administrator may then request that the system save the report. At which time, the system requests the Payroll Administrator to provide the name and location for saving the report.
4. Once the Payroll Administrator provides the requested information and confirms the decision to save the report, the system saves the report to the specified name and location.
5. If the Payroll Administrator did not elect to save the report, the report is discarded.

Alternative Flows

Requested Information Unavailable

If in the **Basic Flow**, the requested information is unavailable, the system will display an error message. The Payroll Administrator can choose to either return to the beginning of the **Basic Flow,** or cancel the operation, at which point the use case ends.

Invalid Format or Insufficient Information

If, in the **Basic Flow**, the Payroll Administrator has not specified sufficient information to create the selected report, the system will prompt the actor for the missing information. The Payroll Administrator can either enter the missing information or choose to cancel the operation, at which point the use case ends.

Special Requirements

None.

Pre-Conditions

The Payroll Administrator must be logged onto the system in order for this use case to begin.

Post-Conditions

The system state is unchanged by this use case.

Extension Points

None.

Create Employee Report

Brief Description

The use case allows the Employee to create a “Total Hours Worked,” “Total Hours Worked for a Project”, “Vacation/Sick Leave,” or “Total Pay Year-to-Date” report.

Flow of Events

Basic Flow

This use case starts when the Employee wishes to create a “Total Hours Worked,” “Total Hours Worked for a Project”, “Vacation/Sick Leave,” or “Total Pay Year-to-Date” report.

1. The system requests that the Employee specify the following report criteria:

   - Report Type (either “Total Hours Worked,” “Total Hours Worked for a Project”, “Vacation/Sick Leave,” or “Total Pay Year-to-Date”)
   - Begin and end dates for the report

2. If the Employee selected the “Total Hours Worked for a Project” report, the system retrieves and displays a list of the available charge numbers from the Project Management Database. The system then requests that the Employee select a charge number.
3. Once the Employee provides the requested information, the system provides the Employee with a report satisfying the report criteria.
4. The Employee may then request that the system save the report. At which time, the system requests the Employee to provide the name and location for saving the report.
5. Once the Employee provides the requested information and confirms the decision to save the report, the system saves the report to the specified name and location.
6. If the Employee did not elect to save the report, the report is discarded.

Alternative Flows

Requested Information Unavailable

If, in the **Basic Flow**, the requested information is unavailable, the system will display an error message. The Employee can choose to either return to the beginning of the **Basic Flow,** or cancel the operation, at which point the use case ends.

Invalid Format or Insufficient Information

If, in the **Basic Flow**, the Employee has not specified sufficient information to create the selected report, the system will prompt the actor for the missing information. The Employee can either enter the missing information or choose to cancel the operation, at which point the use case ends.

Special Requirements

None.

Pre-Conditions

The Employee must be logged onto the system before this use case begins.

Post-Conditions

The system state is unchanged by this use case.

Extension Points

None.

Login

Brief Description

This use case describes how a user logs into the Payroll System.

Flow of Events

Basic Flow

This use case starts when the actor wishes to Login to the Payroll System.

1. The actor enters his/her name and password.
2. The system validates the entered name and password and logs the actor into the system.

Alternative Flows

Invalid Name/Password

If, in the **Basic Flow**, the actor enters an invalid name and/or password, the system displays an error message. The actor can choose to either return to the beginning of the **Basic Flow** or cancel the login, at which point the use case ends.

Special Requirements

None.

Pre-Conditions

The system is in the login state and has the login screen displayed.

Post-Conditions

If the use case was successful, the actor is now logged into the system. If not, the system state is unchanged.

Extension Points

None.

Maintain Employee Information

Brief Description

This use case allows the Payroll Administrator to maintain employee information. This includes adding, changing, and deleting employee information from the system.

Flow of Events

Basic Flow

This use case starts when the Payroll Administrator wishes to add, change, and/or delete employee information from the system.

1. The system requests that the Payroll Administrator specify the function he/she would like to perform (either Add an Employee, Update an Employee, or Delete an Employee)
2. Once the Payroll Administrator provides the requested information, one of the sub flows is executed.
   If the Payroll Administrator selected "Add an Employee", the **Add an Employee** subflow is executed.
   If the Payroll Administrator selected "Update an Employee", the **Update an Employee** subflow is executed.
   If the Payroll Administrator selected "Delete an Employee", the **Delete an Employee** subflow is executed.

Add an Employee

1. The system requests that the Payroll Administrator enter the employee information. This includes:
2. name
   - employee type (hour, salaried, commissioned)
   - mailing address
   - social security number
   - standard tax deductions
   - other deductions (401k, medical)
   - phone number
   - hourly rate (for hourly employees)
   - salary (for salaried and commissioned employees)
   - commission rate (for commissioned employees)
   - hour limit (some employees may not be able to work overtime)
3. Once the Payroll Administrator provides the requested information, the system generates and assigns a unique employee id number to the employee and sets the paycheck delivery method to default of "pickup". The employee is added to the system.
4. The system provides the Payroll Administrator with the new employee id.

Update an Employee

1. The system requests that the Payroll Administrator enter the employee id.
2. The Payroll Administrator enters the employee id. The system retrieves and displays the employee information.
3. The Payroll Administrator makes the desired changes to the employee information. This includes any of the information specified in the Add an Employee sub-flow.
4. Once the Payroll Administrator updates the necessary information, the system updates the employee record with the updated information.

Delete an Employee

1. The system requests that the Payroll Administrator specify the employee id.

2. The Payroll Administrator enters the employee id. The system retrieves and displays the employee information.
3. The system prompts the Payroll Administrator to confirm the deletion of the employee.
4. The Payroll Administrator verifies the deletion.
5. The system marks the employee record for deletion. The next time the payroll is run, the system will generate a final paycheck for the deleted employee and remove the employee from the system.

Alternative Flows

Employee Not Found

If in the **Update an Employee or Delete an Employee** sub-flows, an employee with the specified id number does not exist, the system displays an error message. The Payroll Administrator can then enter a different id number or cancel the operation, at which point the use case ends.

Delete Cancelled

If in the **Delete An Employee** sub-flow, the Payroll Administrator decides not to delete the employee, the delete is cancelled and the **Basic Flow** is re-started at the beginning.

Special Requirements

None.

Pre-Conditions

The Payroll Administrator must be logged onto the system before this use case begins.

Post-Conditions

If the use case was successful, the employee information is added, updated, or deleted from the system. Otherwise, the system state is unchanged.

Extension Points

None.

Maintain Purchase Order

Brief Description

This use case allows a Commissioned Employee to record and maintain purchase orders. This includes adding, changing, and deleting purchase orders. Commissioned employees must record each of their purchase orders in order to receive commissions.

Flow of Events

Basic Flow

This use case starts when the Commissioned Employee wishes to add, change, and/or delete purchase order information from the system.

1. The system requests that the Commissioned Employee specify the function he/she would like to perform (either Create a Purchase Order, Update a Purchase Order, or Delete a Purchase Order)
2. Once the Commissioned Employee provides the requested information, one of the sub flows is executed.
   If the Commissioned Employee selected "Create a Purchase Order ", the **Create a Purchase Order** subflow is executed.
   If the Commissioned Employee selected "Update a Purchase Order ", the **Update a Purchase Order** subflow is executed.
   If the Commissioned Employee selected "Delete a Purchase Order ", the **Delete a Purchase Order** subflow is executed.

Create a Purchase Order

1. The system requests that the Commissioned Employee enter the purchase order information. This includes:

   - customer point of contact
   - customer billing address
   - product(s) purchased
   - date
2. Once the Commissioned Employee provides the requested information, the system generates and assigns a unique purchase order number to the purchase order. The purchase order is added to the system for the Commissioned Employee.
3. The system provides the Commissioned Employee with the new purchase order id.

Update a Purchase Order

1. The system requests that the Commissioned Employee enter the purchase order id.
2. The Commissioned Employee enters the purchase order id.
3. The system retrieves the purchase order associated with the purchase order id.
4. The system verifies that the purchase order is a purchase order for the Commissioned Employee, and that the purchase order is open.
5. The system displays the purchase order.
6. The Commissioned Employee makes the desired changes to the purchase order information. This includes any of the information specified in the **Create a Purchase Order** sub flow.
7. Once the Commissioned Employee updates the necessary information, the system updates the purchase order with the updated information.

Delete a Purchase Order

1. The system requests that the Commissioned Employee specify the purchase order id.
2. The Commissioned Employee enters the purchase order id.
3. The system retrieves the purchase order associated with the purchase order id.
4. The system verifies that the purchase order is a purchase order for the Commissioned Employee, and that the purchase order is open.
5. The system displays the purchase order.
6. The system prompts the Commissioned Employee to confirm the deletion of the purchase order.
7. The Commissioned Employee verifies the deletion.
8. The system removes the purchase order from the system.

Alternative Flows

Purchase Order Not Found

If, in the **Update a Purchase Order or Delete an Purchase Order** sub-flows, an purchase order with the specified id number does not exist, the system displays an error message. The Commissioned Employee can then enter a different id number or cancel the operation, at which point the use case ends.

Invalid Access to a Purchase Order

If, in the **Update a Purchase Order or Delete a Purchase Order** sub-flows, the Commissioned Employee attempts to access a purchase order that is not his, the system displays an error message. The Commissioned Employee can then enter a different id number or cancel the operation, at which point the use case ends.

Purchase Order is Closed

If, in the **Update a Purchase Order or Delete a Purchase Order** sub-flows, the Commissioned Employee attempts to access a purchase order that is closed, the system displays an error message. The Commissioned Employee can then enter a different id number or cancel the operation, at which point the use case ends.

Delete Cancelled

If, in the **Delete A Purchase Order** sub-flow, the Commissioned Employee decides not to delete the purchase order, the delete is cancelled and the **Basic Flow** is re-started at the beginning.

Special Requirements

None.

Pre-Conditions

The Commissioned Employee must be logged onto the system before this use case begins.

Post-Conditions

If the use case was successful, the purchase order information is added, updated, or deleted from the system. Otherwise, the system state is unchanged.

Extension Points

None.

Maintain Timecard

Brief Description

This use case allows the Employee to update and submit timecard information. Hourly and salaried employees must submit weekly timecards recording all hours worked that week and which projects the hours are billed to. An Employee can only make changes to the timecard for the current pay period and before the timecard has been submitted.

Flow of Events

Basic Flow

This use case starts when the Employee wishes to enter hours worked into his current timecard.

1. The system retrieves and displays the current timecard for the Employee. If a timecard does not exist for the Employee for the current pay period, the system creates a new one. The start and end dates of the timecard are set by the system and cannot be changed by the Employee.
2. The system retrieves and displays the list of available charge numbers from the Project Management Database.
3. The Employee selects the appropriate charge numbers and enters the hours worked for any desired date (within the date range of the timecard).
4. Once the Employee has entered the information, the system saves the timecard.

Submit Timecard

1. At any time, the Employee may request that the system submit the timecard.
2. At that time, the system assigns the current date to the timecard as the submitted date and changes the status of the timecard to “submitted.” No changes are permitted to the timecard once it has been submitted.
3. The system validates the timecard by checking the number of hours worked against each charge number. The total number of hours worked against all charge numbers must not exceed any limit established for the Employee (for example, the Employee may not be allowed to work overtime).
4. The system retains the number of hours worked for each charge number in the timecard.
5. The system saves the timecard.
6. The system makes the timecard read-only, and no further changes are allowed once the timecard is submitted.

Alternative Flows

Invalid Number of Hours

If, in the **Basic Flow,** an invalid number of hours is entered for a single day (>24), or the number entered exceeds the maximum allowable for the Employee, the system will display an error message and prompt for a valid number of hours. The Employee must enter a valid number, or cancel the operation, in which case the use case ends.

Timecard Already Submitted

If, in the **Basic Flow,** the Employee’s current timecard has already been submitted, the system displays a read-only copy of the timecard and informs the Employee that the timecard has already been submitted, so no changes can be made to it. The Employee acknowledges the message and the use case ends.

Project Management Database Not Available

If, in the **Basic Flow,** the Project Management Database is not available, the system will display an error message stating that the list of available charge numbers is not available. The Employee acknowledges the error and may either choose to continue (without selectable charge numbers), or to cancel (any timecard changes are discarded and the use case ends).
Note: Without selectable charge numbers, the Employee may change hours for a charge number already listed on the timecard, but he/she may not add hours for a charge number that is not already listed.

Special Requirements

None.

Pre-Conditions

The Employee must be logged onto the system before this use case begins.

Post-Conditions

If the use case was successful, the Employee timecard information is saved to the system. Otherwise, the system state is unchanged.

Extension Points

None.

Run Payroll

Brief Description

The use case describes how the payroll is run every Friday and the last working day of the month.

Flow of Events

Basic Flow

1. The use case begins when it's time to run the payroll. The payroll is run automatically every Friday and the last working day of the month.
2. The system retrieves all employees who should be paid on the current date.
3. The system calculates the pay using entered timecards, purchase orders, employee information (e.g., salary, benefits, etc.) and all legal deductions.
4. If the payment delivery method is mail or pick-up, the system prints a paycheck.
5. If the payment delivery method is direct deposit, the system creates a bank transaction and sends it to the Bank System for processing.
6. The use case ends when all employees receiving pay for the desired date have been processed.

Alternative Flows

Bank System Unavailable

If the Bank System is down, the system will attempt to send the bank transaction again after a specified period. The system will continue to attempt to re-transmit until the Bank System becomes available.

Deleted Employees

After the payroll for an Employee has been processed, if the employee has been marked for deletion (see the **Maintain Employee** use case), then the system will delete the employee.

Special Requirements

None.

Pre-Conditions

None.

Post-Conditions

Payments for each employee eligible to be paid on the current date have been processed.

Extension Points

None.

Select Payment Method

Brief Description

This use case allows an Employee to select a payment method. The payment method controls how the Employee will be paid. The Employee may choose to either: pick up his check directly, receive it in the mail, or have it deposited directly into a specified bank account.

Flow of Events

Basic Flow

This use case starts when the Employee wishes to select a payment method.

1. The system requests that the Employee specify the payment method he would like (either: “pick up”, “mail”, or “direct deposit”).
2. The Employee selects the desired payment method.
3. If the Employee selects the “pick-up” payment method, no additional information is required.
   If the Employee selects the “mail” payment method, the system requests that the Employee specify the address that the paycheck will be mailed to.
   If the Employee selects the “direct deposit” method, the system requests that the Employee specify the bank name and account number.
4. Once the Employee provides the requested information, the system updates the Employee information to reflect the chosen payment method.

Alternative Flows

Employee Not Found

If, in the **Basic Flow**, information for the employee could not be located, the system displays an error message, and the use case ends.

Special Requirements

None.

Pre-Conditions

The Employee must be logged onto the system before this use case begins.

Post-Conditions

If the use case was successful, the payment method for the Employee is updated in the system. Otherwise, the system state is unchanged.

Extension Points

None.