Word document has to have all of the deliverables in it

Powerpoint for the presentation

**Gantt Chart** – horizontal bar chart that represents a grouping of tasks, good for upper management. When will you get the project done. Don’t need arrows between the bars. Should never exceed 1 page.

Microsoft project can make Gantt and PERT charts. You must put in all of the details and then change the view to make the Gantt or PERT charts.

data acquisition plan – need to be in Gannt Chart (part of WBS)

testing plan - need to be in Gannt Chart (part of WBS)

**Work Breakdown Structure**- break it down into big chunks, then smaller chunks. You want to break it down into phases or stages, a phase or stage is usually 1 to 2 months and then Identify Major Deliverables, these are called steps, steps break down to the level of a task. Task can be referred to as an activity, it is a unit of work. This is assigned to a person. Tasks should be 10 days or less. Over is too big, break it down into smaller. With a task, you begin with a verb + deliverable. Every task produces 1 output.

Exam 1 needs to be a milestone in your Work Breakdown Structure

Milestone for a week prior to the deadline of the project for practice and presentation.

DETERMINE DEPENDENCIES – every task must have at least 1 output, that output will be the input to something else

Quality Control – formally review a deliverable

Formalize activities

Prepare the Review

Do the review

Changes as a result

Put this in Work breakdown Structure

data acquisition plan

testing plan

**Business case** – benefits cost and risk, should have a mitigation

**Idea for Business Case**

**First Option**

One option is using Amazon Relational Database Web Services (AWS RDS) from Amazon Web services as a cloud storage method for the company’s data. This will provide a low-cost option for the business to use with the Web Database application we develop. You only pay for each database instance that you launch. The storage ability is scalable if storage needs increase. The service provides automated backups and point-in-time recovery for database information which would provide durability and availability of the service. AWS RDS allows for the ability for initial database creation and manual entry of data that can be used to manually enter the current Information of the Emily Bakes Cake paper file system. The Civic Systems company would be able to manually integrate this information to the cloud service using the MySQL Workbench which is free and would not add to the cost of the new system. The AWS RDS My SQL database engine would be the best choice on the service because of the low cost.

A possible risk is that the closest endpoint for this service is in Ohio which could potentially cause latency issues. However, with the expected use of a small business like Emily Bakes Cakes, the impact of this latency on the business would likely be minimal

Another possible risk is that if there is an issue with the service on the AWS end, the Emily Bakes Cake company would not be able to access their data. This would mean that the company could not get updated orders, employee or customer information from their cloud service. They also would not be able to add or update anything in their cloud service.

**Second Option**

Another option could be to use a free database application like My SQL to create the database tables and hold the data required for the business. My SQL is free, and this option could help to keep costs low. However, this would require a computer upgrade and the purchase of a decent amount of storage. To maintain the system, the occasional hiring of an IT freelancer would be necessary for maintenance updates, altering the database and IT support issues.

The scalability of this type of system would be limited

If there was an issue with this system, there could be significant downtime until an IT freelancer would be able to restore availability

Multiple connections could require RAM or CPU upgrades if performance issues were to occur.

**Feasibility**

Economic feasibility – benefits and costs, TOC total cost of ownership. Calculate the benefits and costs for the entire life of the project.

Development cost -

Implementation /use cost -ex: ink for a printer, over time it is expensive

Technical feasibility – do you have the equipment, infrastructure to do the project, do you have the technical resources to do the project (GREY area between technical and operational)

Operational feasibility – will it actually be used; do you have the people capable of using it.

**Problem and requirements list**

Column for ID number (NEVER CHANGES)- this is done for requirement tracability

Column for who identified the problem

Column for category

Column for ranking

Column for potential solution (don’t overpromise, not mandatory)

\*\*\*\*\*\*\* do a 2 level sort first by category and second by ranking

**Entity Relationship Diagram**

Not about Processes, its about data being related to data

Names of entities must be singular

Verb on line between entities (from parents perspective)

1 to many relationships

The one side is always with the parent

The many side is always with the child

If you have a many to many relationship, turn it into an associative entity and connect the other 2 entities to it. The name of the associative entity is the 2 other entity names put together

Dropdown Lists of prototype need to be an entity in the Entity Relationship Diagram

**Data Flow Diagram**

Anything in the scope boundary of the scope diagram becomes the 0 process box, the connected flows must be going to the process box

DFDs do not show sequence, the number on the box of the processes does not denote sequence

Data store – place where data is resting

Each flow has to have a different name

Double headed arrows are okay but two one way arrows is better

At the next level of drilling down, the process boxes are 1-9, no more than 9

7 boxes max is better than 9

data stores definitely need to come down when you drill down into the process

**Data Dictionary**

YOU COULD HAVE A COLUMN OF COMMENTS IN THE DATA DICITONARY to have the way you want the user to input the information

**Issues log** – a spreadsheet, a way for the team to discuss issues, may or may not actually impact the team

**Problem log** – a spreadsheet of problems, things that are impacting the team, problem log should be smaller than the issues log.

**Change log** - The impact could be so grave that it now involves a change in a past/future deliverable. Change log should be the smallest.

Change Control –change control is changing the requirement, keeping track of changes within the teams so you can communicate it

**Organization Chart**

DO NOT USE ARROW HEADS, BOXES AND LINES ONLY

POSITION MATTERS

Sponsor on top – project manager – assistant -exc.

**Reports**

When designing reports, the essential goal is to match the report to the user’s specific information needs. Depending on their job functions, users might need one or more of the reports described in the following sections

Exception Report

Displays only those records that meet a specific condition or conditions. Useful when a user wants information only on records that might require action but does not need to know the details. For example, a credit manager might use an exception report to identify only those customers with past due accounts, or a customer service manager might want a report on all packages that were not delivered within a specified time period.

Summary Report

Upper-level managers often want to see total figures and do not need supporting details. A sales manager, for example, might want to know total sales for each sales representative but not want a detail report listing every sale made by the item. Or a personnel manager might need to know the total regular and overtime hours worked by employees in each store but might not be interested in the number of hours worked by each employee

Control break – usually causes specific actions to occur, such as printing subtotals for a group of records

Report header – appears at the beginning of the report, identifies the report and contains the report title, date and other necessary information

Report footer – appears at the end of the report, can include grand totals for numerical fields and other end-of -report information

Page Header or Page Footer – displays the report tile

REPORTS \*\*\*\*\*\*\*\*\*\*\*

Purpose? What is the purpose of report

Why would your client need this?

Does your report give them the answer?

Identify the purpose, identify the data, ask your self if that physical thing satisfies the purpose

\*\*\*\*\*\*MOST IMPORTANT STUFF NEEDS TO BE IN THE UPPER LEFT AS MUCH AS POSSIBlE

Reports should have important information put closer together

Self-Contained

You should be able to make a decision by only looking at the report and nothing else

**Ideas for Reports**

Monthly Report on Custom cake sales

Quarterly Report for Custom Cake sales

Monthly Report on Employee time spent in each stage of the custom cake process

Quarterly Report for Employee time spent in each stage of the custom cake process

Monthly Detail Report on order cancellations

Quarterly Detail Report on order cancellations

Monthly Report on completion time for customized cakes

Quarterly report on completion time for customized cakes

Yearly Report on employee status changes

Yearly Report on Customer Status Changes

Quarterly report on changes of prices on cake menu

Monthly report on changes of prices on cake menu