# Design Process Report

This is the final ERD for my database

Diagram

Description automatically generated

During this assessment I will be doing a few things in order to get the data to a usable state and then I’ll be using that data to create a SQL database

## Data Cleansing:

The first step into making this data usable is to go through a process of data cleansing, this must be done because the data we previously got was entered manually by staff, this could lead to multiple errors, whether it is misspelling or just forgetting to enter some data. When I was going about cleaning the data, I followed a 10-step process that highlighted the key things to remove and change in the data, this was quite easy to do as the 10-steps provided where explained well and in detail

## Entity and Attribute selection:

When I started with the design of my database I looked at the data and I thought I saw the perfect entities and attributes to use, the entities that I used at first were, Patient, Surgeon, Referrer, Referral, and the FSA date, but as I was going along with the design and doing my normalisation I soon realised that the FSA date could just be an attribute of one of the entities as it only really carried one piece of information a single date, so in the end the entities that I selected were Patient, Surgeon, Referrer and Referral.

The attributes I used for these entities were the first and last names of the surgeon for the surgeon table this was the same idea for the patient table and the referrer table, for each of my entities I created a automatically generated ID this ID field was used as the primary keys for all the tables, I did this so that I can easily access the different tables and get all the data form the tables, for the surgeon table I also chose the department as an attribute I did this because you would want to get a surgeon in a specific department to deal with the patients need for example having a patient that need to undergo heart surgery, you would want the surgeon that you are assigning to be part of the department that specializes in hear surgery, in the patient table I used attributes like NHI, DOB, gender, the NHI is a unique identifier that is assigned to every person who uses health and disability support services in New Zealand so this is helpful to know what services people have used, I did not use this as my primary key because it would’ve violated the NZ privacy act and that is why each patient gets a unique ID assigned to them, The DOB is used because the staff wanted to have a query to see the age of the person when they were first referred so we could use this DOB information to calculate and display the age of the patient, I used gender so that we have more information to identify patients with.

## Choice of Keys:

When making this database I had to decide on what my primary and foreign keys would be,

For the Surgeon table I had only one key and that was the surgeonID this was the primary key for this table, for the patient table I had to create a new value for patientID because NHI was not able to be used as a primary key, for the referrer I used referrerID as the primary key, when I was looking at these tables I saw that there could be one common table to connect them all and this table was the referral table for this table I created a primary key of referralID to keep track of all the referrals happening, I then created 3 foreign keys these were the surgeonID, patientID, referrerID, using the ID’s as my foreign keys I could easily get any information from these tables by doing a single ID lookup instead of looking at multiple different foreign keys.

## Connectivity/Relationships:

On my first iteration of the ERD I had 4 one to many relationships between the FSA, Surgeon, Referrer, Referral to Referrer, and Referral to Patient

Diagram

Description automatically generated

I came to the conclusion of using this because before I added the Referral table everything was a many to many relationship so the Referral was the bridging table that I used to connect everything,

However in my second iteration I looked at the table and saw some data that doesn’t have to be in its own table eg the FSA table

Diagram

Description automatically generated

In this finalised ERD I have the same concept as before but instead of Surgeon talking to Referrer and then Referrer talking to Referral to get patient data, I now only need to use the Referral and Referral can contact any table and get information about it.

## Use of composite/bridging entities:

In my ERD when I connected them all they were all many to many relationships and we could not have this, to fix this I created a bridging table called Referral that was in charge of keeping all the primary keys of the other tables as foreign keys, this was a simple way of making sure that I reduce the number of steps it takes to get data from wherever you want to get it from, for example in my first ERD if I wanted to get information on the surgeon as the patient I would have to go to Referral to get the ReferrerID then when I have the ReferrerID I then look for the SurgeonID then only can I get to the surgeon, whereas in my second and finalised ERD all I have to do is look at the Referral table then follow it to the Surgeon reducing the steps from 4 to 2

## Extent of normalisation:

For the normalisation I created 4 tables to look at the data separately then from that I went through the steps of creating my 3 Normal form tables(1NF,2NF,3NF) in these I went through the process of getting rid of unnecessary datapoints whether it was a duplicate or offered no use in defining the tables. This is shown by looking at the NF tables in order

## Interesting aspects:

When starting my ERD’s I used some entities that later showed not to be of much use these were the FSA table and the department table that I showed in my first data dictionary, when building out my ERD’s these stuck out like a sore thumb they were not useful at identifying attributes or were just redundant and didn’t need to be in a separate table so in the end I either got rid of them or made the attributes of other tables that made more sense, another point of interest was figuring out my cardinality or relationships, when connecting the patient and surgeon tables we got a many to many relationship so I had to find the best suited bridging table to get rid of this relationship and make it a one to many, I settled on a Referral table as the bridging entity I used this because then following the logic of how a hospital would refer a patient to a surgeon or vice versa this just fit the description so I used this Referral table to get rid of all the many to many relationships in the ERD

## Version Control:

For my version control I used Github to keep track of all the things added and changed as I was working on this project, to view the commits you can go to my github profile <https://github.com/KeaganErasmus?tab=repositories> and look into the BCDE103 repository this will detail all the commits made and what was changed with each commit