Criterion B: Design

Functional Decomposition Diagram

The functional decomposition diagram breaks down the whole solution system into the sub-functions that it should cover. By assembling a functional decomposition diagram, I am able to plan and establish the functions I need to have integrated into the final system.

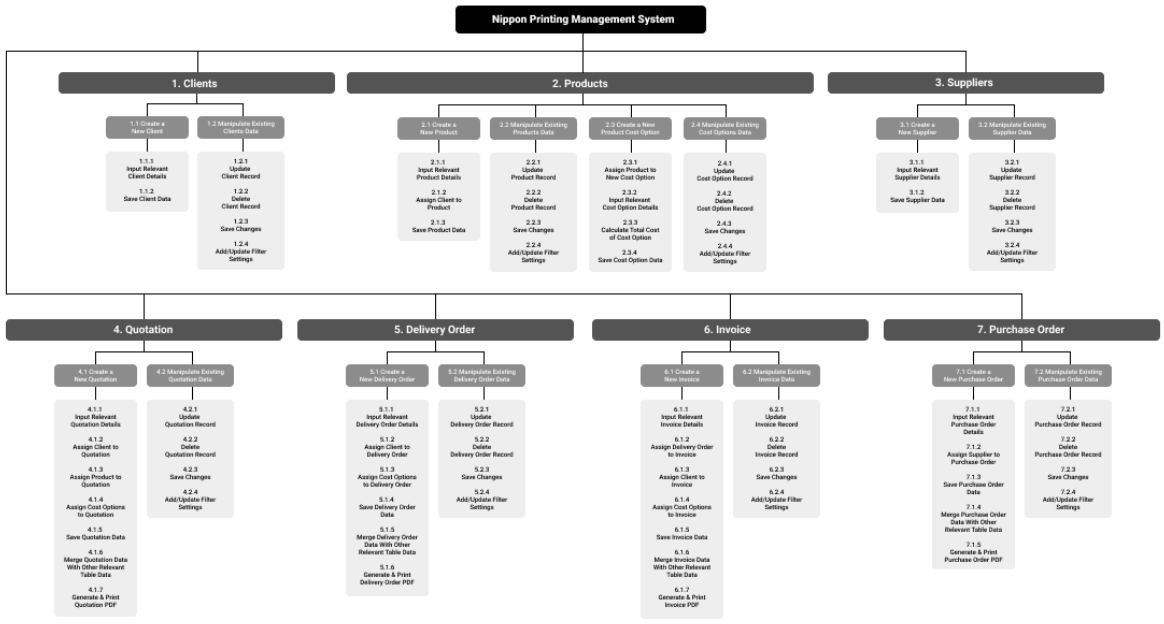

Figure 1. Functional Decomposition Diagram

NOTE: As the diagram is very detailed, it is advisable to use the zoom in feature to get a closer analysis.

Entity Relationship Diagram

It can be difficult to understand how all the elements (e.g. fields) of a database will interact and relate to each other. An entity relationship diagram is a visual way to depict how all the elements will relate to each other and work together.

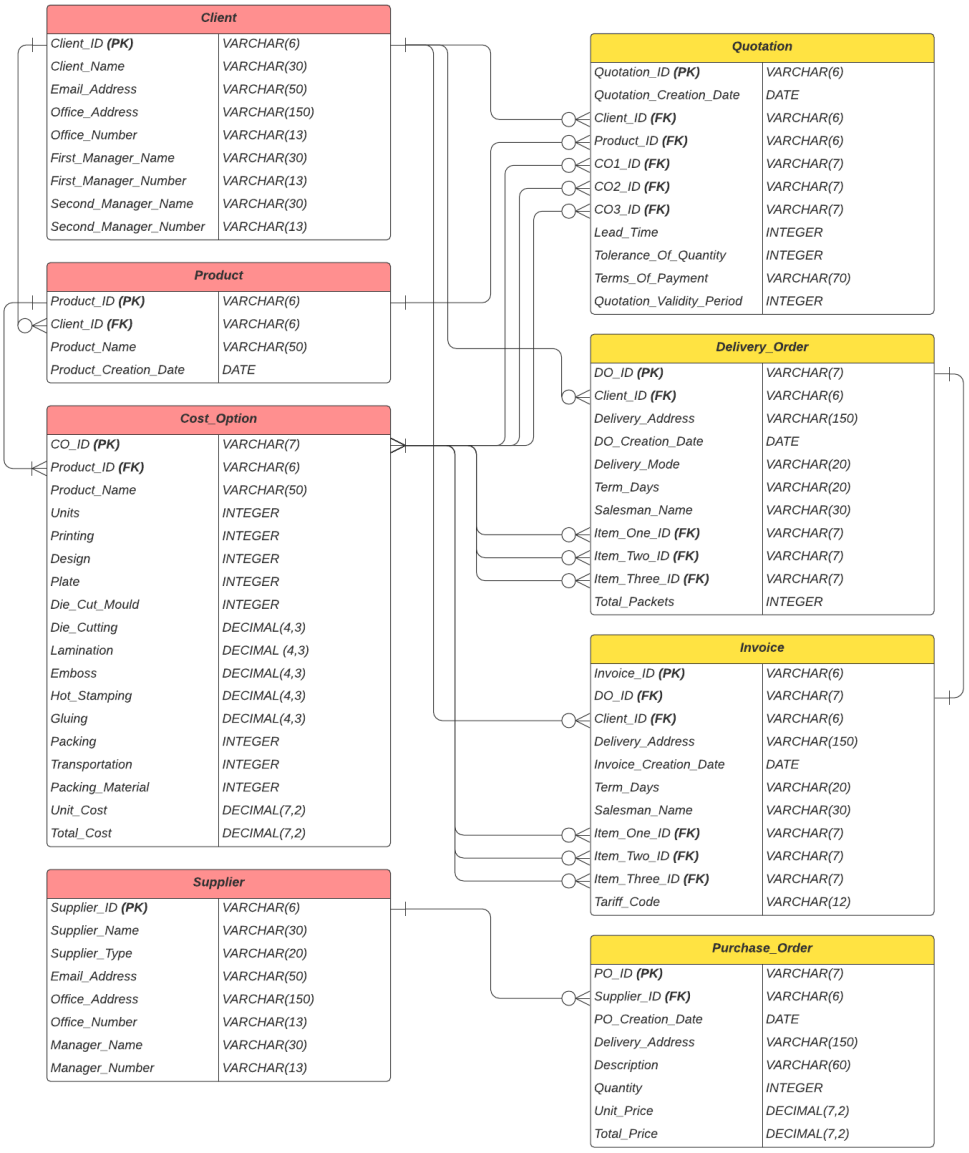

Figure 2. Entity Relationship Diagram

Data Dictionary

The data dictionary consists of all the tables that will be created and used in the database, and their columns, data types, constraints and a quick description of the contents of each field.

Note:

- **PK:** Primary Key
- **M:** Not Null
- **U:** Unique

- **FK:** Foreign Key

Client			
Column Name	**Data Type**	**Description**	**Constraints**
Client_ID	String	Auto-incrementing, auto-generated ID to identify specific clients. (e.g. C-0001)	PK, M, U
Client_Name	String	The name of the specific client.	M, U
Email_Address	String	The business email address of the client.	U
Office_Address	String	The office address of the client.	U
Office_Number	String	The phone number of the client's main office.	U
First_Manager_Name	String	The name of the main manager.	M
First_Manager_Number	String	The phone number of the main manager.	M, U
Second_Manager_Name	String	The name of the next manager.	
Second_Manager_Number	String	The phone number of the next manager.	U

Product			
Column Name	**Data Type**	**Description**	**Constraints**
Product_ID	String	Auto-incrementing, auto-generated ID to identify specific products. (e.g. P-0001)	PK, M, U
Client_ID	String	Foreign key linking the client to the product.	FK, M
Product_Name	String	The name of the product.	M, U
Product_Creation_Date	Date	The date of creation of the product.	M

Cost_Option			
Column Name	**Data**	**Description**	**Constraints**

	Type		
CO_ID	String	Auto-incrementing, auto-generated ID to identify specific cost options of products. (e.g. CO-0001)	PK, M, U
Product_ID	String	Foreign key linking the product to the cost option.	FK, M
Product_Name	String	The name of the cost option's product.	M
Units	Integer	The number of units of this cost option.	M
Printing	Integer	The integral value of the printing costs.	M
Design	Integer	The integral value of the design costs.	M
Plate	Integer	The integral value of the plate costs.	M
Die_Cut_Mould	Integer	The integral value of the die cut mould costs.	M
Die_Cutting	Decimal	The decimal per-unit rate of the die cutting costs.	M
Lamination	Decimal	The decimal per-unit rate of the lamination costs.	
Emboss	Decimal	The decimal per-unit rate of the emboss costs.	
Hot_Stamping	Decimal	The decimal per-unit rate of the hot stamping costs.	
Gluing	Decimal	The decimal per-unit rate of the gluing costs.	
Packing	Integer	The integral value of the packing costs.	M
Transportation	Integer	The integral value of the transportation costs.	M
Packing_Material	Integer	The integral value of the packing material costs.	M
Unit_Cost	Decimal	The decimal value of the unit cost of this cost option. This field is automatically calculated using the fields above.	
Total_Cost	Decimal	The decimal value of the total cost of this	

		cost option. This field is also automatically calculated using the fields above.	

Supplier			
Column Name	**Data Type**	**Description**	**Constraints**
Supplier_ID	String	Auto-incrementing, auto-generated ID to identify specific suppliers. (e.g. S-0001)	PK, M, U
Supplier_Name	String	The name of the supplier.	M, U
Supplier_Type	String	The type of product/raw material provided by the supplier. (e.g. paper)	M
Email_Address	String	The contact email address of the supplier.	U
Office_Address	String	The address of the supplier's main office.	U
Office_Number	String	The phone number of the supplier's main office.	U
Manager_Name	String	The name of the main manager.	M
Manager_Number	String	The phone number of the main manager.	M, U

Quotation			
Column Name	**Data Type**	**Description**	**Constraints**
Quotation_ID	String	Auto-incrementing, auto-generated ID to identify specific quotations. (E.g. Q-0001)	PK, M, U
Quotation_Creation_Date	Date	The date of creation of this quotation.	M
Client_ID	String	Foreign key linking the client to the quotation.	FK, M
Product_ID	String	Foreign key linking the product to the quotation.	FK, M
CO1_ID	String	Foreign key linking the first cost option of the product to the quotation.	FK, M

CO2_ID	String	Foreign key linking the second cost option of the product to the quotation (if available).	FK
CO3_ID	String	Foreign key linking the third cost option of the product to the quotation (if available).	FK
Lead_Time	Integer	The estimated time of the production process in working days.	M
Tolerance_Of_Quantity	Integer	Approximation error of the final quantity.	M
Terms_Of_Payment	String	The agreed term of payment of the quotation.	M
Quotation_Validity_Period	Integer	The period for which the quotation is valid for in working days.	M

Delivery_Order			
Column Name	**Data Type**	**Description**	**Constraints**
DO_ID	String	Auto-incrementing, auto-generated ID to identify specific delivery orders. (E.g. DO-0001)	PK, M, U
Client_ID	String	Foreign key linking the client to the delivery order.	FK, M
Delivery_Address	String	The delivery address of the delivery order.	M
DO_Creation_Date	Date	The date of creation of this delivery order.	M
Delivery_Mode	String	The mode of delivery of the delivery order.	M
Term_Days	String	The time period for which the payment should be made in.	M
Salesman_Name	String	The salesman in charge of this delivery order.	M
Item_One_ID	String	Foreign key linking the cost option of a product to the delivery order - acting as the first item of the delivery order.	FK, M

Item_Two_ID	String	Foreign key linking the cost option of a product to the delivery order - acting as the second item of the delivery order. (If available)	FK
Item_Three_ID	String	Foreign key linking the cost option of a product to the delivery order - acting as the third item of the delivery order. (If available)	FK
Total_Packets	Integer	The integral value of the number of total packets delivered.	M

Invoice			
Column Name	**Data Type**	**Description**	**Constraints**
Invoice_ID	String	Auto-incrementing, auto-generated ID to identify specific invoices. (E.g. I-0001)	PK, M, U
DO_ID	String	Foreign key linking the DO to the invoice.	FK, M
Client_ID	String	Foreign key linking the client to the invoice.	FK, M
Delivery_Address	String	The delivery address of the invoice.	M
Invoice_Creation_Date	Date	The date of creation of the invoice.	M
Term_Days	String	The time period for which the payment should be made in.	M
Salesman_Name	String	The salesman in charge of this invoice.	M
Item_One_ID	String	Foreign key linking the cost option of a product to the invoice - acting as the first item of the invoice.	FK, M
Item_Two_ID	String	Foreign key linking the cost option of a product to the invoice - acting as the second item of the invoice.	FK
Item_Three_ID	String	Foreign key linking the cost option of a product to the invoice - acting as the third item of the invoice.	FK
Tariff_Code	String	The tariff code of the invoice.	M

Purchase_Order			
Column Name	**Data Type**	**Description**	**Constraints**
PO_ID	String	Auto-incrementing, auto-generated ID to identify specific purchase orders. (E.g. PO-0001)	PK, M, U
Supplier_ID	String	Foreign key linking the supplier to the purchase order.	FK, M
PO_Creation_Date	Date	The date of creation of this purchase order.	M
Delivery_Address	String	The delivery address of this purchase order.	M
Description	String	A brief description of the product purchased.	M
Quantity	Integer	The integral value of the quantity of the product purchased.	M
Unit_Price	Decimal	The unit price of the product purchased.	M
Total_Price	Decimal	Automatically calculated field of total price using quantity * unit_price.	

Views

The views will be created using queries that extract the required data from the different tables above. The fields of the current tables above have somewhat vague information about the different entities (e.g. in that the cost options have zero details about themselves in the Quotations table), and thus there is a need to merge the different columns from different tables together to create records that tell us the full information. Foreign keys will be used to merge the correct data together. The data from these views will be used to fill out the fields of the automatically generated PDF documentations.

Complete_Quotation	
Column Name	**Description**
Quotation_ID	These are basic quotation fields from the original quotation table.

Quotation_Creation_Date	
Client_ID	
Product_ID	
CO1_ID	
CO2_ID	
CO3_ID	
Lead_Time	
Tolerance_Of_Quantity	
Terms_Of_Payment	
Quotation_Validity_Period	
Client_Name	Merged from the client table, using the corresponding Client_ID from above, these are the basic client details for which the quotation is for.
Office_Address	
First_Manager_Name	
First_Manager_Number	
Second_Manager_Name	
Second_Manager_Number	
Product_Name	Merged from the product table, using the corresponding Product_ID above, this single field is for the name of the product that the quotation is about.
CO1_Units	Merged from the cost options table, using the corresponding cost option IDs that CO1_ID, CO2_ID and CO3_ID above are referring to - these are the columns for the units and unit costs of the cost options.
CO1_Unit_Cost	
CO2_Units	
CO2_Unit_Cost	
CO3_Units	
CO3_Unit_Cost	

Complete_Delivery_Order

Column Name	Description
DO_ID	These are basic delivery order fields from the original delivery order table.
Client_ID	
Delivery_Address	
DO_Creation_Date	
Delivery_Mode	
Term_Days	
Salesman_Name	
Item_One_ID	
Item_Two_ID	
Item_Three_ID	
Total_Packets	
Client_Name	Merged from the client table, using the corresponding Client_ID above, this field is for the name of the client for which this delivery order is for.
Item_One_Name	Merged from the cost options table, using the corresponding cost option IDs that the Item_…_ID's above are referring to. These fields are about the corresponding product names and units of Item_One_ID, Item_Two_ID and Item_Three_ID.
Item_One_Units	
Item_Two_Name	
Item_Two_Units	
Item_Three_Name	
Item_Three_Units	

Complete_Invoice	
Column Name	**Description**
Invoice_ID	These are the basic invoice fields from the original invoice table.
DO_ID	
Client_ID	

Delivery_Address	
Invoice_Creation_Date	
Term_Days	
Salesman_Name	
Item_One_ID	
Item_Two_ID	
Item_Three_ID	
Tariff_Code	
Client_Name	Merged from the client table, using the corresponding Client_ID above - these fields are about some basic details of the client for which this invoice is for.
Office_Address	
Office_Number	
Email_Address	
Item_One_Name	Merged from the cost options table, using the corresponding cost option IDs that Item_…_ID's above are referring to. These fields are about the corresponding product names, units, unit costs and total costs of Item_One_ID, Item_Two_ID and Item_Three_ID.
Item_One_Units	
Item_One_Unit_Cost	
Item_One_Total_Cost	
Item_Two_Name	
Item_Two_Units	
Item_Two_Unit_Cost	
Item_Two_Total_Cost	
Item_Three_Name	
Item_Three_Units	
Item_Three_Unit_Cost	
Item_Three_Total_Cost	
Sales_Tax_Amount	This is a calculated field created by adding the total costs of all the items together then multiplying by 0.1, to get 10% of the total, which is the sales tax amount.

Total_Price	This is a calculated field created by adding the total costs of all the items together, and the sales tax amount, resulting in the total price of the invoice.

Complete_Purchase_Order	
Column Name	**Description**
PO_ID	These are the basic purchase order fields from the original purchase order table.
Supplier_ID	
PO_Creation_Date	
Delivery_Address	
Description	
Quantity	
Unit_Price	
Total_Price	
Supplier_Type	Merged from the supplier table, using the corresponding supplier ID above, these are some basic details about the supplier that this purchase order is made to.
Supplier_Name	
Office_Address	
Office_Number	
Manager_Name	
Manager_Number	

Database-layer Data Validation

Data validation is about checking the accuracy and quality of data entries. It typically occurs when data is added, updated or deleted. I will be implementing data validation in the database-layer because data entries by the user will occur here, hence there is a need to check its correctness to prevent any complicated and unwanted database errors. It is important to note that we already have the constraints and data types planned from before, that also act as tools for data validation. However, the data validation rules below are additional unique validations that help to greater ensure accurate data entry.

Client	
Column Name	**Validation Type**
Email_Address	Format check - (e.g. user**@gmail.com**) helps to make sure that a valid email address is entered in this field.
Office_Number	Format check - (e.g. 012-3456 7890) helps to ensure that the phone numbers are written in a consistent and neat format.
First_Manager_Number	Format check - (e.g. 012-3456 7890) helps to ensure that the phone numbers are written in a consistent and neat format.
Second_Manager_Number	Format check - (e.g. 012-3456 7890) helps to ensure that the phone numbers are written in a consistent and neat format.

Cost_Option	
Column Name	**Validation Type**
Units, Printing, Design, Plate, Die_Cut_Mould	Range check - value must be greater than 0 (cannot be negative or less than 0) but there is not an upper limit set on this range, as an underestimation of these values is undesirable.
die_cutting, lamination, emboss, hot_stamping, gluing,	Range check - value must be in between or equal to 0 and 1 (cannot be less than 0 or greater than 1).
packing, transportation, packing_material	Range check - value must be greater than 0 (cannot be negative or less than 0) but there is not an upper limit set on this range, as an underestimation of these values is undesirable.

Supplier	
Column Name	**Validation Type**
Supplier_Type	Code check - values must be either “printing”, “paper” or “finishing.” User must not input any other value in this field.
Email_Address	Format check - (e.g. user**@gmail.com**) helps to make sure that a valid email address is entered in this field.
Office_Number	Format check - (e.g. 012-3456 7890) helps to ensure that the phone numbers are written in a consistent and neat format.
Manager_Number	Format check - (e.g. 012-3456 7890) helps to ensure that the phone numbers are written in a consistent and neat format.

Quotation	
Column Name	**Validation Type**
Lead_Time	Range check - value must be in between or equal to 1 and 365 (cannot be less than 1 or greater than 365).
Tolerance_Of_Quantity	Range check - value must be in between or equal to 1 and 25 (cannot be less than 1 or greater than 25).
Quotation_Validity_Period	Range check - value must be in between or equal to 1 and 30 (cannot be less than 1 or greater than 30).

Delivery_Order	
Column Name	**Validation Type**
Total_Packets	Range check - value must be in between or equal to 1 and 1000 (cannot be less than 1 or greater than 1000).

Invoice	
Column Name	**Validation Type**
Tariff_Code	Format check - values must be in the format “1234-56-7890” as the numbers in tariff codes are separated.

Application-layer Data Validation

As I also use forms in the web-based application connected to our database, there is also the need for data validation for the data entries in those forms. Below is a summary of all the special data validations implemented in the forms of the web-based application.

Database Viewer	
Form	**Validation Type**
General Filter Settings	• Presence check: the user must enter a value upon clicking the search button.
Date Filter Settings	• Presence check: the user must enter a value upon clicking the search button. • Format check: the user must enter a date value in the form of YYYY-MM-DD.

Documentation Generator	
Form	**Validation Type**
Document ID Input Form	• Presence check: the user must enter a value upon clicking the generate button. • Format check: the user must enter a document ID value in the form of “Q-0001” or “I-0001” or “DO-0001” or “PO-0001” depending on the documentation selected.

Test Plan

The below table highlights all the tests that will be carried out to ensure that the final product meets each point in the success criteria.

Success Criteria	Test
To have all the required tables set up beforehand.	Check the tables created against the data dictionary.
To have a form for each table for the user to easily input the desired records into the database.	Use the form editor to input appropriate sample data into the tables.
To have preset views that merge together the desired tables that are needed to assemble the required fields and records together for the PDF document generation.	Check the views in the database against the views in this Criterion B document.
To be able to filter records from tables by inputting the value to be searched for in the table.	Input different values into the filter forms of different tables and check if the correct records appear.
To be able to automatically generate the selected documentation by inputting the appropriate document ID.	Input different document IDs for each of the four types of documents and check if the right data is presented in the generated documentation.
To have a responsive website UI that scales with the window size.	Play around with the window size and see if the page elements appropriately change in size as well.
To make sure the records and data in the generated PDF fit within the margins and borders (especially with longer record values),	Generate different documentations and see if any of the data overlaps with other PDF elements or margins and borders.

and do not overlap with any of the other elements of the PDF.	
Constraints to be set on database fields to minimise data entry errors and prevent database errors from occurring.	Test each constraint and see if they work according to their purpose.
Data validation for both the forms on the web-based application and forms in the MySQL database.	Violate all the data validation rules and see if the correct error messages appear.