Incident Management and Decision Support System

# Objective of the project

The customer has a dot-net based request management system. The problems with the system are

1. No centralized tracking and storing of information from current and past incidents
2. Repeated incidents need quicker resolution and preferably an automated process.
3. Current system is in older Dot net framework and will be migrated into MVC Dot Net framework. (not in scope of this project, but ongoing project to get this done by the client)

The proposal is to develop a system that can

1. Develop a knowledge base system to store incidents and resolutions for future reference
2. Develop a decision support system using the knowledge base to help efficient incident resolution
3. The solution should be plugged in to existing system with little or no integration changes

# Purpose of this document

To request needful information to initiate and complete requirements elicitation process through outlining the assumptions and raising relevant questionnaire to which the answers from the customer would help in arriving at right business requirements to complete the development.

# Technical (hardware/software) questionnaire/assumptions

1. List of technologies used in the current application to which the plug in will be integrated and their versions?
   1. Windows server version?
   2. Server configuration
   3. Database servers and version used
   4. Dot net framework version
   5. Any third party tools used?
   6. Iis version

Will assume latest servers and we can make the code compatible if any version is different

1. Is the existing system custom built or off-the-shelf product customized per customer needs?

Assuming custom built application. The to-be-built product will be standalone system that can plug into the in-house application.

1. Assuming the system is web based application, is it an intranet or internet application? And how does the authentication work (windows or app specific?)

Assuming windows authentication as it is quite common with dot net apps. It is intranet application.

1. About the iis on the windows server hosting the current application, canor should we use the existing application pool and application directory or should we build a new instance?

Build new stand alone instance

1. Will we be given remote access into your developer Virtual machines or we will develop the code in our computers and migrate once development is complete?

We will build on Hexaring developer machine and migrate code once complete. The sample data will be assumed

1. Can we receive the ER diagram of the key tables that we have to use in our system to ensure we don’t duplicate the database objects in our data model?

We will build the data model based on the needs of a typical decision support system.

1. Will we be given sample data model and data (masked and dummy data is also acceptable)

NO

1. Are we using any third party tools in the existing system or is the existing system integrated with any third party tools?

Assuming none

1. Any web services that are integrated with the existing application and method of authentication (for e.g, token based authentication etc)

Unknown.

1. Is it possible to take the existing code and develop the new features on top of it (code will be accessed through client’s VM only in that case) or the code should be developed as a separate module and should be integrated

No existing code shared for now

# Functional requirements

1. Briefly outline the features of the existing system and desired features of the new system

The new system will get the incident information including the images or documents with the details of the incident and resolution comments. This information will be stored and key words will be identified automatically based on the commonly occurring patterns. When the new customer tries to create new incident or would like to know about this issue from the past, the information will be fetched without going through repeated steps again.

1. If there is a system for existing incident management, how is it resolved and closed? Do we have the closing comments in a physical file format or somewhere in the database etc?
2. Who are all the stakeholders of this system?
   1. Help desk team
   2. Mechanical crew – any specifics?
   3. Engineering team –
   4. Hospitality crew
3. Who will submit the incident and what are the methods (call? Web Application? Mobile?) by which the incidents will be submitted into the application?

Assume call, Web Application and Mobile.

1. Once the incident is resolved, is someone updating the incident with resolution comments and steps before closing the request/incident?

Assume yes

1. What are the broad categories of the types of incidents expected in the system? E.g,
   1. New hardware needed.
   2. Existing hardware to be repaired and needs professional service or a handy man on board etc
   3. Emergency – non health conditions (weather)
   4. Emergency - Health conditions
   5. Hospitality related care needs
   6. Missing items
   7. On board announcements
2. If we develop a knowledge base, who will be the direct user – will the person opening the incident search for the similar historical incident or help desk does the research and suggests the crew that needs help? How can crew access this data if they do have access?
3. Do we have an asset management system that keeps track of all the items purchased and its lifetime, specifications etc?

# Non-Functional requirements

1. What was the precipitating incident driving this initiative?
2. Who will own system/process (role) in the project and after the project is live with the client?
3. **Specific** goals/outcome? Specific to what the interactions look like, and what specific goals are expected? (Decrease in xxx issues/incidents, improved maintenance process, improved customer interactions/experience, etc)
4. Short and Long-term goals of the application or the systems in scope? (3 months/6 months/3 years) – how do we like the system to be and what new features are anticipated, if any?
   1. Integration of incident management and decision support systems
5. Examples of all the existing systems relevant to the system in scope for development or that integrate with this application?
6. What do we consider the success of the application? Rank them in order if possible.
7. What do we consider as risks to achieving the success? (in terms of functionality and customers utilization of the developed features and **not** about the project delivery)

# Next steps

Once we have the brief understanding of the existing systems and desired features, we can propose a solution with the MVC framework in a loosely coupled manner so it can be easily plugged into the existing system.

We can propose a blue print of the architecture and upon approval, we will share the project plan, milestones and resources schedule. The target is to complete the development in 6-8 weeks or less.