Final Project

Tracy Mann

CSE210

Bro. Parrish

Section A8

6/30/2023

v1.0

# What does the program do?

At my Job, at Retail Business Services, my team manages the Advanced Marketing Solution product. This product was originally called Copient before it was purchased by NCR. The code is proprietary to NCR and their AMS team of engineers and developers perform all code changes, patches and fixes internal to the code. Our business users will work with our architects to request new features to be added to the product from NCR. When NCR completes any code fix, patch, or upgrade that the business has determined that they want deployed to production, it is my team’s responsibility to evaluate the readmes, release notes, and any installation instructions in order to build a plan to test and implement this code, through the product lifecycle in our organization.

Currently, our team uses a few limited MS Project licenses for upper-level planning that will be presented with power point slides to our director. These plans are constructed by our team leads and manager from lower-level plans completed in MS Excel. There is no template and the format typically changes from person to person and project to project. This reduces the ability to take lessons learned and best practices easily from one project to the next.

My focus, is primarily on the implementation plan side of this process. This program should allow a user to enter tasks required to do to complete the implementation plan. Some of the tasks may also be designated as visible benchmarks that should be tracked as such. There are also some tasks that are deemed as mitigations for risks in a plan or process, these are also identified separately. There are also benchmarks that are a decisive point to either continue or turn back and return to the previous version, these are referred to as go/no go points. In our plans, we typically have a single go/no go point. This can trigger the safety device for change control called a back-out plan. In some cases, the back-out plan is the implementation plan reversed, but there exist other special circumstances that may force the back-out plan to follow a different process than a simple reversal of the original plan.

For this program to help people build and manage implementation plans used to deploy code it will have the following types of tasks:

* Task – A step required to be completed in order to complete the process.
* Benchmark Task – A task that shows a measurable progress point that can be reported to the various stakeholders in the process as a progress point.
* Go Or No-Go Point – A Benchmark task that has a decision point on if the next task in the implementation plan is to be performed or to begin the back-out plan.
* Mitigation Task – A task that works as a workaround or prevention to an identified risk in the process.

Another supporting object will include:

* Risk – An identified potential situation or state that may cause failure of the implementation, process exceptions or even undesirable results during the process.

The tasks will need to be held one of two container objects as follows:

* Implementation Plan – The collection of tasks, benchmarks, mitigation tasks, and go or no-go points required to complete an implementation.
* Back-Out Plan – The collection of tasks, benchmarks, mitigation tasks required to back-out an implementation. This will need to have a complete backout entry point, in the event that all steps in an implementation plan have been completed as well as an entry point for any no-go decision point in the implementation plan.

The plans can be unassigned templates, paper test runs, developmental runs and production runs.

## Functional Requirements:

The program must do the following:

1. Allow the user to define the name and description of the plan and designate a plan manager.
2. Allow the user to add, edit, copy and delete tasks in the plan.
3. The types of tasks must be allowed to be regular tasks, benchmarks, go/no go points, and risk mitigation tasks.
4. Tasks will be ordered based on pre-requisites listed to a task.
5. Before a plan can be executed the tasks must have a person assigned to them.
6. Before a plan can be executed all risks must be mitigated or accepted as confirmed that the risk is acceptable.
7. While executing the plan, the original estimated plan completion time and the updated based on current results estimated completion time will be displayed.

# Class Diagram

