***SOFTWARE ANALYSIS SPECIFICATION***

# **Introduction**

## **Goals and objectives**

### The goal:

The goal of Osric’s project is to determine the effectiveness of Osric’s scheme. We must take Osric’s plan where the system prioritizes customers and develop an efficient software that may save the company time and money. By doing so, we must simulate the effectiveness of the scheme using data and statistics on a job mix. The statistics will identify the average waiting time before a job has started, the average queue length, the percentage of time the queue is empty on day and night, the number of blocks when a technician is idle, and the number of jobs that cannot be continued at night because no technician is available. Afterwards, we must compare Osric’s scheme to the pure first-come-first-serve bases using the same job list. Once the simulation is complete, we must look at the statistics from both schemes and determine which is more time and cost effective for Osric.

**1.1.2 The objectives:**

* + Software Project Plan.
  + Requirements Phase
  + Design.
  + Development (Processing)
  + Implementation and Source Code
  + Testing/Integration
  + Statistics Gathered
  + Conclusion
  + Plan for Change
  + Time & Budget Management

## **Statement of Scope**

There is a need to ration the services offered by prioritizing the customers when they’re requesting a service. Once a service is requested, it will be sent into a priority queue where its priority will be established, and from there it will be sent into a waiting list where it will await until the job is selected to be completed.

* **Major inputs:**
  + Customer information
  + Approval of Job
  + Job to be completed
  + Technician Job list
* **Processing functionality:** 
  + New customer requests services
  + Assistant receives customer information
  + Assistant takes customer’s request and seeks Manager’s approval
  + Manager looks at job list
  + Manager Accepts or declines request
  + If list is too long, decline request
  + If request is approved, the manager will notify the assistant
  + Job is then created
  + Assistant takes the job and assigns it to a priority system
  + There are four priority systems that are each first come first serve
  + Waiting list holds the jobs in a determinate order
  + Technician is assigned a job and completes job
  + Time-block is the amount of time it takes to fully complete a job
  + Customers are notified when job is ready to begin and when it will be complete
  + Billing info are generated
  + Customer receives survey
  + Assistant looks at survey and stores the information
* **Major outputs:**
  + The system will be able to give out estimates for Osric and for traditional scheme.
  + Single job & Overall Jobs-
    - Job duration
    - Average waiting time before a job is started
    - Standard Deviation and mean
    - Shortest Time
    - Longest Time
    - Cost
  + When customer calls-
    - How soon a service will be available
    - Assist obtains Name, Billing address, Phone #, Priority
    - When a customer is added, they are given a customer number
    - Length of the current waiting list
    - Average time to complete job
    - Worst Case (length to reach priority 4)
  + Priority Queue-
    - Average queue length
    - Percentage that the queue is empty- day & night
    - First-come-first-serve for the regular priority without Osric’s Scheme
  + Technician-
    - Number of blocks when a technician is idle
    - Number of jobs that cannot be continued because no technician is Available

## **Software context:**

* Manager approves new jobs
* Assistant contacts customer
* Assistant confirms job
* Assistant assigns priority
* Assistant updates waiting list
* Assistant adds and updates technicians shifts 1-3 (8 hours each)
* Assistant reports to technician
* Assistant mails bill
* Customer calls assistant
* Customer request a job
* Customer is assigned a technician
* Customer has a bill
* Technician reports to assistant
* Technician is assigned to a customer
* Technician is assigned a job
* Technician takes time-block to complete job
* Software gathers data
* Software calculates estimates
* Software generates results

## **Major constraints:**

* Lack of experience
* Missed requirements
* Poor design
* Unpredicted risks
* All knowing different languages

**2.0 Usage scenario**

**2.1 User profiles**

* + Manager – Approve or decline new job request
  + Customer – Requests services, provide customer information to the assistant, use survey to rate technician
  + Assistant- Collect information from manager, customer, and technician. The collected information will be placed into the software and manage other software operation, such as creating a report and a payment invoice for the customer etc.
  + Technician- Use the software to update the job status.

**2.2 Major software functionality**

* + Requests can either be accepted or declined for new customer
  + Store new customer information
  + Enter existing user information
  + Assigns customer priority
  + Add job to the waiting list and priority queue
  + Assigns technician to the customer
  + Assign job to the technician
  + Update/Sort waiting list
  + Assign job with a time block
  + Generate invoice payment
  + Send notification to both customer and technician
  + Retrieve report

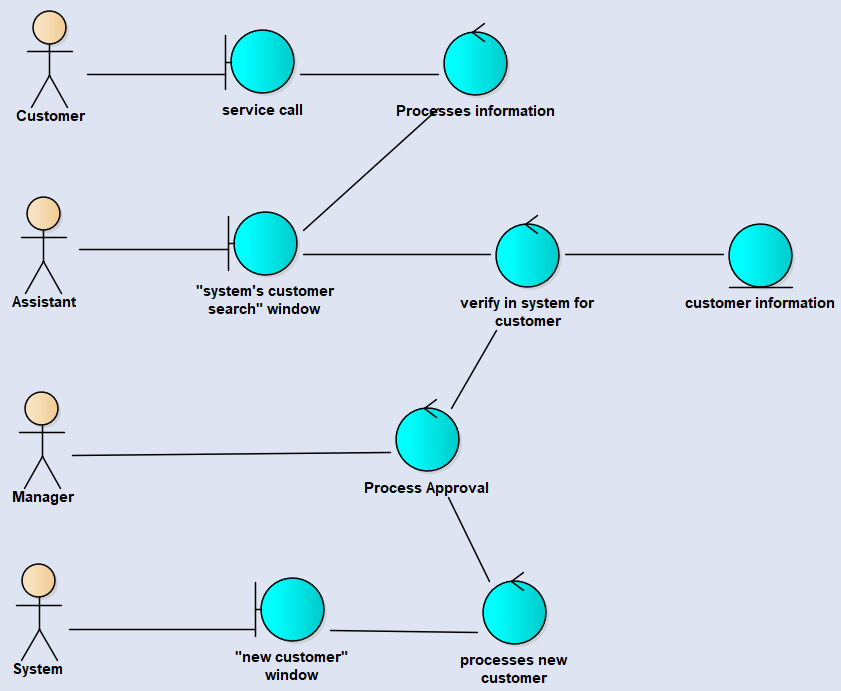
**2.3 Special usage considerations**

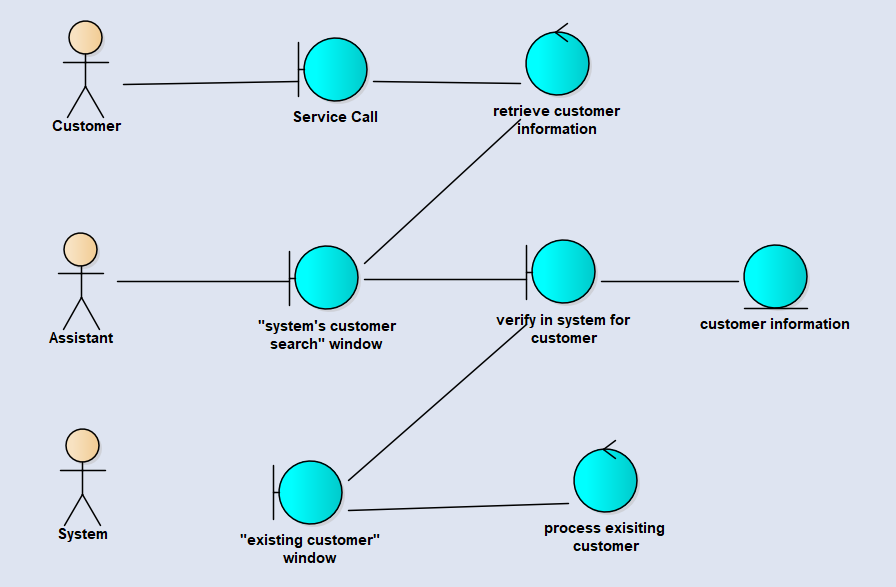
* + User/Customer must have some type device that have access to the internet or callable mobile
  + The customer must able to speak and understand English
  + Notify customer
  + Customer with priority 0, VIP

**3.0 Data Model and Description**

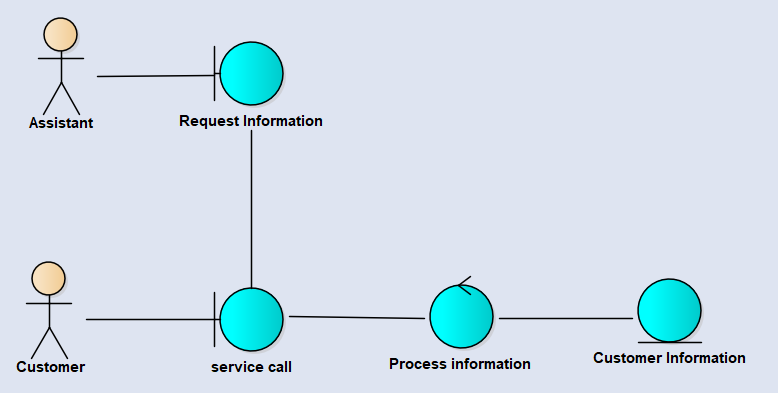
**3.1 Data Description**

**Use Case #1: New Customer**

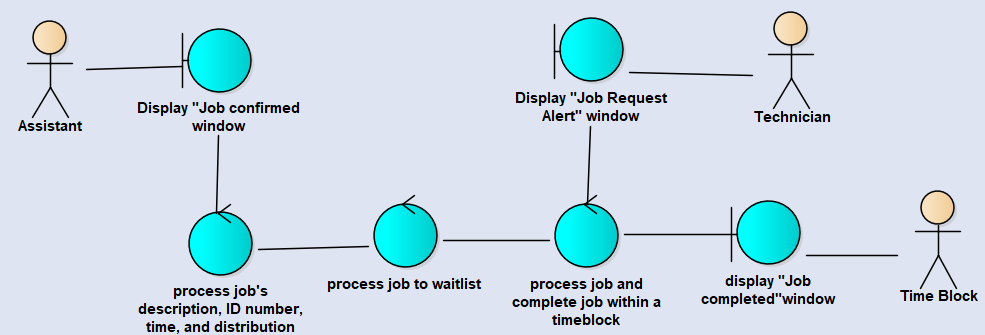




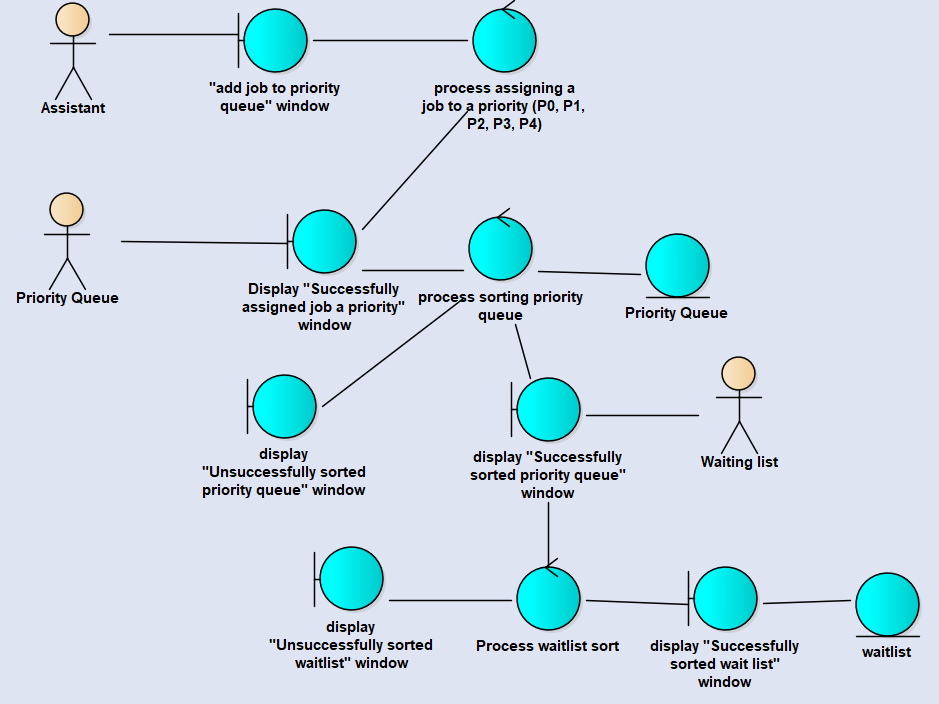
**Use Case #2: Customer Information**



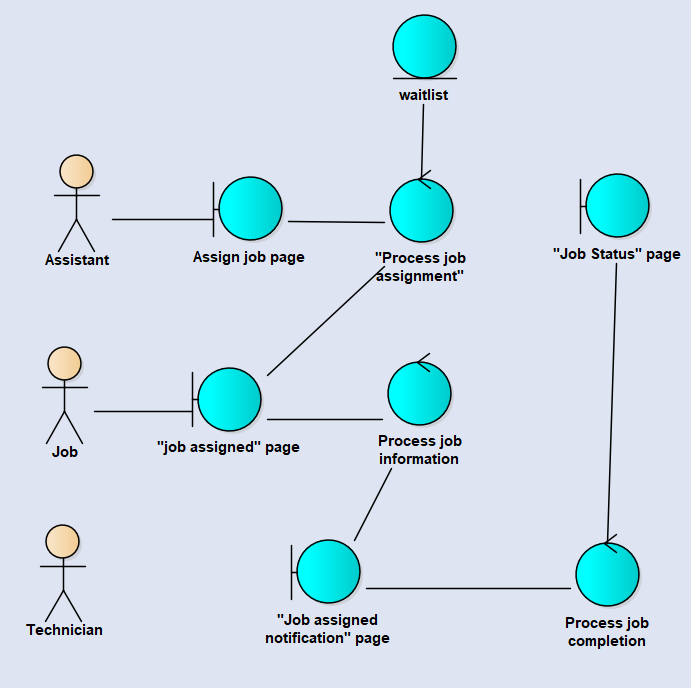
**Use Case 3: Service Request**



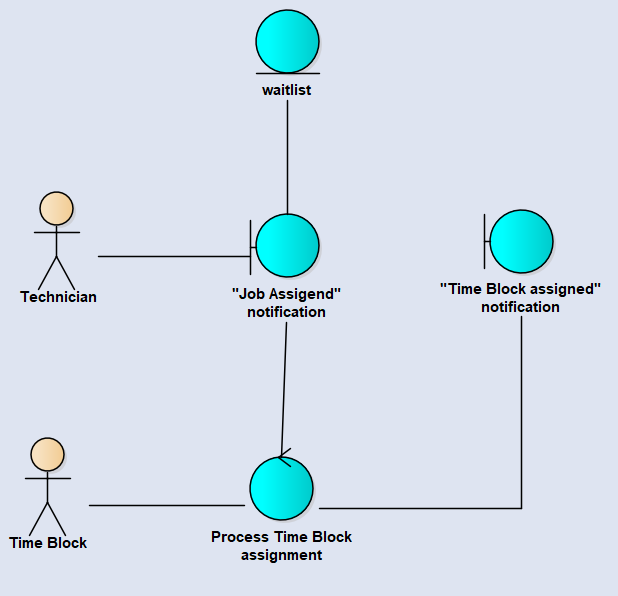
**Use Case 4: List/Priority**



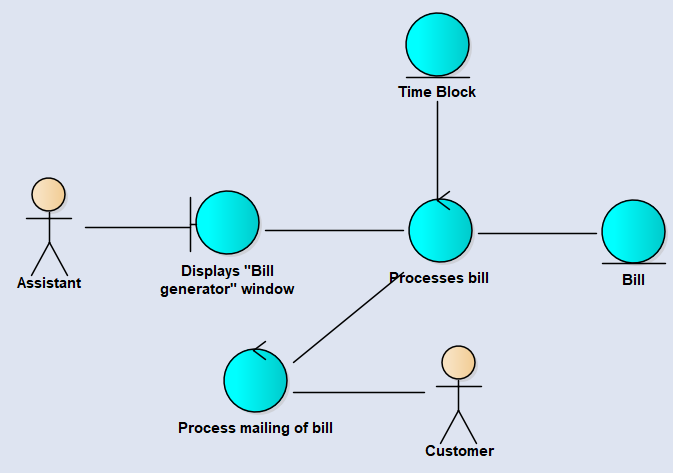
**Use Case #5: Technician and System**



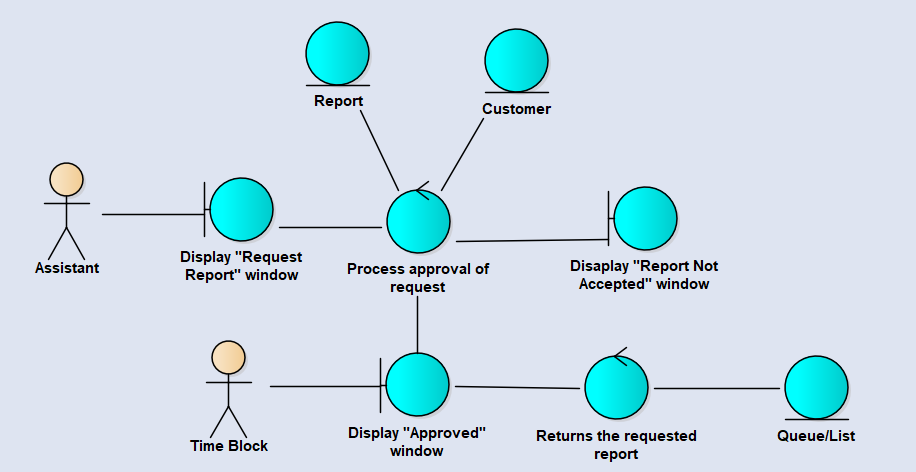
**Use Case 6: Time Block & Job**



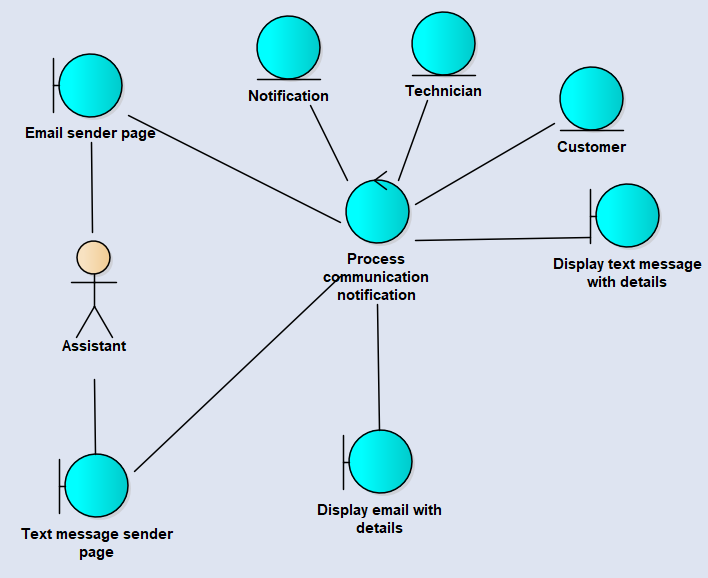
**Use Case 7: Generate Customer Bill**



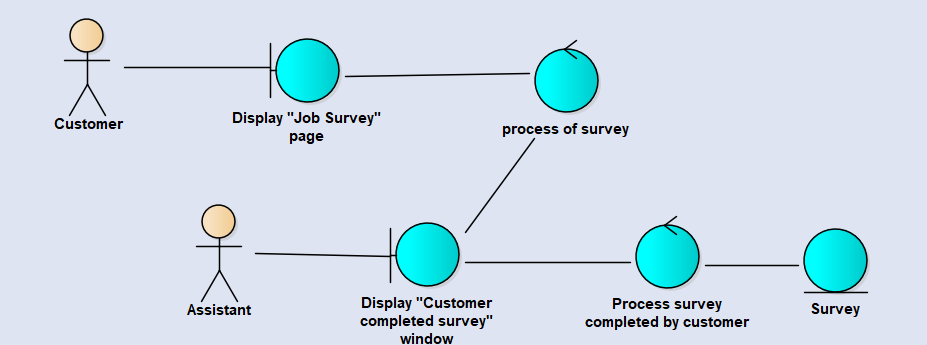
**Use Case #8: Assistant Obtains Reports**



**Use Case #9: Notification**



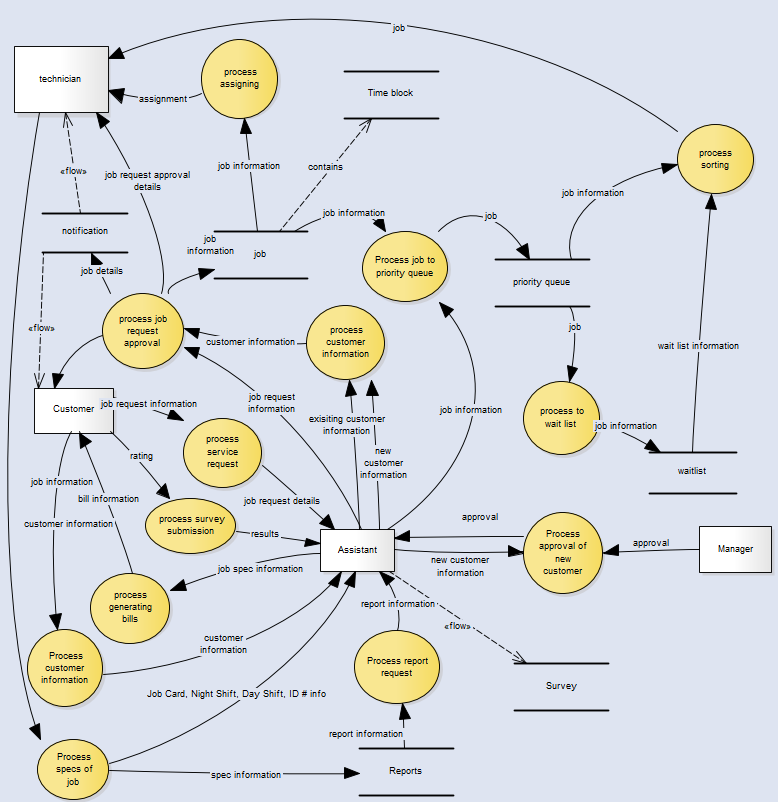
**Use Cases #10: Survey**



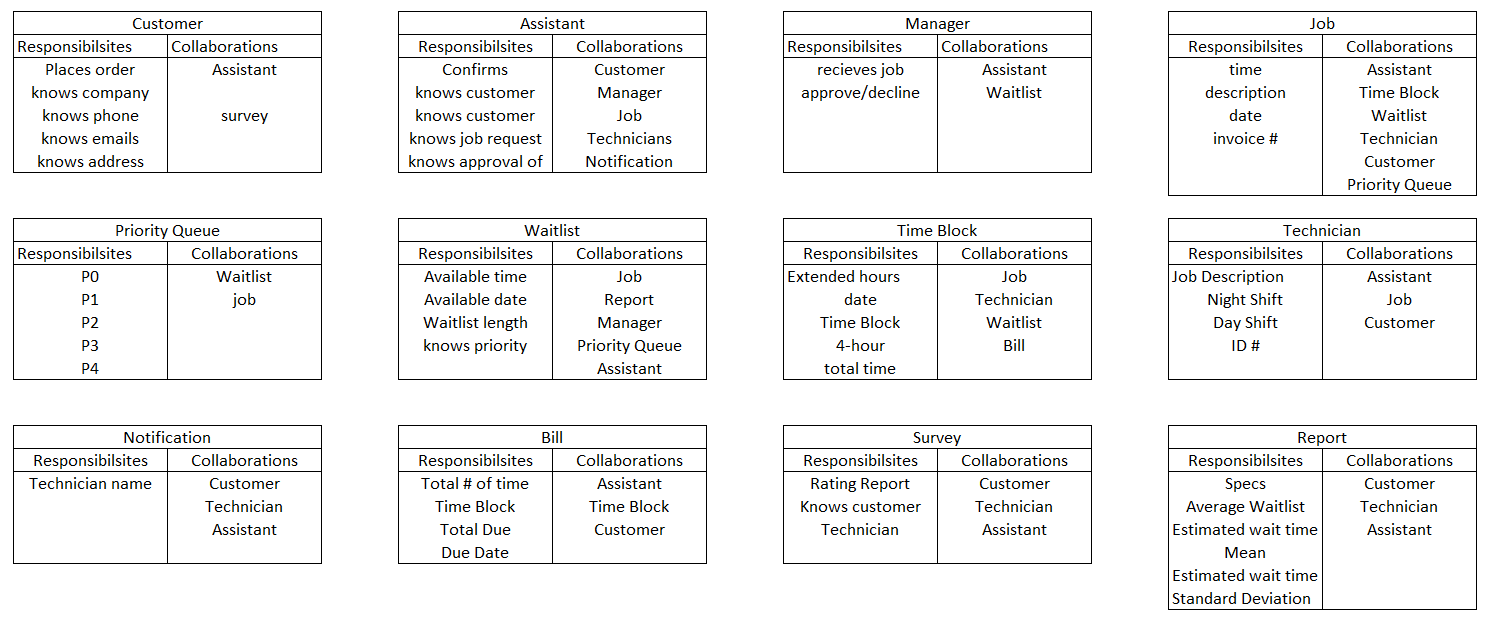
**3.1.1 Entity-Relationship Diagram**

[**Complete ERD**](https://onedrive.live.com/edit.aspx?cid=9f4a43508d51317f&page=view&resid=9F4A43508D51317F!1573&parId=9F4A43508D51317F!1565&app=Word)

**3.1.2 Data Flow Diagram**



**3.1.3 Object Relationships**

**Relationships among data objects are described using CRC cards. No attempt is made to provide detail at this stage.** 

**3.1.4 Complete data model**

**An UML Class model (class diagram) for the software is developed – through attributes and actions (not data typing, method signatures, access)**

In the folder submitted under 3.1.4\_UML Class diagram

**3.1.5 Data dictionary**

In the folder submitted under 3.1.5\_Data Dictionary

**4.0 Functional Model and Description**

**Description of major software functions along with UML Use Case, sequence, and communication diagrams.**

**4.1 Use cases**

**A detailed description of each software function is presented by completing the use case template.**

**Cross reference this document with file name of use case summary document**

**LIST all of the use cases cross-listed with the file names of actual document**

* + New customer and Services Request
    - Upon manager approval for new customer, the assistant will use to function to create and store the information of a new customer, then the system will assign the customer with a priority number for future services. Now the customer are assigned with a priority now and added to the waitlist for the next available technician.
  + Customer Information
    - Store the customer information into the database file
  + List and Priority
    - Add customer to the waiting list for the next available technician and assign the customer with a priority number
  + Technician and System
  + Time Block and Jobs
    - While the technician is on duty, the technician will be assigned with a time block of 4 hours, it will be used for tracking the number of time block that the job will take.
  + Bill Generator
    - Once the job has been complete, the assistant will input the number of time block that provided to generate the payment for the customer
  + Retrieving Reports
    - Function can access to database, turn the data into a report when the assistant when then assistant want to obtain a report
  + Notification
    - Function that will send both no-reply text and email to customer and technician
  + Survey
    - Upon job completion, a survey will send to the customer for further analysis of the technician, improvement and customer satisfaction.

**4.2 Software Interface Description**

**The software interface(s)to the outside world is(are) described.**

**4.2.1 External machine interfaces**

**Interfaces to other machines (computers or devices) are described**

**Most computing will be done in a Microsoft Windows OS environment, however for use cases Notification and Survey, external devices, such as user computers, cell phones, tablets, will be used for technicians and customers to receive emails.**

* + - Login
      * Manager
      * Assistant
        + New Job
        + New Customer
        + Waiting List

Update

* + - * + Technician

Add

Delete

* + - * + Modify Customer

Customer Contact

Customer Priority

Services Request

* + - * + Payment

Cash

Check

Credit Card

* + - * + Report

Invoice report

Waiting-list report

Statistics report

Services request assignment report

Outstanding job report

Wait Time

Average wait time

* + - * + Survey

Rate Technician

* + - * Technician
        + Report Job Status

**4.2.2 External system interfaces**

**Interfaces to other systems, products or networks are described.**

* + - **External System interface**
      * All database storage will remain in the internal system. External networks will be accessed to send emails to customers or technicians for the use cases Notification and Survey.

**4.2.3 Human interface**

**An overview of any human interfaces to be designed for the software is presented.**

**4.2.3.1 User screen interface layouts**

**Be sure to include “exception” screens, if any**

We will use a graphical user interface to implement our design. All coding will be done in C++ and program will be accessed through the console in Windows 0S. Exception screen will occur if a user fails to login or enters invalid information.

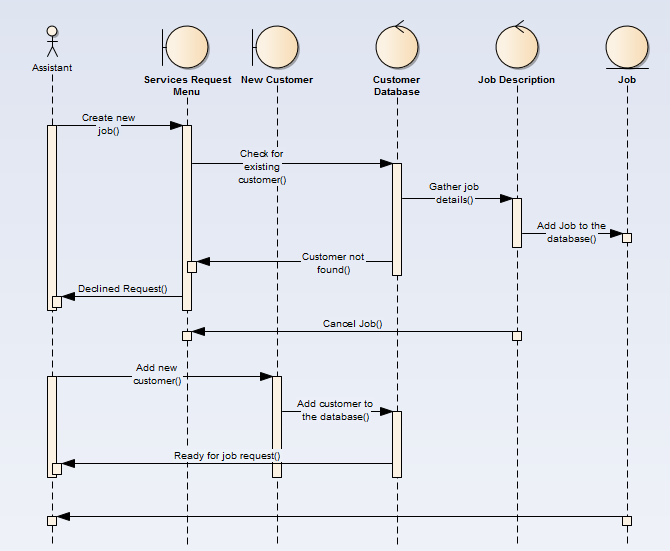
**4.2.3.2 Report layouts**

**Be sure to include “exception” reports, if any**

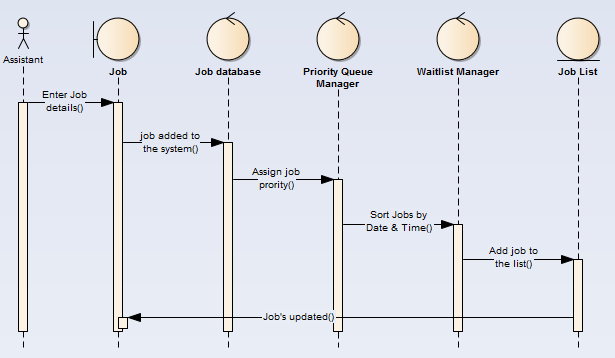
**4.3 Sequence Diagrams**

**Used to model the class interactions needed for the use cases.**

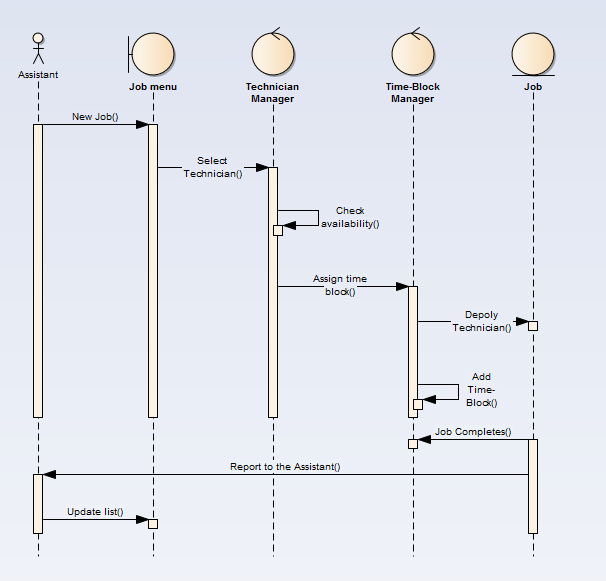
**Customer and Service Request:**



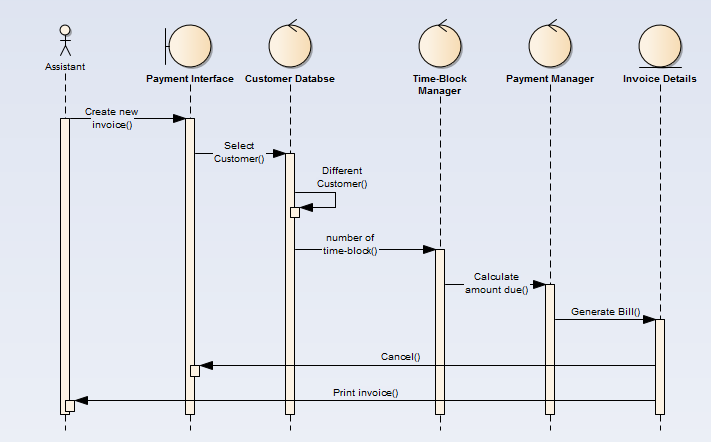
List and Priority:



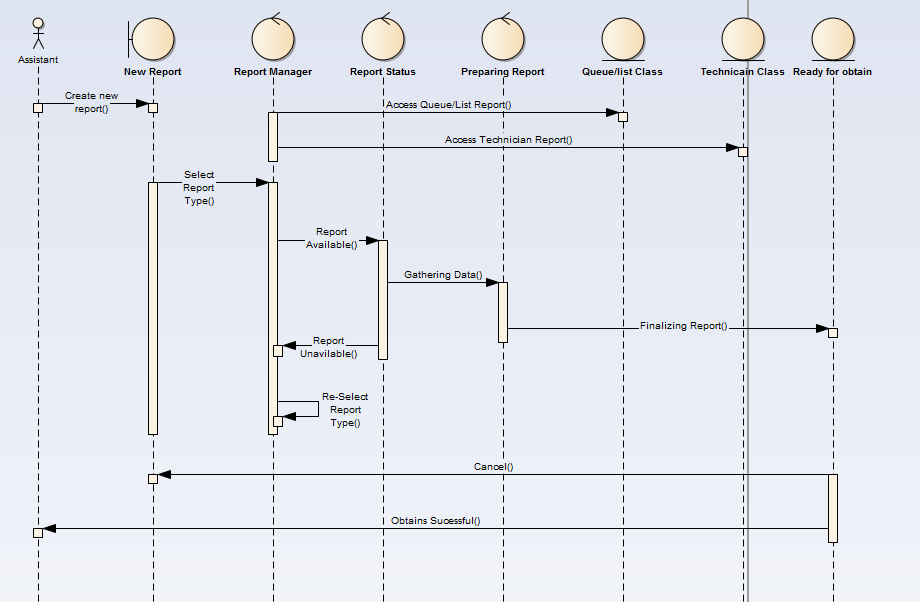
Technician and System + Time Block and Jobs:



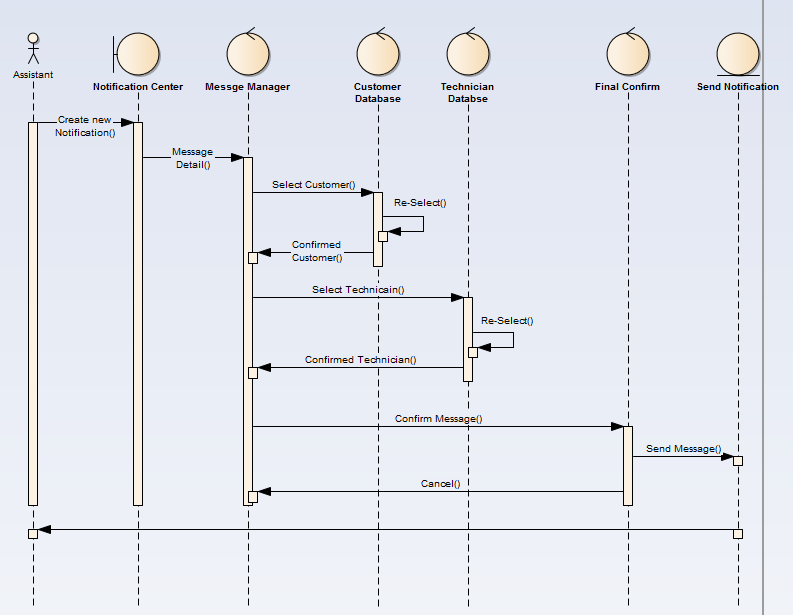
Bill Generator:



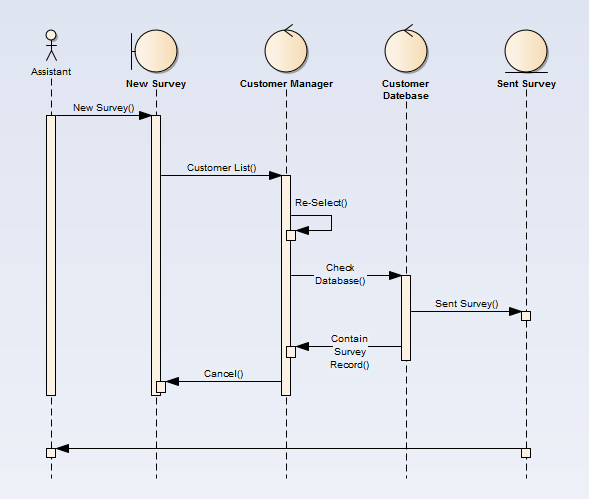
Retrieving Reports:



Notification:

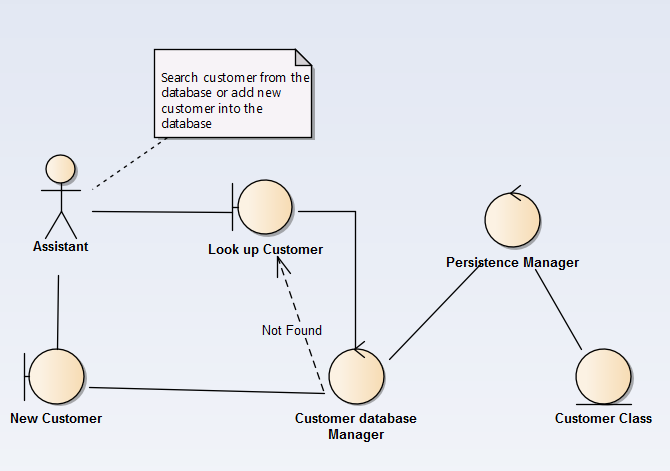


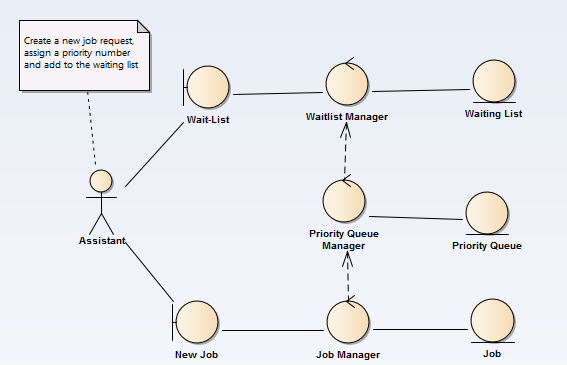
Survey:

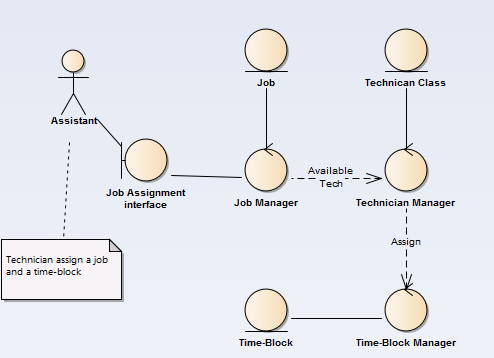


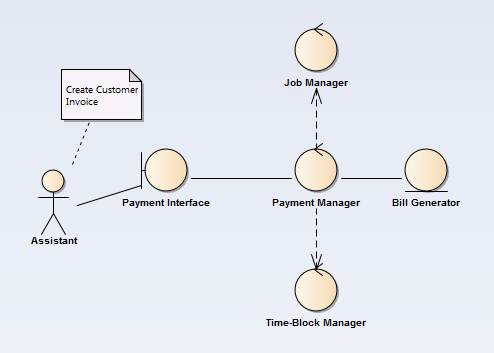
**4.4 Communication Diagrams**

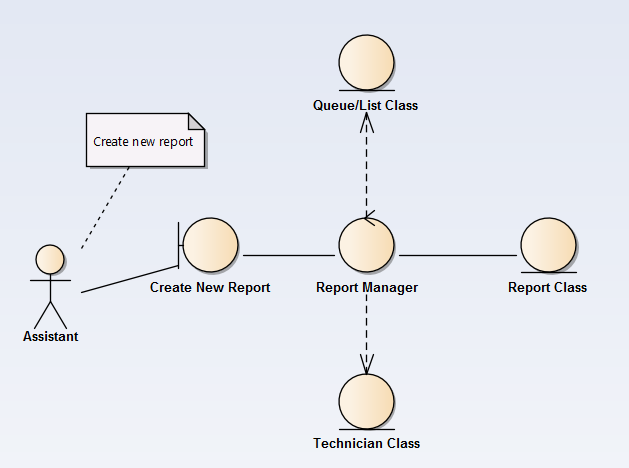
**Used to model the message passing structure of the system functions.**

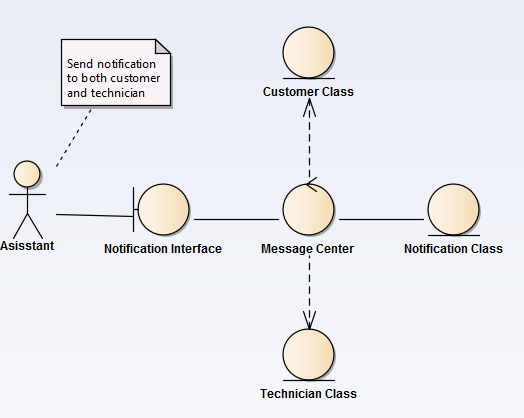


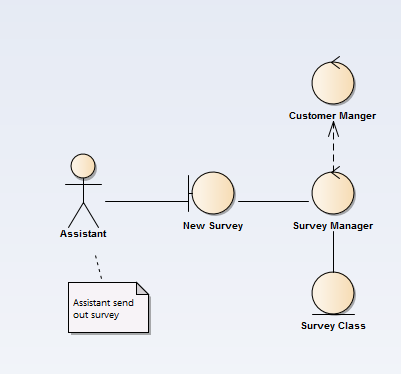












**5.0 Behavioral Model and Description**

**5.1 Events**

The customer will call Osric Company

-The customer can request a service

-The customer can request information regarding time wait

The assistant will:

-Ask for company name, customer number

-Check if the customer is in the existing database

-if the customer is new, will request manager

-if the customer is already in the database, will process service request

The manager will:

-Accept new customer request

-Decline new customer request

-Report status to the assistant

The assistant will process the job request if:

-The customer exists in the system already

-The manager has approved the new customer

The assistant will:

-Record the job information

-Date, Time, Description, ID

-Assign the job in the priority queue

-P0, P1, P2, P3, P4

-Add job to the Waiting list

-Sort by time and date, if necessary

-Check list status

-Number of jobs

-Total amount of wait

-Report status to customer

The assistant will assign job to technician to complete a determinate job.

-Check technician status

-Check the first job in the waiting list

-Assign technician to job

-Update list

Technician will complete a job

-Works 8-hour shifts

-Day/ Night shift

-Will take time block to complete job

-Time block has total time, 4-hour block, extended block

-Will complete a job card at the end

-Will report back to the assistant when done.

The assistant will generate a bill

-The assistant will receive the job card with time block info

-The assistant will be able to charge the total amount by

-Day/Night rate

-The assistant will notify the customer

-mail

-email

-text

The customer will:

-Receive the bill

-Pay bill

-Seek out information regarding service cost

-Rate the technician

-1 to 5

-Rate Osric’s services

-1 to 5

The assistant will be able to generate a list of reports

-Traditional scheme Vs Osric

-Overall Jobs

Job duration

Average waiting time before a job is started

Standard Deviation and mean

Shortest Time

Longest Time

Cost

-When customer calls

How soon a service will be available

Length of the current waiting list

Average time to complete job

Worst Case (length to reach priority 4)

-Priority Queue

Average queue length

Percentage that the queue is empty- day & night

-Technician

Number of blocks when a technician is idle

Number of jobs that cannot be continued because no technician is available

**5.1.2 States**

1) Customer calls Osric

2) New Customer Manager/Approval

3) Existing Customer Assistant processes request

4) Job is sent to Priority Queue

5) Job is sent to Waiting List

6) Job is picked up by the Technician to be completed

7) Job is completed by the technician

--> Report status to assistant

--> Assistant will update queue

8) The assistant will generate a bill

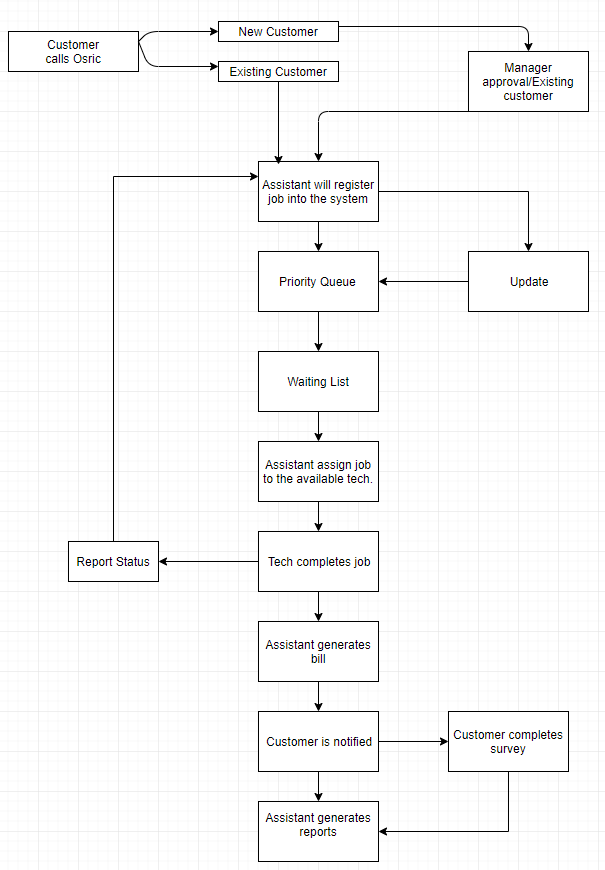
9) The customer will be notified

10) The customer will take a survey (optional)

11) The assistant will generate reports

**5.2 State Transition Diagrams**

**If it needs modifications, I have a saved copy in the drive.**



**6.0 Restrictions, Limitations, and Constraints**

**Performance/Behavior constraints**

* Response time from system once data has been "fed"
* Size of data
* Client to software communication
* Software design
* Graphical User Interface (GUI)

**Management constraints**

* Resources
* Time
* Place
* Policy
* Human behavior

**Technical constraints**

* Lack of experience
* Missed requirements
* Poor design
* Unpredicted risks
* All knowing different languages

**7.0 Validation Criteria**

**The approach to software validation is described.**

**7.1 Classes of tests/Test Strategy**

**The types of tests to be conducted are specified, including as much detail as is possible at this stage. Emphasis here is on black- box testing.**

**Customer:**

**Marco**

* **Customer Name:** Characters
* **Company Name:** Letter, space, character, special character, int
* **Company Phone number:** int
* **Company Address :** int,string

Manager:

* **Accept or Decline** : Boolean

Assistant:

* **Date Request :** Letter and number
* **Job Description :** Letter/Characters, numbers
* **Priority :** Number
* **Identification Number**: Number
* **Service Request Numbe**r: Number

Technician

* **Technician name**: Character
* **Technician Numbe**r: Number
* **Time/Date Completed**: Characters, numbers
* **Time Block:** Number
* **Day Blocks**: Number
* **Night Blocks**: Number

Bill:

* Total amount due (Calculated by time blocks): Float
* Due Date: Number or String

**7.2 Expected software response**

**The expected results from testing are specified.**

**Statistics:**

* Provide Average time on waiting list: time and date
* Provide Standard deviation and mean of the completed task
* Average Time in each priority
* Total Time

**7.3 Performance bounds**

**Special performance requirements are specified.**

**Additional Information:**

* A standard first come first serve will then be compared with the original software and determine which would be more efficient to use.

**8.0 Appendices**

**Presents information that supplements the Requirements Specification**

**8.1** [**System traceability matrix**](https://d.docs.live.net/9f4a43508d51317f/Documents/Cis375/Artifacts/System%20Traceability%20Matrix.xlsx)

In the folder submitted under 8.1\_System Trackability Matrix

**8.2 Product Strategies**

**If the specification is developed for a product, a description of relevant product strategy is presented here.**

The strategy of our product is to make sure there is limited mistakes by the systems end. Our product should work in an efficient manner along with it being easy for the user to enter in the data needed. We also plan on having this system decrease the waiting time for customers with a higher priority to make sure they are satisfied compared to the typical first come first serve.

**8.3 Analysis metrics to be used**

**A description of all analysis metrics to be used during the analysis activity is noted here.**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Objective/ Deliverable Metrics** | **Responsible for Delivery** | **Supporting Team** | **Time (In hours)** | **Delivery Date** |
| **1.1 Goals and Objectives** | **Mohammed** | **Marco** | **1.5** | **7/20/2018** |
| **1.2 Statement of Scope** | **Mohammed** | **Marco** | **.25** | **7/20/2018** |
| **1.3 Software Context** | **Mohammed** | **Marco** | **.25** | **7/20/2018** |
| **1.4 Major Constraints** | **Mohammed** | **Marco** | **.25** | **7/20/2018** |
| **2.1 User Profiles** | **Jason** | **Marco** | **.5** | **7/16/2018** |
| **2.2 Major Software Functionality** | **Jason** | **N/A** | **.5** | **7/16/2018** |
| **2.3 Special Usage Considerations** | **Jason** | **N/A** | **.5** | **7/16/2018** |
| **3.1 Data Description** | **Maricruz** | **Martin** | **20** | **7/24/2018** |
| **3.1.1 Entity-Relationship Diagram** | **Group** | **Group** | **3** | **7/21/2018** |
| **3.1.2 Data Flow Diagram** | **Maricruz** | **N/A** | **4** | **7/26/2018** |
| **3.1.3 Object Relationships** | **Maricruz** | **Marco** | **2** | **7/26/2018** |
| **3.1.4 Complete data model** | **Marco** | **Mohammed** | **2** | **7/24/2018** |
| **3.1.5 Data dictionary** | **Maricruz** | **Marco** | **3** | **7/24/2018** |
| **4.1 Use Cases** | **Jason** | **N/A** | **1** | **7/22/2018** |
| **4.2 Software Interface Description** | **Audrey** | **Jason** | **1** | **7/22/2018** |
| **4.3 Sequence Diagrams** | **Audrey** | **Jason** | **20** | **7/25/2018** |
| **4.4 Communication Diagrams** | **Jason** | **Audrey** | **5** | **7/25/2018** |
| **5.1 Events** | **Martin** | **N/A** | **3** | **7/18/2018** |
| **5.2 State Transition Diagrams** | **Martin** | **N/A** | **4** | **7/18/2018** |
| **6.0 Restrictions, Limitations, and Constraints** | **Marco** | **Audrey** | **1** | **7/26/2018** |
| **7.1 Classes of Tests/ Test Strategy** | **Marco** | **N/A** | **2** | **7/26/2018** |
| **7.2 Expected Software Response** | **Marco** | **N/A** | **.5** | **7/26/2018** |
| **7.3 Performance Bounds** | **Marco** | **N/A** | **.20** | **7/28/2018** |
| **8.1 System Traceability Matrix** | **Mohammed** | **Group** | **4** | **7/30/2018** |
| **8.2 Product Strategies** | **Marco** | **N/A** | **.20** | **7/30/2018** |
| **8.3 Analysis Metrics to be used** | **Marco** | **Audrey/Maricruz** | **2** | **7/30/2018** |
| **8.4 Supplementary Information** | **Group** | **N/A** | **9** | **7/30/2018** |

**8.4 Supplementary information (as required)**

In the folder submitted under 8.4\_Build ERD