Ryan Bishop
Aamir Ali
Ruchi Gupta
Akhyar Zaman
Pooja Aggarwal
Christian Angel

## Group #1 - Milestone 2 (Analysis Phase)

**Table of Contents:**

<u>**Executive Summary**</u>

Initially we define our nonfunctional and functional requirements which include a learning management, employee records management, employee skills management, and a customer relations management system. Our 'Systems Request' divulges the essential information, such as the business value and purpose of the project. Specific functionality can be derived from the use case diagram, along with the accompanying use case descriptions. An economic, technical, and organizational feasibility analysis shows us the project is fiscally and technically viable. We included activity diagrams detailing each use case, and CRC cards to define each relevant element to the SPMS. The learning management system aids administrators in creating and controlling course data,  employee records management system is meant to allow supervisors to manage the service engineers' course completion and skill level, the employee skills management system helps retain employee records, and the customer relations management system includes the interface for clients to schedule service orders. The class diagram helps link all these elements as well as define their relationships. We included behavioral state machines to lay out use case states and communication diagrams to relay the relations between the actors. The CRUDE matrices further define these relationships per use case; lastly the appendix provides a high-level overview of the project.

**1.) System Request - Service Personnel Management System (SPMS)**

| **Project Sponsor:** | - Thomas Tong, Expert Research Engineer of Agilent Technologies. |
|---|---|
| **Business Need:** | - SPMS has to be able to manage the training, scheduling and maintenance of service engineers. |
| **Business Requirements:** | - Be able to operate a database<br>- Be available to students with proper developmental tools |

**Business Value:** System is expected to greatly improve facilitation of training for the service engineers, as well as scheduling and management carried out by their respective field supervisors, as well as improve user interface for clients creating SOs. This will allow employees to handle more service orders, improve customer satisfaction and reduce costs by increasing overall efficiency of business processes within the company.
Estimation of tangible benefits to the company after four years:
- $5,000,000 increase in income from services
- $1,000,000 saved from customer complaints
- $2,000,000 saved from improved efficiency and integration of SE training program

**Special Issues or Constraints:**
- Students must have a basic knowledge of programming before learning how to use the

| | |
|---|---|
| system.<br>- Deciding who can be granted full admin access | |

**2.) Feasibility Analysis**

<u>Technical Feasibility: Can We Build It?</u>

This project will contain a low level of risk due to the advancement of technology in the recent years. Technology is advancing every year and the number of proficient technicians and developers also increase. The IT department will recruit the employees that best fit our team and are well rehearsed with the technology we will be utilizing. The size of the project group would be around 5-6 people, so the project size is considered medium risk. Current size of the team makes it easier to communicate and integrate thoughts and ideas. Being able to communicate ideas effectively will make this project go smoother and make the group work together cohesively. We will avoid compatibility issues due to the fact we have a few employees that are familiar with the current technology and will be able to work effectively with the system.

<u>Economic Feasibility: Should We Build it?</u>

First, we must identify the costs and benefits (both tangible and intangible) of building the system:

| Development Costs | Operational Costs |
|---|---|
| ● Servers (Apache/NGINX)<br>● Domain Names<br>● Database License<br>● Application Development Labor | ● Advertising<br>● Service Equipment (Tools)<br>● Office Supplies<br>● Rent<br>● Salaries - SE, FP<br>● Service Vehicles<br>● Travel Costs<br>● Utilities |
| **Tangible Benefits** | **Intangible Benefits** |
| ● Increased Income from Services | ● Knowledgeable/Skilled Employees |

| | |
|---|---|
| ● Reduction in Customer Complaints<br>● Reduction in Staff | ● Increased Customer Satisfaction<br>● Better System Reliability |

As we can see, there are two main types of costs ("Development" - from building the SPMS software applications, and "Operational" - costs that come from actually servicing repairs/running the brick-and-mortar business).

Now we can begin to assign values to these costs. It is important to conduct thorough research to obtain accurate estimates on the values:

| | 2016 | 2017 | 2018 | 2019 | Total |
|---|---|---|---|---|---|
| Increased Income from Services | 920,000 | 1,058,000 | 1,216,700 | 1,399,205 | |
| Reduction in Customer Complaints | 210,000 | 241,500 | 277,725 | 319,384 | |
| Reduction in Staff | 543,194 | 543,194 | 543,194 | 543,194 | |
| **TOTAL BENEFITS:** | 1,673,194 | 1,842,694 | 2,037,619 | 2,261,783 | |
| **PV OF BENEFITS:** | **1,624,460** | **1,736,916** | **1,864,710** | **2,009,565** | **7,235,651** |
| **PV OF ALL BENEFITS:** | **1,624,460** | **3,361,376** | **5,226,086** | **7,235,651** | |
| | | | | | |
| 4 Servers @ $1,630 | 6,520 | 6,520 | 6,520 | 6,520 | |
| 4 Website domain names @ $15 | 60 | 60 | 60 | 60 | |
| Database license | 5,000 | 5,000 | 5,000 | 5,000 | |
| Development labor | 453,096 | 0 | 0 | 0 | |
| **TOTAL DEVELOPMENT COSTS:** | 464,676 | 11,580 | 11,580 | 11,580 | |
| | | | | | |
| Advertising Expense | 4,500 | 2,500 | 2,000 | 2,000 | |
| Equipment Expense | 30,000 | 3,000 | 2,000 | 2,000 | |
| Office Supply Expense | 12,000 | 4,000 | 2,500 | 2,000 | |
| Rent Expense | 288,000 | 288,000 | 288,000 | 288,000 | |
| Salary Expense - SE | 376,820 | 376,820 | 376,820 | 376,820 | |
| Salary Expense - FP | 166,374 | 166,374 | 166,374 | 166,374 | |
| Salary Expense - Trainer | 200,000 | 200,000 | 200,000 | 200,000 | |
| Salary Expense - Schedulers | 110,000 | 110,000 | 110,000 | 110,000 | |
| Salary Expense - Misc. | 150,000 | 150,000 | 150,000 | 150,000 | |
| Vehicle Expense @ $30,000 | 150,000 | 0 | 0 | 0 | |
| Travel Expense | 10,400 | 10,400 | 10,400 | 10,400 | |
| Utility Expense | 12,000 | 12,000 | 12,000 | 12,000 | |
| **TOTAL OPERATIONAL COSTS:** | 1,510,094 | 1,323,094 | 1,320,094 | 1,319,594 | |
| | | | | | |
| **TOTAL COSTS:** | 1,974,770 | 1,334,674 | 1,331,674 | 1,331,174 | |
| **PV OF COSTS:** | **1,917,252** | **1,258,058** | **1,218,670** | **1,182,731** | **5,576,712** |
| **PV OF ALL COSTS:** | **1,917,252** | **3,175,311** | **4,393,981** | **5,576,712** | |
| | | | | | |
| **TOTAL PROJECT BENEFITS - COSTS:** | (301,576) | 508,020 | 705,945 | 930,609 | |
| **YEARLY NPV:** | (292,792) | 478,858 | 646,040 | 826,834 | **1,658,939** |
| **CUMULATIVE NPV:** | (292,792) | 186,065 | 832,105 | 1,658,939 | |
| | | | | | |
| **RETURN ON INVESTMENT:** | **29.75%** *(1,658,939 / 5,576,712)* | | | | |
| | | | | | |
| **BREAK-EVEN POINT:** | **1.61** *(break-even occurs in year 2; 478,858 - 186,065/478,858 = 0.61)* | | | | |
| | | | | | |
| **INTANGIBLE BENEFITS:** | Knowledgeable/Skilled Employees | | | | |
| | Increased Customer Satisfaction | | | | |
| | Better System Reliability | | | | |

**Break-Even Point**
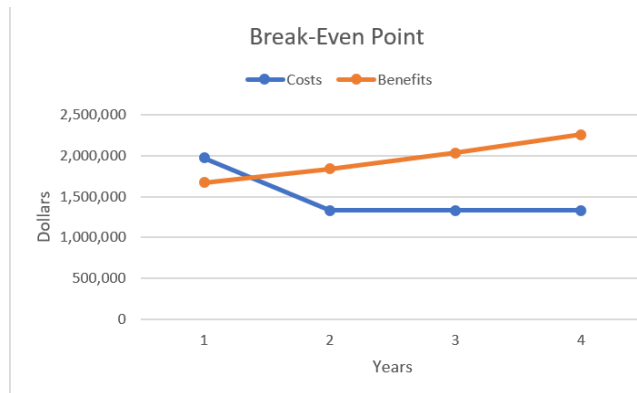
As we can see, this system is very cost effective/economically feasible. In year 2 we break-even and by year 4 we will make a profit of approximately $1.7M (that's nearly 30% ROI).

Cost-Benefit-Analysis Justifications & Sources:
- Increased Income from Services and Reduction in Customer Complaints are assumed to improve at a compound rate of 15% (employees will become more knowledgeable/skillful with each additional year of experience, thus increasing their operational efficiency and leading to less customer complaints).
- Since SEs are being trained as needed, we do not have to employ a specialist in each service field. We can reduce the number of Service Engineers by 50% (from 10 to 5). The average annual salary of an SE is $75,364 according to Glassdoor: https://www.glassdoor.com/Salaries/san-jose-field-service-engineer-salary-SRCH_IL.0,8_IM761_KO9,31.htm. Similarly, we can reduce the number of FPs by 50% (from 4 to 2) - we do not need have as many supervisors. Average annual salary of an FP is $83,187 according to Glassdoor: https://www.glassdoor.com/Salaries/san-jose-field-operations-supervisor-salary-SRCH_IL.0,8_IM761_KO9,36.htm
- Server costs were estimated from AWS (Amazon Web Services): https://aws.amazon.com/ec2/pricing/on-demand/
- Domain name costs were estimated from Namecheap: https://www.namecheap.com/
- Database costs were estimated from the MySQL website: https://www.mysql.com/tcosavings/
- Development cost was estimated by looking up salaries on Glassdoor:
  - https://www.glassdoor.com/Salaries/san-jose-senior-software-engineer-salary-SRCH_IL.0,8_IM761_KO9,33.htm

6

- - https://www.glassdoor.com/Salaries/san-jose-senior-database-administrator-salary-SRCH_IL.0,8_IM761_KO9,38.htm
    - https://www.glassdoor.com/Salaries/san-jose-full-stack-web-developer-salary-SRCH_IL.0,8_IM761_KO9,33.htm
    - https://www.glassdoor.com/Salaries/san-jose-junior-software-developer-salary-SRCH_IL.0,8_IM761_KO9,34.htm
    - *Note*: Dev. team consists of 4 total - 1 senior software engineer, 1 senior database administrator, 1 full-stack web developer and 1 junior software developer
  - Salary expenses were also found on Glassdoor:
    - *Note*: Assuming SE team consists of 5 employees, FP team consists of 2 supervisors, training team consists of 2 trainers and schedulers.
  - Rent expense was estimated for a 12,000 sq. ft. building in San Jose: http://www.loopnet.com/california/san-jose_office-space-for-lease/
  - Travel expense was calculated from current gas price per gallon times filling 5 service vehicles once per week for 52 weeks.
  - All PV calculations used a 3% interest rate.

Organizational Feasibility: If We Build It, Will They Come?

From an organizational standpoint, the project is low risk. The goals of the project is to make a system that makes it easier for the Field Supervisors to manage the Service engineers' training, scheduling, and maintenance. This project has a champion, Thomas Tong, Expert Research Engineer of Agilent Technologies. Tong will keep the project management updated and educated on the goals of the project. Since the request came because managing the service engineers was difficult. The new system will have allow field supervisors to see whether they need to train engineers more or hire more, so the user acceptance will be high seeing how it will improve the business' overall performance.

**3.) Work Plan**

Work Breakdown Structure (WBS)

| Task Number | Task Name | Duration (in weeks) | Dependency | Status |
|---|---|---|---|---|
| 1 | Milestone 1 | 2 | | Complete |
| 1.1 | System Request | | | Complete |

| | | | | |
|---|---|---|---|---|
| 1.1.1 | Establish Project Sponsor | | | Complete |
| 1.1.2 | Determine business need | | | Complete |
| 1.1.3 | Determine business requirements | | | Complete |
| 1.1.4 | Determine business value | | | Complete |
| 1.1.5 | Determine special issue or constrains | | | Complete |
| 1.2 | Feasibility analysis | | All sub parts in 1.1 | Complete |
| 1.2.1 | Determine Technical feasibility | | | Complete |
| 1.2.2 | Determine Economic feasibility | | | Complete |
| 1.2.3 | Determine Organizational feasibility | | | Complete |
| 1.3 | Workplan | | All sub parts in 1.1, 1.2 | Complete |
| 1.3.1 | Create work breakdown structure | | | Complete |
| 1.3.2 | Develop a Gantt chart | | | Complete |
| 1.3.3 | Develop a project effort estimation | | | Complete |
| 1.4 | Staffing plan | | 1.1 | Complete |
| 1.5 | Project monitoring | | All sub parts in 1.1, 1.2, 1.3 | Complete |
| 2 | Milestone 2 | 2 | 1 | Complete |
| 2.1 | Develop an analysis strategy | | | Complete |
| 2.1.1 | Functional Requirements | | 1 | Complete |

| 2.1.2 | Nonfunctional Requirements | | 1 | Complete |
|---|---|---|---|---|
| 2.1.3 | System proposal Outline | | | |
| 2.2 | Create Functional, Structural, and behavioral models | | All subtasks of 2.1 | Complete |
| 3 | Milestone 3 | 4 | 1,2 and all subtasks | In progress |
| 3.1 | Design class and methods | | 2.2 | In progress |
| 3.2 | Design data management layer | | | Open |
| 3.3 | Design human computer interaction layer | | | Open |
| 3.4 | Design Physical Architecture layer | | | Open |
| 4 | Final Milestone | | 1,2,3 and all subtasks | Open |
| 4.1 | Construct system | | | Open |
| 4.2 | Install system | | | Open |
| 4.3 | Put system into operation | | | Open |

Gantt Chart

| ID | Task name | Duration | Start | Finish |
|---|---|---|---|---|
| 1 | Milestone 1 | 2 | 1/8 | 1/23 |
| 2 | Milestone 2 | 2 | 1/30 | 2/13 |
| 3 | Milestone 3 | 4 | 2/14 | 3/6 |
| 4 | Milestone 4 | 1 | 3/7 | 3/13 |

## GANTT CHART
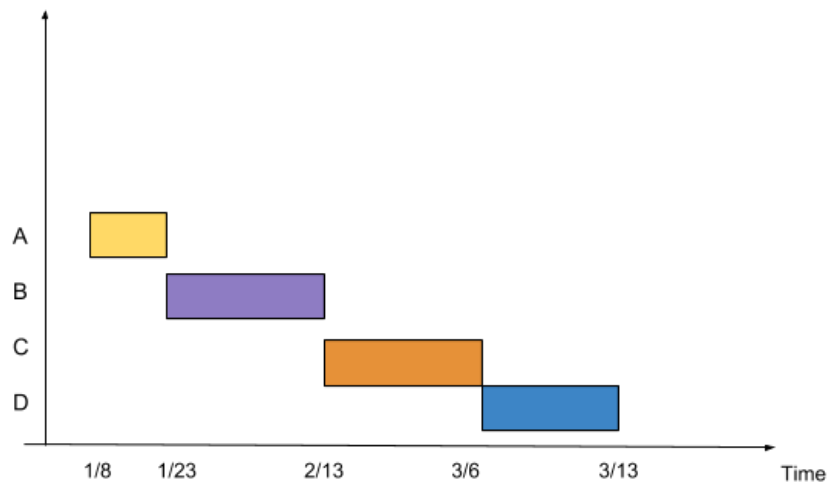


- Task A: Milestone 1
- Task B: Milestone 2
- Task C: Milestone 3
- Task D: Milestone 4

Project Effort Estimation
Unadjusted Actor Weighting table

| Actor type | Description | Weight Factor | Number | Result |
|---|---|---|---|---|
| Simple | External system with well-defined API | 1 | 1 | 1 |
| Average | Protocol based interface system | 2 | 2 | 5 |
| Complex | Human | 3 | 3 | 10 |
| Unadjusted actor weight total (UAW) | | | | 16 |

Unadjusted Use Case Weight table

| Use Case Type | Description | Weight Factor | Number | Result |
|---|---|---|---|---|
| Simple | 1-3 transactions | 5 | 1 | 5 |
| Average | 4-7 transactions | 10 | 1 | 10 |
| Complex | >7 transactions | 15 | 1 | 15 |
| Unadjusted use case wight total(UUCW) | | | | 30 |

Unadjusted use case points (UUCP)= UAW+UUCW 46 = 16 + 30

Technical Complexity Factors

| Factor Number | Description | Weight | Assigned Value (0-5) | Weight Value | Notes |
|---|---|---|---|---|---|
| T1 | Distributed system | 1.0 | 5 | 5.0 | |
| T2 | Response time | 1.0 | 5 | 5.0 | |
| T3 | End-user online efficiency | 1.0 | 5 | 5.0 | |
| T4 | Complex internal processing | 1.0 | 3 | 3.0 | |
| T5 | Reusability of code | 1.0 | 3 | 3.0 | |
| T6 | Easy to install | 0.5 | 3 | 1.5 | |
| T7 | Ease of use | 0.5 | 5 | 2.5 | |
| T8 | Portability | 0.5 | 2 | 1.0 | |
| T9 | Ease of change | 1.0 | 3 | 3.0 | |
| T10 | Concurrency | 1.0 | 3 | 3.0 | |

| | | | | | |
|---|---|---|---|---|---|
| T11 | Security | 1.0 | 5 | 5.0 | |
| T12 | Access for third parties | 1.0 | 5 | 5.0 | |
| T13 | User training | 1.0 | 4 | 4.0 | |
| Technical Factor Value (TFactor) | | | | 46.0 | |

Technical complexity factor (TFC) = 0.6 + (0.01 * TFactor)  1.06 = 0.6 + (0.01 * 46)

Environmental Factors

| Factor Number | Description | Weight | Assigned Value (0-5) | Weighted Value | Notes |
|---|---|---|---|---|---|
| E1 | Familiarity with system development process | 1.5 | 2 | 3.0 | |
| E2 | Application experience | 0.5 | 2 | 1.0 | |
| E3 | Object-oriented experience | 1.0 | 2 | 2.0 | |
| E4 | Lead analyst capability | 0.5 | 2 | 1.0 | |
| E5 | Motivation | 1.0 | 3 | 3.0 | |
| E6 | Requirements stability | 2.0 | 2 | 4.0 | |
| E7 | Part time staff | -1.0 | 0 | 0 | |
| E8 | Difficulty of programming language | -1.0 | 2 | -3.0 | |
| Environmental Factor Value (EFactor) | | | | 11.0 | |

Environmental factor (EF)  =1.4 + (–0.03 * EFactor) 1.07 = 1.4 + (–.03 * 11)

Adjusted use case points (UCP) = UUCP *TCF *ECF  52.17 = 46 * 1.06 * 1.07

Person hours multiplier (PHM) PHM = 20

Person hours = UPC * PHM 1,043.4 = 52.17 * 20

**4.) Staffing Plan**

| Role | Description | Assigned To |
|---|---|---|
| Project Manager | Oversees the project to ensure that it meets its objectives in time and maximizes cost/benefit ratio | |
| Infrastructure Analyst | Oversees that the project has no technical issues. Documents the project appropriately. | |
| System Analysts | Focuses on the project's issues with Information Systems | |
| Business Analyst | Focuses on the project's issues with the Business to design the Information System | |
| Data Analytics Specialist | Interpreting data, analyzing results using statistical techniques. | |
| Programmer | Codes the Information System | |
| | Reports to | Everyone |

-Reporting structure diagram goes here

**5.) Monitoring the Project**

| | Duration | Dependency |
|---|---|---|
| I.    Business Modeling<br>     A. Inception<br>        1. Understand current Business Situation<br>        2. Uncover Business process problem<br>        3. Identify potential projects<br>     B. Elaboration<br>     C. Construction<br>     D. Transition<br>     E. Production | | |
| II.    Requirements<br>     A. Inception<br>        1. Identify appropriate requirements analysis technique<br>        2. Identify appropriate requirements gathering techniques<br>        3. Identify functional and nonfunctional requirements<br>        4. Analyze current systems<br>        5. Create requirements definition<br>            a) Determine requirements to track<br>            b) Compile requirements as they are elicited<br>            c) Review requirements with sponsor<br>     B. Elaboration<br>     C. Construction<br>     D. Transition<br>     E. Production | | II.A.1, II.A.2<br><br>II.A.1, II.A.2<br>II.A.3, II.A.4<br><br>II.A.5.a<br><br>II.A.5.b |
| III.    Analysis<br>     A. Inception<br>        1. Identify business processes<br>        2. Identify use cases<br>     B. Elaboration<br>     C. Construction<br>     D. Transition<br>     E. Production | | <br><br>III.A.1 |

| | | |
|---|---|---|
| IV. Design<br>    A. Inception<br>        1. Identify potential classes<br>    B. Elaboration<br>    C. Construction<br>    D. Transition<br>    E. Production | | III.A |
| V. Implementation<br>    A. Inception<br>    B. Elaboration<br>    C. Construction<br>    D. Transition<br>    E. Production | | |
| VI. Test<br>    A. Inception<br>    B. Elaboration<br>    C. Construction<br>    D. Transition<br>    E. Production | | |
| VII. Deployment<br>    A. Inception<br>    B. Elaboration<br>    C. Construction<br>    D. Transition<br>    E. Production | | |
| VIII. Configuration and Change Management<br>    A. Inception<br>        1. Identify necessary access controls for developed artifacts<br>        2. Identify version control mechanisms for developed artifacts<br>    B. Elaboration<br>    C. Construction<br>    D. Transition<br>    E. Production | | |
| IX. Project Management | | |

| | | |
|---|---|---|
| A. Inception | | |
|     1. Create work-plan for the inception phase | | |
|     2. Create system request | | |
|     3. Perform feasibility analysis | IX.A.2 | |
|         a) Perform technical feasibility analysis | | |
|         b) Perform economic feasibility analysis | | |
|         c) Perform organizational feasibility analysis | | |
|     4. Identify project size | IX.A.3 | |
|     5. Identify staffing requirements | IX.A.4 | |
|     6. Compute cost estimate | IX.A.5 | |
|     7. Create work-plan for first iteration of the elaboration phase | IX.A.1 | |
|     8. Assess inception phase | I.A, II.A, III.A IV.A, V.A, VI.A VII.A, VIII.A, IX.A, X.A, XI.A XII.A | |
| B. Elaboration | | |
| C. Construction | | |
| D. Transition | | |
| E. Production | | |
| X.   Environment | | |
|   A. Inception | | |
|     1. Acquire and install CASE tool | | |
|     2. Acquire and install programming environment | | |
|     3. Acquire and install configuration and change management tools | | |
|     4. Acquire and install project management tools | | |
|   B. Elaboration | | |
|   C. Construction | | |
|   D. Transition | | |
|   E. Production | | |
| XI.   Operations and Support | | |
|   A. Inception | | |

| | | |
|---|---|---|
| B. Elaboration<br>C. Construction<br>D. Transition<br>E. Production | | |
| XII.    Infrastructure Management<br>    A. Inception<br>        1. Identify appropriate standards and enterprise models<br>        2. Identify reuse opportunities, such as patterns, frameworks, and libraries<br>        3. Identify similar past projects<br>    B. Elaboration<br>    C. Construction<br>    D. Transition<br>    E. Production | | |

## 6.) Requirements Definition

The first step in the Analysis Phase is to define what the system must do, or what characteristic the system must have. To determine the scope of the system, we will clearly list all performance/usability requirements (non-functional), as well as all process-oriented requirements (functional):

---

**Nonfunctional Requirements**

**1. Operational Requirements**
    1.1 The system should establish a connection with client though the request
    1.2 The system should create backlog every bug/error that is reported
**2. Performance Requirements**
    2.1 Build a log of every activity/retain records for auditing purposes
    2.2 Alert when no SE is found for SO
    2.3 Automatic and manual marking for completion of courses
**3. Security Requirements**
    3.1 Release updates that make the necessary changes to fix issue
    3.2 Allow admin access to anything in application
**4. Cultural and Political Requirements**
    4.1 No special cultural and political requirements are anticipated

---

**Functional Requirements**

**1. Learning System Managemen**
      1.1 Create courses
      1.2 Create course instances
      1.3 Register to courses
      1.4 Mark courses complete
**2. Employee Skills Management**
      2.1 Allow Field Supervisors to register direct reports for courses
      2.2 Allows supervisor to manage skill and skill level of direct reports
      2.3 Allow Product Specialist to create/edit equipment/course/training items association
      2.4 Allow Scheduler to view any SE's skills and courses
**3. Employee Records Management**
      3.1 Record entry to each employee when hired
      3.2 Record change in employee status
      3.3 Retain employee records for auditing
**4. Customer Relationship Management System**
      4.1 Creation and scheduling of Service Order's (SO)
      4.2 Searching for matching SE
      4.3 Alert when no SE is found for the SO
      4.4 Schedule the SE for the SO
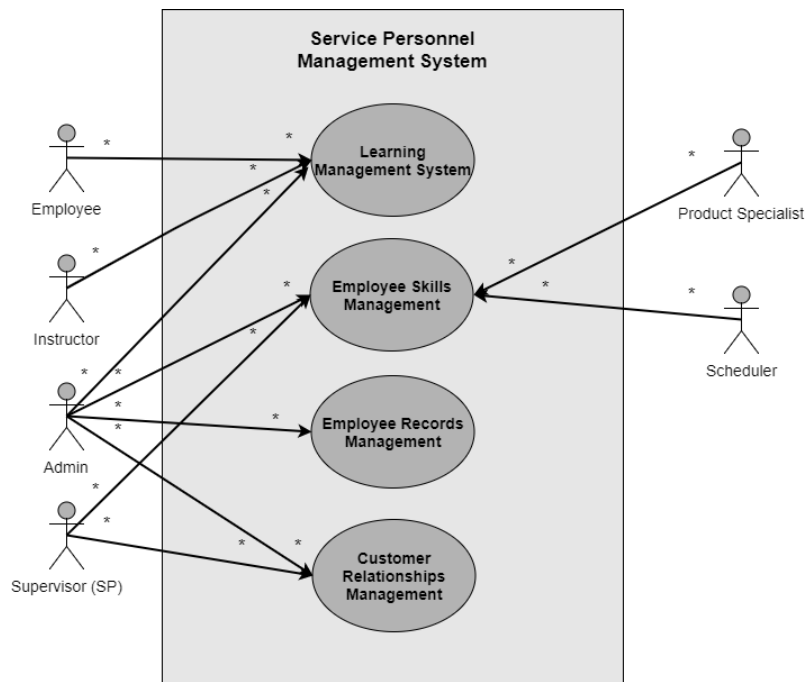      4.5 Record when the SO has been serviced

From here we can create a rudimentary outline of the Systems Proposal for the Service Personnel Management System:

**Outline of the Systems Proposal**

1. Table of Contents
2. Executive Summary
3. System Request
4. Economic Feasibility
5. Evolutionary Work Breakdown Structure
6. Requirements Definition
7. Functional Model (to be completed)
8. Structural Models (to be completed)
9. Behavioral Model (to be completed)

**7.) Functional Models**

Now we can begin the process of turning requirements into functional models. First, we will make a Use Case Diagram for the Service Personnel Management System:



Now we will create Use Case Descriptions for each use case:

| Use Case Name: Learning Management System | | ID: 1 | Importance Level: High |
|---|---|---|---|
| Primary Actor: Instructor | | Use Case Type: Overview, Essential | |
| Stakeholders and Interests:<br>   • Instructor can create courses<br>   • Instructor can create course instances<br>   • Employee can register to courses<br>   • Instructor (or system) marks courses complete for employee | | | |
| Brief Description: This use case describes how the learning system works for both instructors and employees. | | | |
| Trigger:<br>     Type: | | | |
| Relationships:<br>     Association: Instructor, Employee, Employee Records Management<br>     Include:<br>     Extend:<br>     Generalization: | | | |
| Normal Flow of Events:<br>   1. | | | |
| SubFlows:<br>   **S-1:** | | | |
| Alternate/Exceptional Flows: | | | |

| Use Case Name: Employee Skills Management | ID: 2 | Importance Level: High |
|---|---|---|
| Primary Actor: Supervisor (SP) | Use Case Type: Overview, Essential | |

**Stakeholders and Interests:**
- Allow Field Supervisors to register direct reports for courses
- Allows supervisor to manage skill and skill level of direct reports
- Allow Product Specialist to create/edit equipment/course/training items association
- Allow Scheduler to view any SE's skills and courses

Brief Description: This use case describes how supervisors can manage employees' skills.

Trigger: Employee completes new skill.
       Type: External

Relationships:
       Association: Supervisor (SP), Admin, Schedulers, Product Specialist, Learning System Management,
                    Employee Records Management, CRM
       Include:
       Extend:
       Generalization:

Normal Flow of Events:
    1.

SubFlows:
   **S-1:**

Alternate/Exceptional Flows:

:

| Use Case Name: Employee Records Management | ID: 3 | Importance Level: High |
|---|---|---|
| Primary Actor: Admin | Use Case Type: Overview, Essential | |

Stakeholders and Interests:
- Record entry to each employee when hired
- Record change in employee status
- Retain employee records for auditing

Brief Description: This use case describes how employee's records are automatically updated when changes to other use cases occur.

Trigger: Employee records change in a table schema.
    Type: Internal

Relationships:
    Association: Admin, Learning System Management, Employee Skills Management, CRM
    Include:
    Extend:
    Generalization:

Normal Flow of Events:
    1.

SubFlows:
    **S-1:**

Alternate/Exceptional Flows:

| Use Case Name: Customer Relationship Management | | ID: 4 | Importance Level: High |
|---|---|---|---|
| Primary Actor: Supervisor (SP) | | Use Case Type: Overview, Essential | |

Stakeholders and Interests:
- Creation and scheduling of Service Order's (SO)
- Searching for matching SE
- Alert when no SE is found for the SO
- Schedule the SE for the SO
- Record when the SO has been serviced

Brief Description: This use case describes how customers issue orders and the SPMS fulfills the order with an SE.

Trigger: Customer issues Service Order (SO).
        Type: External

Relationships:
        Association: Supervisor (SP), Admin, Employee Skills Management, Employee Records
                    Management
        Include:
        Extend:
        Generalization:

Normal Flow of Events:
    1.

SubFlows:
    **S-1:**

Alternate/Exceptional Flows:

Now we will create the activity diagram for the use cases:
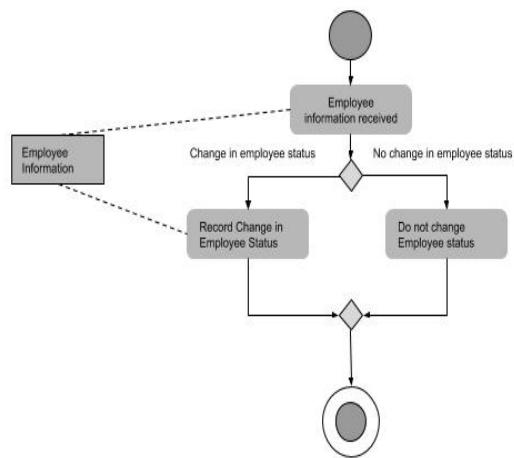
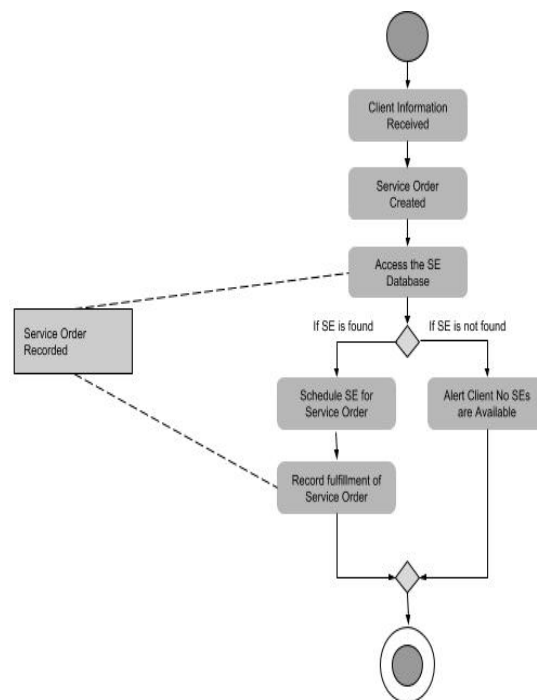Activity Diagram for Learning System Management.)
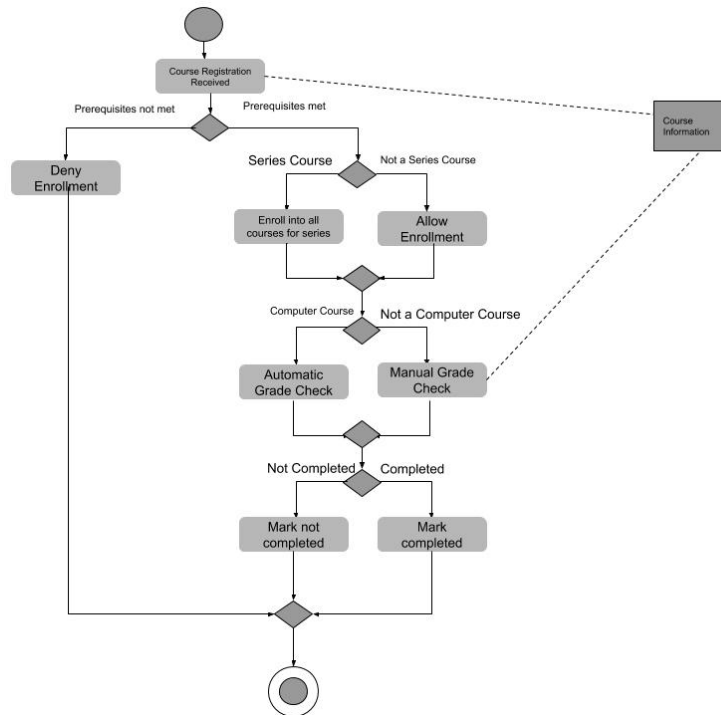
Activity Diagram for Employee skills Management.)
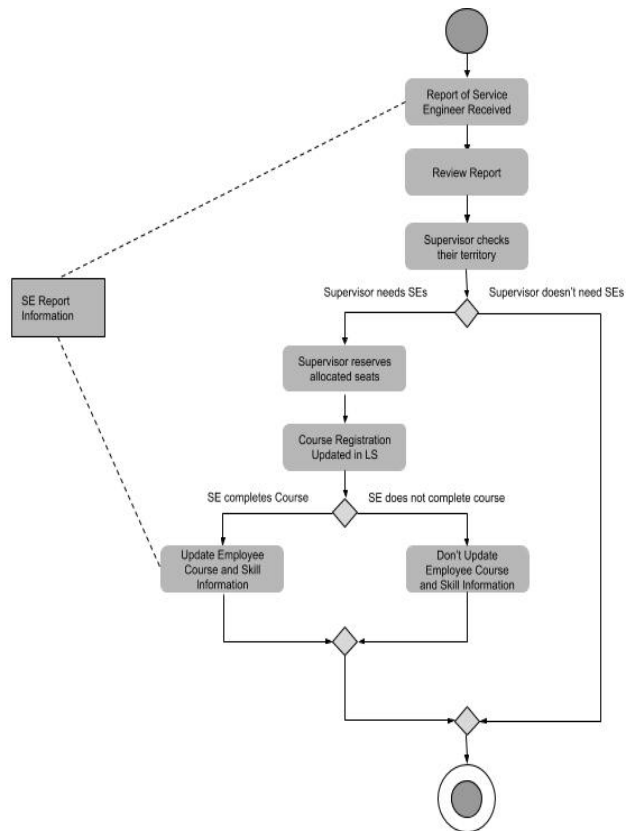
Activity Diagram for Employee Records Management.)

Activity Diagram for Customer Relationships Management.)

**8.) Structural Models**

a.) CRC Cards

Represents a kind of person, place, or thing about which the system will need to capture and store information.

Learning Management System

Front:

Class Name: Instructor     ID : 1     Type: concrete , domain
Description: Teaches the learning system to employees who are unfamiliar.

Responsibilities:                    Collaborators:
- Instructor can create courses.      Employee
- Instructor can create course
instances.
- Instructor marks courses
complete for employees.

Back

Attributes:
EmployeeID(double):
FirstName(text):
LastName(text):
Role(text):
Department(text):

Relationships:
Generalization:
Aggregation: Instructor
Other Associations:

Employee Skills Management

Front

Class Name:Supervisor          Type:concrete,domain
ID: 1

Description: Manage Employees' smills.

Responsibilities:                 Collaborators:
- Register direct reports for     Admin
courses.                          Scheduler
- Manage skill and skill level.   Product Specialist

Back

Attributes:
EmployeeID(double);
Clearance(double):
FirstName(text):
LastName(text):

Relationships:
Generalizations:
Aggregation: Admin
              Scheduler
              Product Specialist
Other Association:

Front:

Class Name: Admin          Type: concrete, domain
ID: 2

Description: Manage employees' information and skills.

Responsibilities:          Collaborators:
- Enter Employee's         Supervisor
Information.               Scheduler
- Change Employee's        Product Specialist
Information.

Back

Attributes:
EmployeeID(double):
Clearance(double):

Relationships:
Generalizations:
Aggregation: Supervisor
              Scheduler
              Product Specialist
Other Associations:

Front

Class Name:Product Specialist                    ID: 3
Type: concrete, domain
Description: Creates or edits training resources.


 Responsibilities:              Collaborators:
 - Create equipment.            Supervisor
 - Create training items.       Admin
 - Help edit the course.        Scheduler



Back

Attributes:
EmployeeID(double):
EmployeeSkill_ID(double):
FirstName(text):
LastName(text):

Relationships:
Generalizations:
Aggregation:Supervisor
             Admin
             Scheduler
Other Associations:

Front

Class Name: Scheduler                ID: 4
Type: concrete, domain
Description: View any SE's skills and
courses.

Responsibilities:              Collaborators:
- Schedule courses for      Supervisor
employees.                      Admin
- Maintain SE's                Product Specialist
courses.
- View SE's skills.

Back

Attributes:
EmployeeID(double):
FirstName(text):
LastName(text):

Relationships:
Generalizations:
Aggregation: Supervisor
                    Admin
                    Product Specialist
Other Associations:

Employee Records Management

## Admin

**Front**

**Class Name:** Admin **ID:** 1   **Type:** Concrete, domain
**Description**: manage employee records

**Responsibilities**                **Collaborators**
Enter employee info when hired       Employee
Change employee info

**Back**
**Attributes:**
EmployeeID(double):
Clearance(double):

**Relationships**
**Generalization:**
**Aggregation:**          Employee

**Other Associations:**

Employee

Front

**Class Name:** Employee **ID:** 2   **Type:** Concrete, domain
**Description**: holds records for employees in company

**Responsibilities**              **Collaborators**
Maintain info                     Admin

Back
**Attributes:**
EmployeeID(double):
Role(text):
FirstName(text):
LastName(text):          Department(text):

**Relationships**
**Generalization:**
**Aggregation:**          Admin

**Other Associations:**

Customer Relationship Management

## Client

**Front**

**Class Name:** Client   **ID:** 1   **Type:** Concrete, domain
**Description**: commissions engineers from Agilent

**Responsibilities**                **Collaborators**
Create SO's                         Service Engineer
Mark SO complete                    Supervisor
                                    Service Order

**Back**
**Attributes:**
CompanyID(double):          AddressStreet(text):
CompanyName(text):          AddressCity(text):
ContactFirstName(text):     AddressState(text):
ContactLastName(text):      PhoneNumber(text):

**Relationships**
**Generalization:**
**Aggregation:**           Service Engineer
                           Supervisor
**Other Associations:** Service Order

## Service Order

**Front**

**Class Name:** Service Order  **ID:** 2    **Type:** Concrete, domain

**Description**: details task required by client

| **Responsibilities** | **Collaborators** |
|---|---|
| Provide task/skills required of SE | Supervisor |
| Provide client's info | Service Engineer(SE) |
| | Client |

**Back**

**Attributes:**

ServiceOrderid(double):          Status(text):

CustomerID(double):

EmployeeSkillid(double):

Productid(double):

**Relationships**

**Generalization:**

**Aggregation:**

**Other Associations:** Supervisor

                    Service Engineer        Client

Supervisor

Front

**Class Name:** Client   **ID:** 3   **Type:** Concrete, domain
**Description**: Manages placing of direct reports (SEs)

**Responsibilities**                    **Collaborators**
Track alerts through system            Client
Notify appropriate SE                  Service Engineer
                                       Service Order

Back
**Attributes:**
EmployeeID(double):          FirstName(text):
Clearance(double):           LastName(text):

**Relationships**
**Generalization:**
**Aggregation:**          Service Engineer
                          Client
**Other Associations:** Service Order

Service Engineer

**Front**

**Class Name:** Service Engineer  **ID:** 4    **Type:** Concrete, domain
**Description**: fulfills service requested by SO's

**Responsibilities**                          **Collaborators**
Complete service required            Supervisor
        by SO when SP notifies       Service Order

**Back**
**Attributes:**
EmployeeID(double):
EmployeeSkillid(double):
ContactFirstName(text):
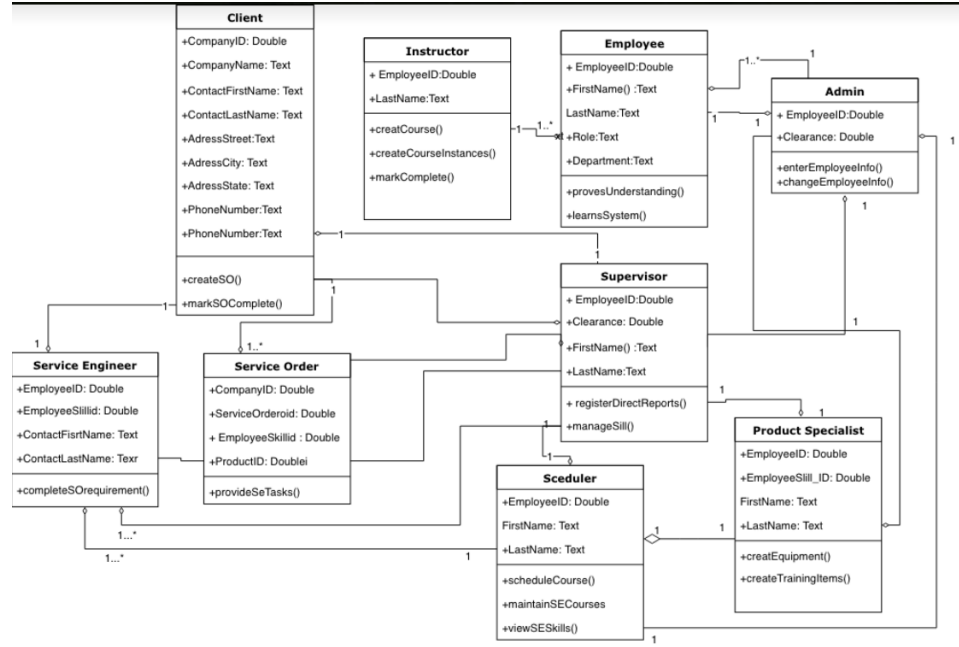ContactLastName(text):          PhoneNumber(text):
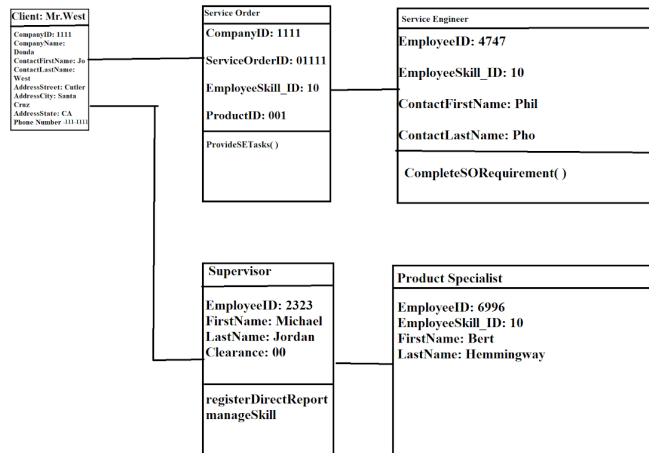
**Relationships**
**Generalization:**
**Aggregation:**          Client
                          Supervisor
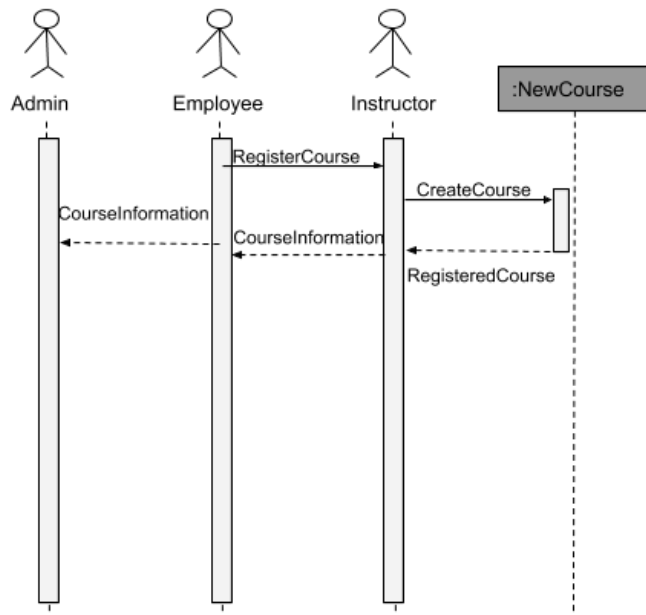**Other Associations:** Service Order
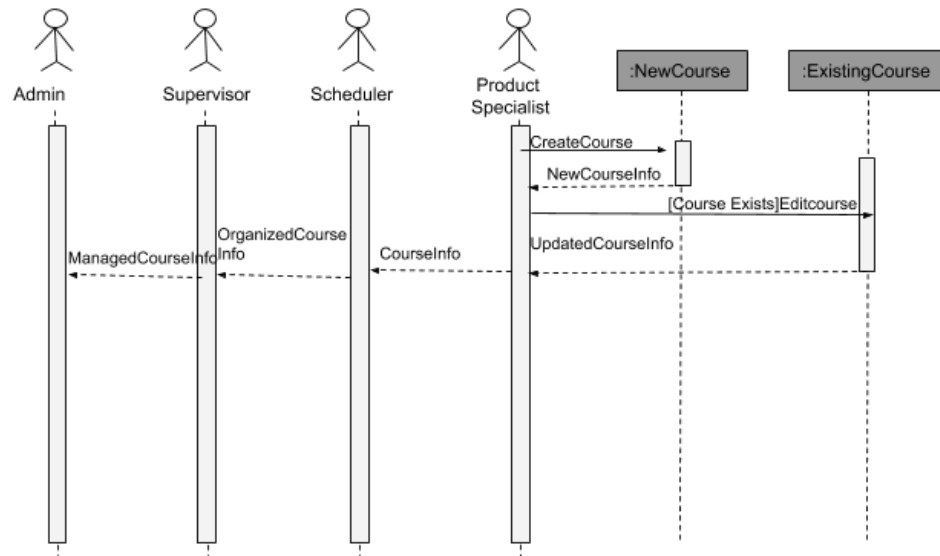
b.) Class Diagram



c.) Object Diagrams
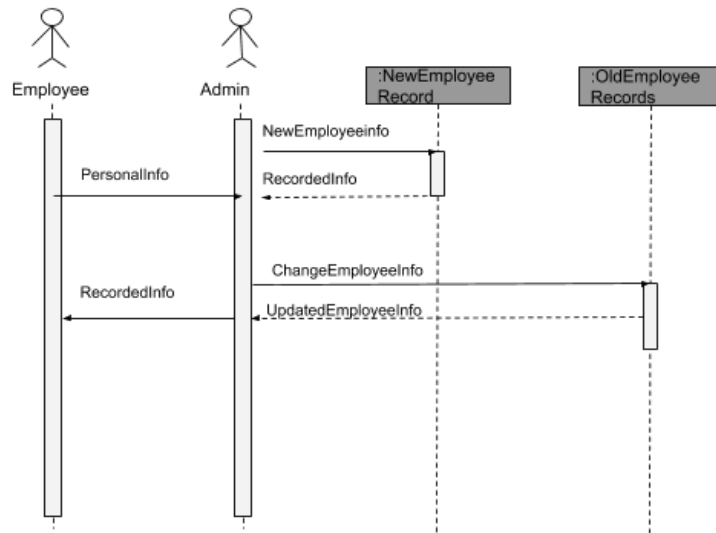
**9.) Behavioral Models**

a.) Sequence Diagrams

Learning Management System Use Case

Admin
Employee
Instructor
:NewCourse

RegisterCourse

CreateCourse

CourseInformation

CourseInformation

RegisteredCourse

Employee Skills Management System Use Case

Admin    Supervisor    Scheduler    Product Specialist    :NewCourse    :ExistingCourse

CreateCourse

NewCourseInfo

[Course Exists]Editcourse

OrganizedCourse Info

CourseInfo

ManagedCourseInfo

UpdatedCourseInfo

Employee Records Management System Use Case

Employee     Admin     :NewEmployee Record     :OldEmployee Records

NewEmployeeinfo

PersonalInfo     RecordedInfo

ChangeEmployeeInfo

RecordedInfo     UpdatedEmployeeInfo

Client | Service order | Supervisor | Service Engineer | :SO service

PerformService

ComissionsServiceOrder

DetailedTask

TaskRequest

CompletedService

TracksFinishedService

OrganizedCompletedService

CompletedService

b.) Communication Diagrams

Learning Management
System Use Case

5:CourseInformation

1:RegisterCourse

2:CreateCourse

3:RegisteredCourse

Admin          Employee          Instructor          :NewCourse

4:CourseInformation

Employee Skills
Management
System Use Case

1:CreateCourse

2:NewCourseInfo

:NewCourse

7:ManagedCourseInfo     6:OrganizedCourseInfo     5:CourseInfo

Admin          Supervisor          Scheduler          Product
Specialist

3:EditCourse          :ExistingCourse

4:UpdatedCourseInfo

45

1:PersonalInfo

2:NewEmployeeinfo

6:RecordedInfo

3:RecordedInfo

:NewEmployee
Record

Employee

Admin

4:ChangeEmployeeInfo

:OldEmployee
Records

5:UpdatedEmployeeInfo

1:ComissionsServiceOrder    2:DetailedTask    3:TaskRequest    4:PerformService

Client    Service    Supervisor    Service    :SO service
order    Engineer

8:CompletedService    7:OrganizedCompletedService    6:TracksFinishedService    5:CompletedService

c.) Behavioral-State Machines

1. CRUDE Matrix- Learning management system

|  | Instructor | Courses | Employee |
|---|---|---|---|
| Instructor |  |  |  |
| Courses | C, R, U, D |  | R, E |
| Employee | C, U, D |  |  |

2. CRUDE Matrix- Employee skills management

|  | Supervisor | Skills | Admin | Product Specialist |
|---|---|---|---|---|

| | | | C, R, U | |
|---|---|---|---|---|
| Supervisor | | | C, R, U | |
| Skills | C, R, U | | C, R, U, D | C, R, U, D |
| Admin | | | | |
| Product Specialist | | | C, R, U, D | |

3.  CRUDE Matrix- Employee Records Management

| | Admin | Courses | Employee (SE) |
|---|---|---|---|
| Admin | | | R |
| Courses | C, R, U, D | | R, E |
| Employee (SE) | C, R, U | | |

4.  CRUDE Matrix- Customer's Relationship Mgmt System

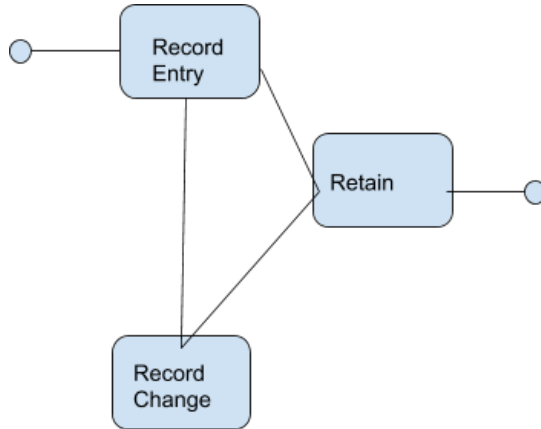| | Client | Service Order | Supervisor(SE) | Service Engineer |
|---|---|---|---|---|
| Client | | | C, R, U, E | R |
| Service Order | C, R, U, D | | | E |
| Supervisor | | | | |
| Service Engineer | D, U | | | |

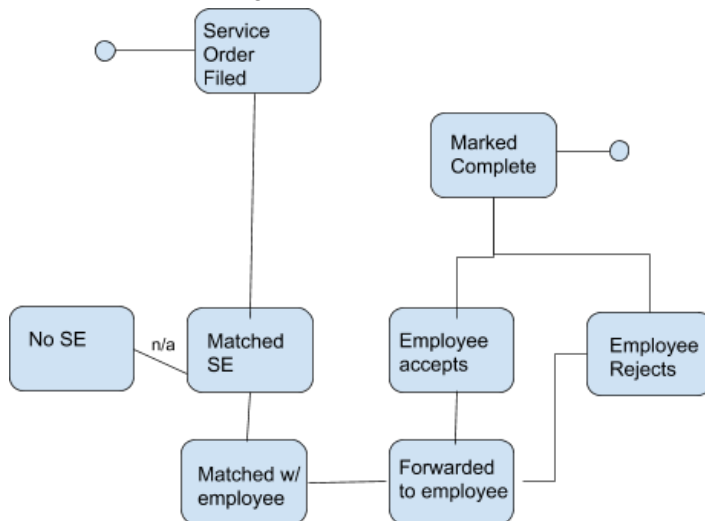d) Behavioral Models
  1. Learning Mgmt System
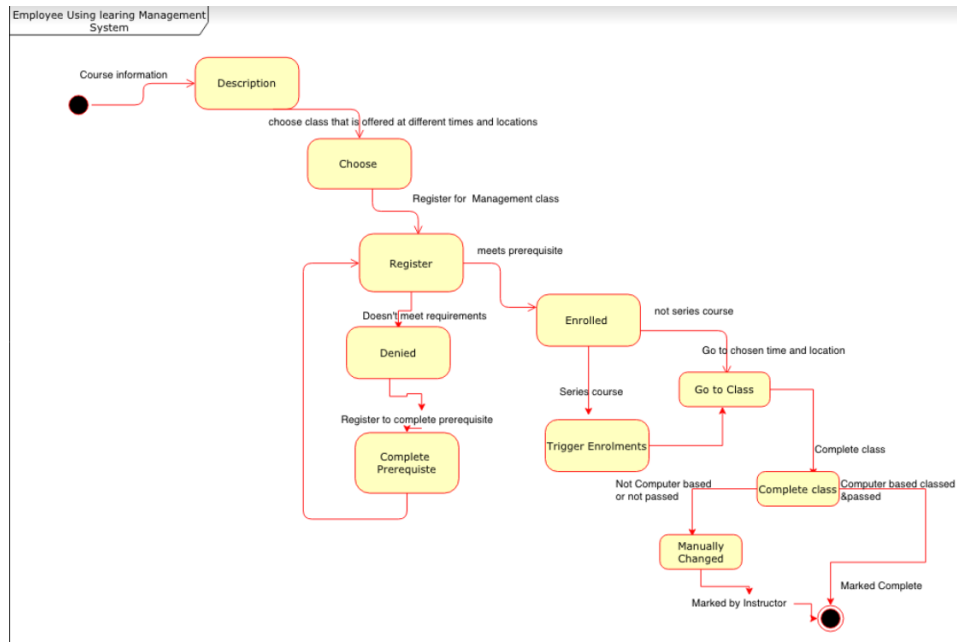


  2. Employee Skills Mgmt
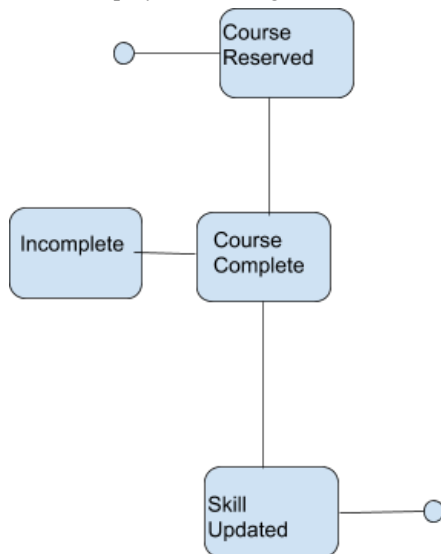
3. Employee Records Mgmt



4. Customer Relations Mgmt

## 10.) Appendices

The Service Personnel Management System could be very beneficial system to Agilent Technologies because other companies use this system and benefit greatly off of it. A company such as Smart and Final. They have a system that allows them to train employees when requested and always updates their statuses on the Dayforce Application. The system they have also allows them to know when someone is training so they won't be scheduled for work. They also monitor our performance as Cashiers with cameras to make sure if we need to be trained again they can provide the training although it rarely happens. Smart and Final also has a Website called SmartU that allows their employees to take course to specialize in other fields such as produce and meat. This system greatly benefits Smart and Final, and so we believe it would benefit Agilent Technologies greatly as well.