**Planning Inspectorate – Documentation**

Overall Design

Diagram

Description automatically generated  
*Diagram showing sections of the web application and transfer of data between them*

API Layer:

This layer is responsible for handling incoming API connections and hosting the endpoints, no processing occurs on this layer. When a request is received this layer will call the correct processing function on the service layer. The service layer will return the processed data to the API layer which can use it as a response to the request.

Service Layer:

This layer is responsible for processing and collating, this is the actual backend where any algorithms or procedures on data occur. It cannot interact with the API or database but can make calls to the data layer and is called by the API layer.

Data Layer:  
This layer handles database interaction and exposes the database to the Service layer.

Frontend design

Website Navigation

Diagram

Description automatically generated  
*Flowchart showing how to user can interact with the website*

Initial page mockups

Diagram

Description automatically generated  
*Home page*

Text, letter

Description automatically generated  
*Create consultation list*

A picture containing graphical user interface

Description automatically generated  
*View consultation list*  
Text

Description automatically generated with low confidence *Manage database*

Diagram

Description automatically generated  
*Edit/create record*

Diagram

Description automatically generated *Confirm/reject changes pop-up*

Graphical user interface

Description automatically generated  
*View updates done via a one time link*

API Requests

Table

Description automatically generated

Database Design

The meeting with the client on 19/11/21 allowed us to take away a few points about the 'master sheet':

1. Each contact could be a related to several departments and organisations. (many to many relationships)
2. Sometimes contact information is put into the 'notes' column rather than the relevant columns (e.g. email)
3. Contact details can occur in the relevant columns as well as in the notes
4. For organisations such as the environment agency and Historic England there are people you need to contact every time as well as contacts based on the location
5. Not all locations identified by the GIS software need to be consulted. This is based on the information stored in the 'tests' column and 'notes'
6. 'Special emails' are addressed to specific people. 'Generic emails' are when they send an email to an organisation wide inbox (e.g. 'info@[org].gov.uk')

Based on this we need join tables between the contacts and departments and columns containing the tests and notes must be kept. Also the lack or regular structure in where data goes would make it challenging to create a script that moves data from the 'master sheet' to our database.

Entity Relationship Diagram

Diagram

Description automatically generated

Tables

|  |  |
| --- | --- |
| Organisation | |
| Organisation\_id\* | Integer |
| name | Integer |
| special | Integer |

|  |  |
| --- | --- |
| Department | |
| Department\_id\* | Integer |
| Organisation\_id | Integer |
| name | varchar |
| test | text |
| notes | text |

|  |  |
| --- | --- |
| Person | |
| Person\_id\* | Integer |
| name | Varchar |
| AddressBlock | Text |
| email | Varchar |
| method | Varchar |

|  |  |
| --- | --- |
| Contact | |
| Contact\_id\* | Integer |
| Department\_id | Integer |
| Person\_id | Integer |
| Otp | Varchar |

|  |  |
| --- | --- |
| SpecialContact | |
| SpecialContact\_id\* | Integer |
| Person\_id | Integer |
| Organisation\_id | Integer |