**DBAS32100 - Project Proposal**

Date: \_\_\_\_\_July 14th, 2023\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

Class: \_\_\_\_ DBAS32100 (Friday 11am – 2pm) \_\_\_**\_\_\_\_**\_\_\_

Group: \_\_\_\_\_\_**\_**\_ GR\_32122\_3 \_\_\_**\_\_\_**\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

**Responsibilities:** The deliverable lead is ultimately responsible for the quality of the deliverables. They are also responsible to check for any academic integrity violations.

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| **Deliverable** | **Task** | **Members Names** | | | | **Deadline** |
| **Kyle** | **Sayaka** |  |  |
| Task 1 | 1. Deliverable Leader | **○** |  |  |  | **July 31st (Mon)** |
|  |  |  |  |  |
|  |  |  |  |  |
| Task 2 | 1. Deliverable Leader |  | **○** |  |  | **July 31st (Mon)** |
|  |  |  |  |  |
|  |  |  |  |  |
| Task 3  ER | 1. Deliverable Leader |  | **○** |  |  | **July31st (Mon)** |
|  |  |  |  |  |
|  |  |  |  |  |
| Task 4 | 1. Deliverable Leader | **○** |  |  |  | **August 6th (Sun)** |
|  |  |  |  |  |
|  |  |  |  |  |
| Task 5 | 1. Deliverable Leader | **○** |  |  |  | **August 6th (Sun)** |
|  |  |  |  |  |
|  |  |  |  |  |
| Task 6 | 1. Deliverable Leader |  | **○** |  |  | **August 6th (Sun)** |
|  |  |  |  |  |
|  |  |  |  |  |

**Task Dependencies**

**A diagram of a system

Description automatically generated**

**State in this box what tasks depend on other tasks. Dependencies should be also reflected in the task deadlines.**

All dependencies stated in this document are considered Finish to Start dependencies. The Task Dependencies identified for this Project are as follows:

**Task 1: Predecessor to Task 2.**

Task 1 requires that the Address Table be refreshed with up-to-date data from the Master SQL Table. To verify the addresses in Task 2, this task must be completed. It is also notable that it is during this task that the bonus part will be completed.

**Task 2: Successor to Task 1. Predecessor to Task 4.**

Task 2’s requirement for the Address Table to be refreshed necessitates it succeeding Task 1. Task 4 includes uploading and formatting the data uploaded in Task 2 into the Star Schema (Data Mart), requiring Task 4 to succeed Task 2.

**Task 3: Predecessor to Task 4.**

Task 3 involves designing the Star Schema and Entity Relation Diagram for the Donations Data Mart. Task 4 cannot be started until at least Star Schema has been created, which requires it to be a successor for Task 3.

**Task 4: Successor to Task 2 and Task 3. Predecessor to Task 5.**

Task 4 involves loading Donations Data from the Central Donations Repository into the donations Data Mart. This requires that the Star Schema (Data Mart) to be created as in Task 3 and that the donations data has been uploaded as in Task 2. Task 5 involves creating views stored in the created Data Mart, requiring Task 5 to succeed Task 4.

**Task 5: Successor to Task 4. Predecessor to Task 6.**

Task 5 requires that the Data Mart first be created before the views to access the specific data can be stored within. Task 6 involves creating user roles and security to access the Database and the Task 5 views, requiring it to be a successor to Task 4.

**Task 6: Successor to Task 5.**

To create the users which can access the database and views, all the previous tasks must be completed, and the views must be created as in task 5.

**Proposed Solutions**

**Proposed Solutions for Tasks 1, 2 and 4**

**Task 1: Refresh the Address Table in the Oracle Central Donation Repository**

In Task 1, the data will be loaded from the Master SQL Table into Talend Open Studio, where necessary transformations will be performed on the data to make it valid for integration into the Oracle Address Database. This includes performing title-casing for strings, to accommodate the support for case sensitivity in Oracle. In addition, Talend Open Studio supports the use of Restful APIs, which can be used to retrieve the Postal Code for each address before inclusion in the Oracle database, which will at this point complete the bonus part of the project.

To include a sequential ID in the Oracle database for each address, a trigger will be included in the Database which will automatically increment a sequential ID when the data is inserted. Alternatively, the ID may be generated in Talend, though this would necessitate wiping the Oracle table first to prevent duplicates.

**Task 2: Load the Donation Data from .csv Files into the Central Donation Repository**

In Task 2, Talend Open Studio will be used to load the .csv files from their containing folder. The tMap component will be used to classify the valid and invalid data. The valid data will be uploaded to the Central Repository, while the invalid data will be exported to a .csv file for later analysis.

When the .csv files are loaded, some work will be required to ensure that the data types of the .csv files match the data types of schema in the Central Donation Repository, and .csv files do not contain the metadata for correctly identifying the data type of containing values.

**Task 4: Load the Data in the Central Donation Repository into the Data Mart**

In Task 4, the data from the Central Repository will be loaded into the Star Schema (Data Mart). This Task will be performed entirely with PL/SQL. A cursor will be used to retrieve the Donation Data, which will then be looped through to execute update queries on the Data Mart. If the ID for the Data already exists within the Data Mart, it will be updated with the most up-to-to date data, otherwise all new data will be inserted into the Data Mart.

**Group**

**Members: NAME SIGNATURE**

1. **\_\_\_\_\_\_\_Kyle Galway \_\_\_\_\_\_\_\_\_\_\_\_Kyle Galway\_\_\_\_\_\_\_\_**
2. **\_\_\_\_\_\_\_Sayaka Nakamura \_\_\_\_\_\_\_ Sayaka Nakamura \_\_**