



Design Specification

ReadySetResource.com

Contents

5.1 - Data Flow Diagrams 4

5.2 - System Architecture 4

5.3 - Use Case Descriptions 4

5.4 - Use Case Diagrams 5

www. 5

get. 6

legal. 7

contact. 8

help. 9

admin. 10

dev. 11

sa. 12

head. 13

im. 14

5.5 - Activity Diagrams 15

5.6 - Class Diagrams 15

5.7 - Sequence and Communication Diagrams 15

5.8 - State Diagrams 15

5.9 - Normalisation 16

5.10 - Data Model 17

5.11 - Form Layouts 17

5.12 - Object-Relational Mapping 17

5.13 - Class Definition Document 17

5.14 - Data Dictionary 17

5.15 - Conceptual Entity Relationship Diagrams 18

Key 18

Conceptual ERD 20

5.16 - Physical Entity Relationship Diagrams 22

Key 22

Physical ERD 23

5.17 - Screen Layout 25

5.18 - Validation Control Forms 25

5.19 - Event-Handling Forms 25

5.20 - Identifier Lists 25

5.21 - Bibliography 25

5.22 - Software Requirements Spec. Work Log 25

# 5.1 - Data Flow Diagrams

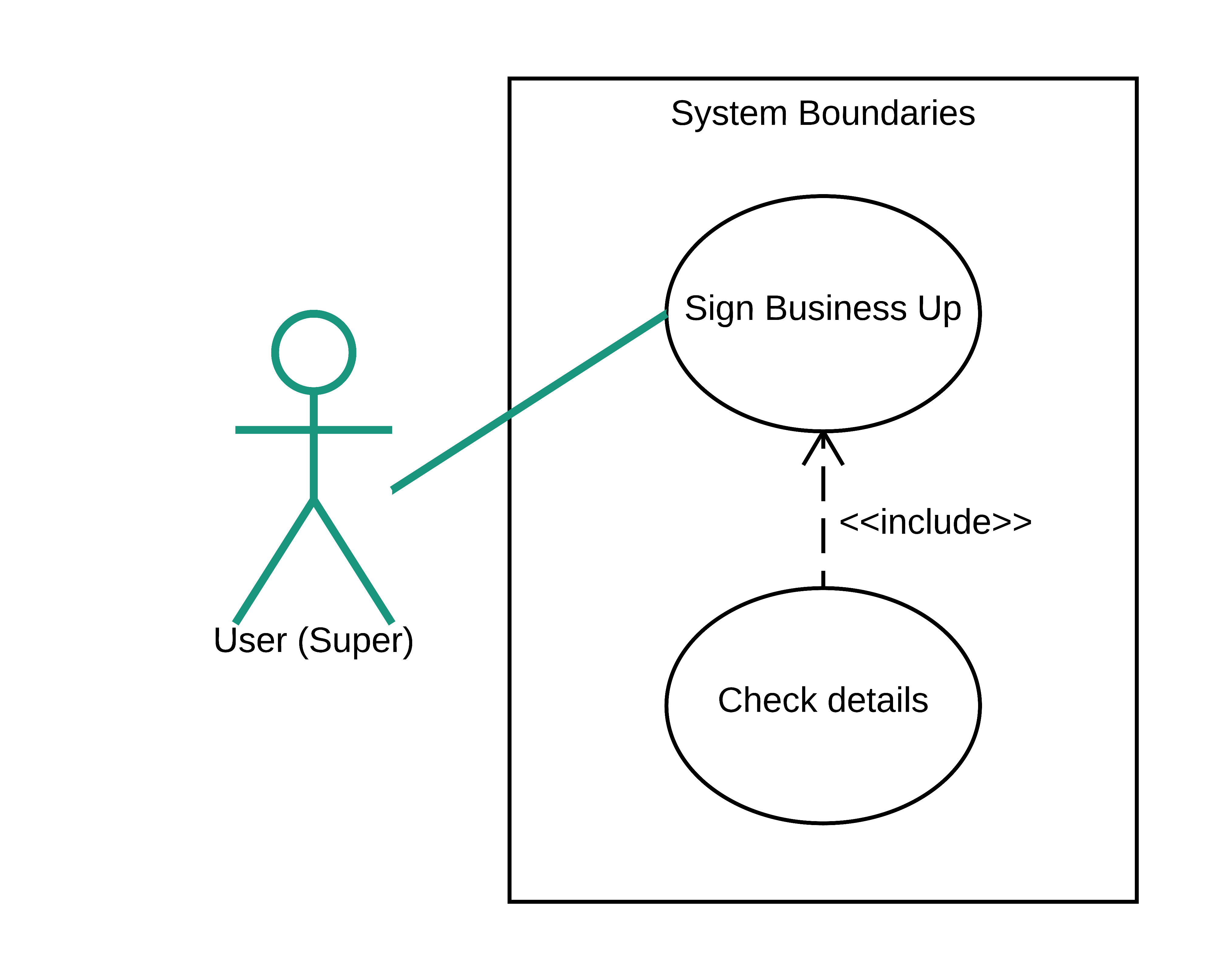
# 5.2 - System Architecture

# 5.3 - Use Case Descriptions

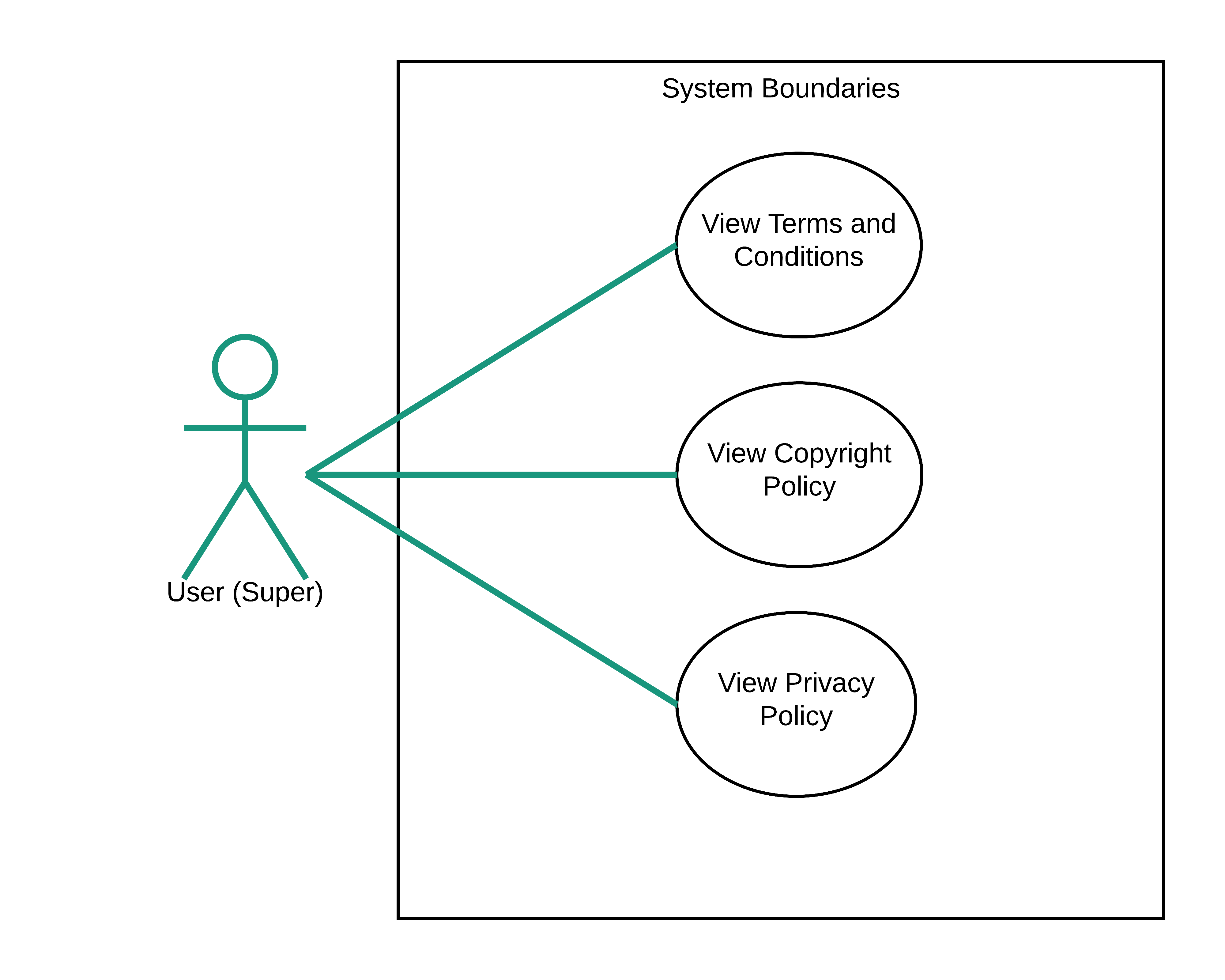
# 5.4 - Use Case Diagrams

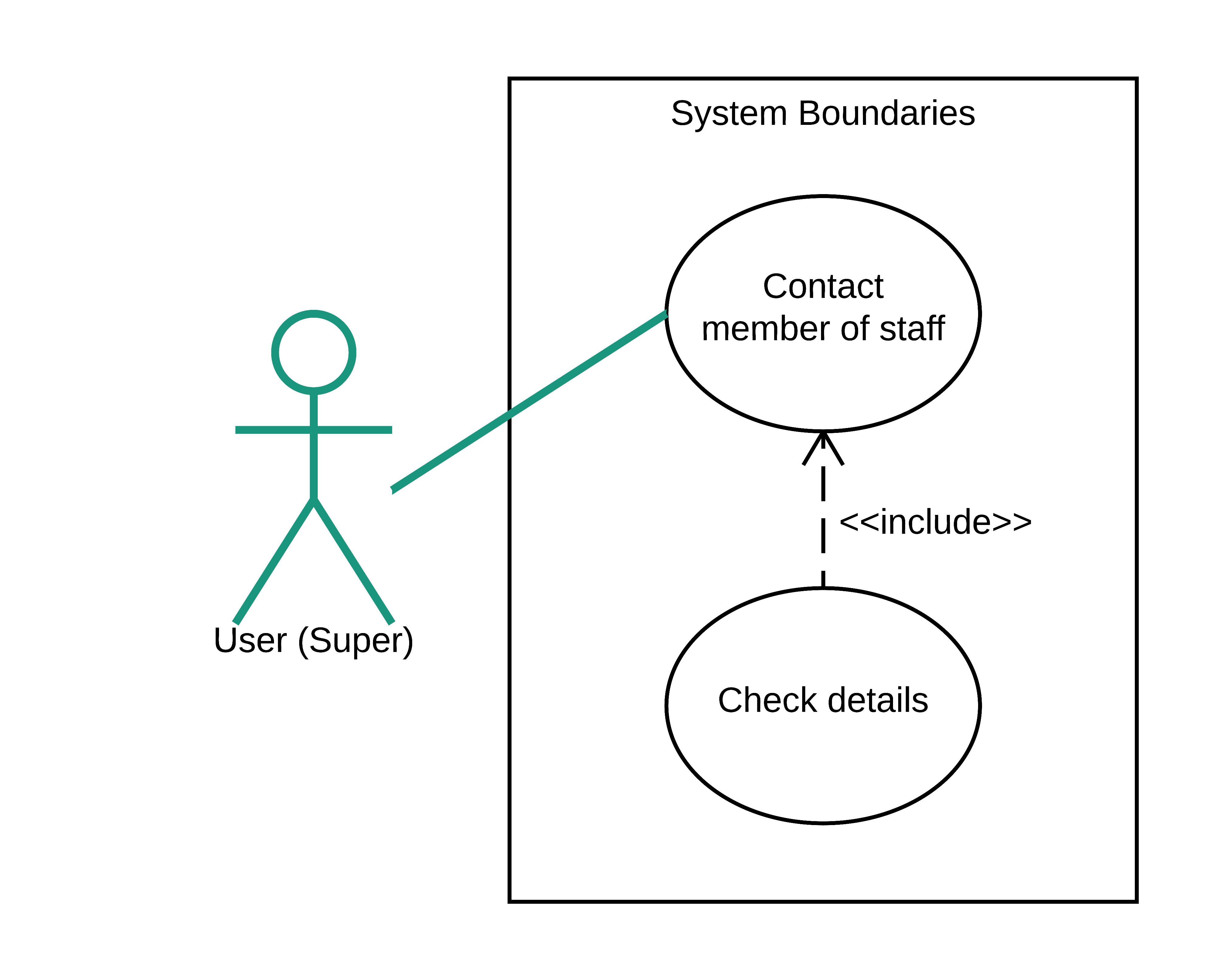
www.

get.



legal.



contact.

help.

admin.

dev.

sa.

head.

im.

# 5.5 - Activity Diagrams

# 5.6 - Class Diagrams

# 5.7 - Sequence and Communication Diagrams

# 5.8 - State Diagrams

# 5.9 - Normalisation

| UNF | 1NF | 2NF | 3NF |
| --- | --- | --- | --- |
| Order Id | Order Id | Order Id | Order Id |
| Order Date | Order Date | Order Date | Order Date |
| Staff Name | Staff Name | Staff Name | Staff Id\* |
| Staff Id | Staff Id | Staff Id | Customer Id\* |
| Customer Id | Customer Id | Customer Id |  |
| Customer Name | Customer Name | Customer Name | Staff Id |
| Street | Street | Street | Staff Name |
| Town | Town | Town |  |
| Postcode | Postcode | Postcode | Customer Id |
| Customer Phone Number | Customer Phone Number | Customer Phone Number | Customer Name |
| Product Id |  |  | Street |
| Product Name | Order Id | Order Id | Town |
| Quantity | Product Id | Product Id | Postcode |
| Product Price | Product Name | Order Total | Customer Phone Number |
| Order Total | Quantity |  |  |
|  | Product Price | Product Id | Product Id |
|  | Order Total | Product Name | Product Name |
|  |  | Quantity | Quantity |
|  |  | Product Price | Product Price |
|  |  |  |  |

Note: I decided to remove Order Total as it is derived data.

# 5.10 - Data Model

# 5.11 - Form Layouts

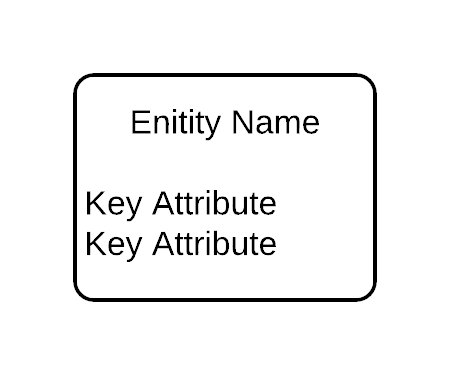
# 5.12 - Object-Relational Mapping

# 5.13 - Class Definition Document

# 5.14 - Data Dictionary

# 5.15 - Conceptual Entity Relationship Diagrams

Key

This is an entity. It has a name, key section and attributes.

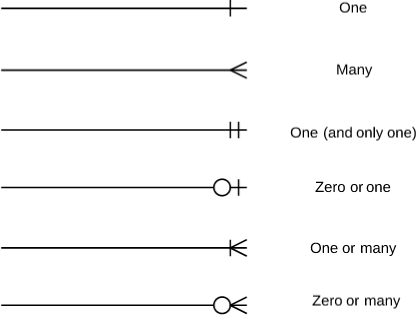
Key Section: The key section can hold a:

* + Primary Key (#)
  + Mandatory Field (\*)
  + Optional Field (o)
  + Candidate Unique Identifier (#1)
  + Part of a Candidate UID (#1-1), (#1-2), etc…

Attributes: The name of the attribute that will be stored in that entity.

Two entities may have one or more relationships. This is shown as a line from one entity to the other. Relationships between entities have cardinality and ordinality or optionality:

* + - Cardinality: “Cardinality refers to the maximum number of times an instance in one entity can relate to instances of another entity.” (A - Lucidchart, 2017)
    - Ordinality: “Ordinality, on the other hand, is the minimum number of times an instance in one entity can be associated with an instance in the related entity.” (B - Lucidchart, 2017)
    - Optionality: Optionality, like ordinality, refers to whether an instance of an entity needs to have another instance of an entity related to it. Ordinality and optionality both mean the same thing but are diagrammed differently. I have decided to use ordinality to diagram my Entity Relationship Diagram (ERD).



Cardinality and Ordinality

(C - Lucidchart, 2017)

One

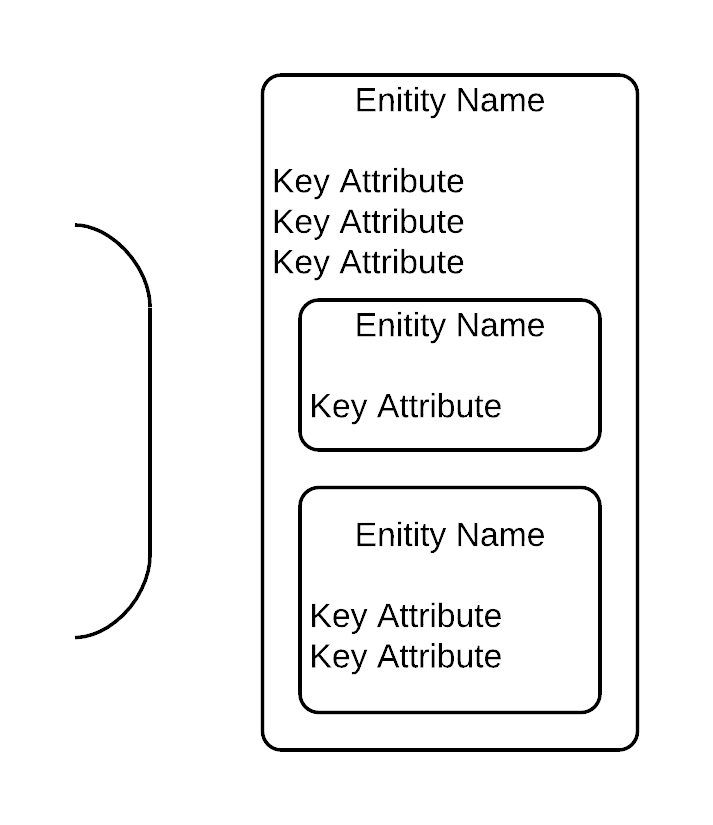
Many

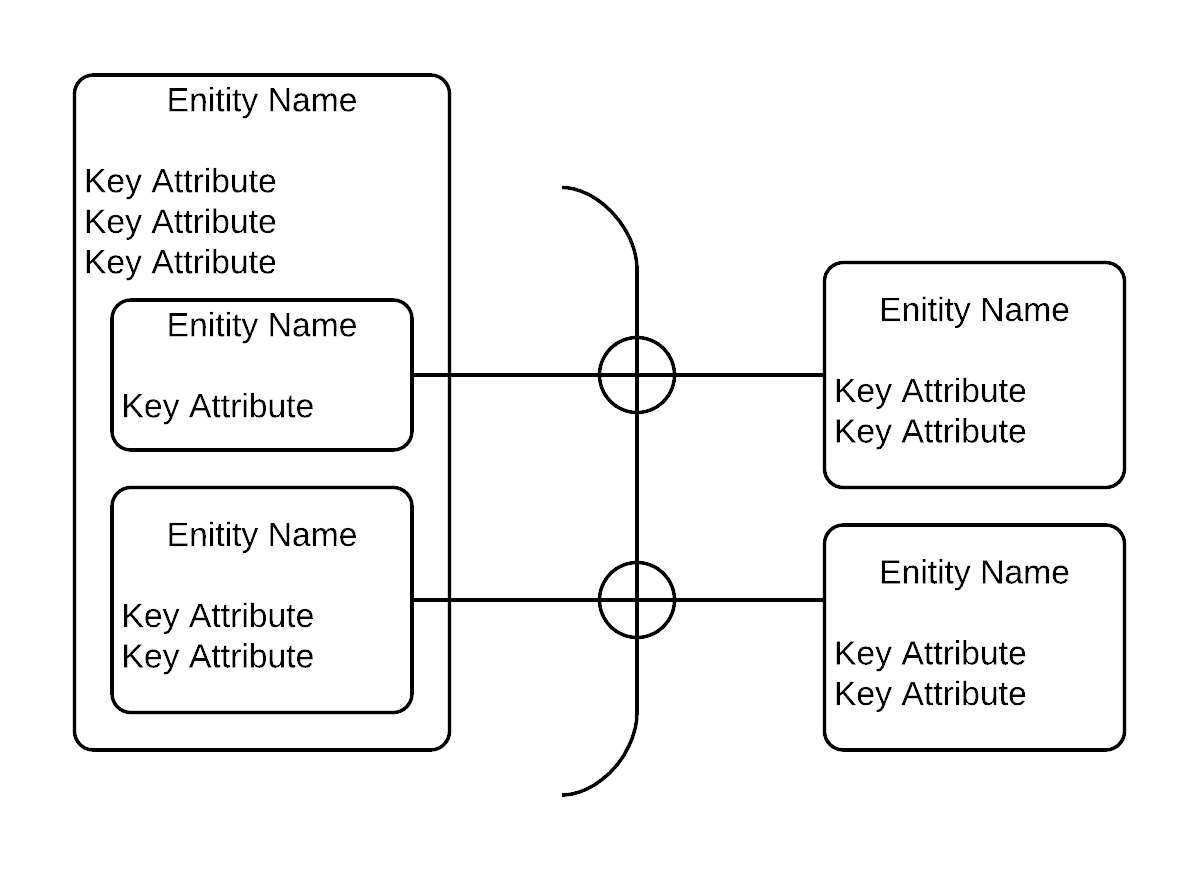
One (and only one)

Zero or one

One or many

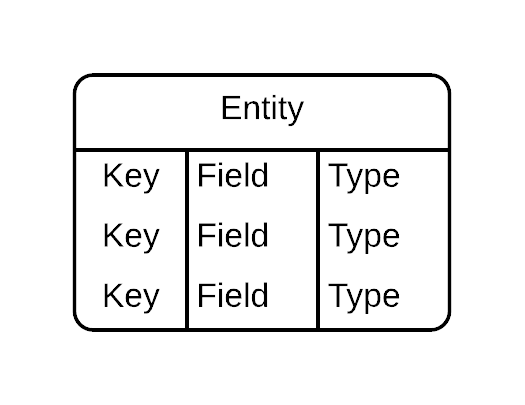
Zero to many

An entity is able to be a subtype or a supertype if required. If two or more objects have similar characteristics then you could make a supertype from extracting all the characteristics which are the same and then creating subtypes from the remaining differences of the objects. Think of a supertype as a blueprint of the basic object that you are trying to create and then a subtype as the alterations and additional features you will add. In conceptual modelling a supertype wraps around its subtypes as shown on the left.

Arcs are diagrammed in a conceptual model to show an Exclusive OR relationship. This means that a decision needs to be made for each instance of the object. Depending on a certain reason the entity can either have one relationship with a certain entity or another. It must decide one or the other. It cannot have both or neither.

Conceptual ERD

# 5.16 - Physical Entity Relationship Diagrams

Key

This is an entity. It has an entity name, keys, fields and types.

Keys: Keys can be a Primary Key (PK) or a Foreign Key (FK).

Fields: The name of the field that will be stored in that entity.

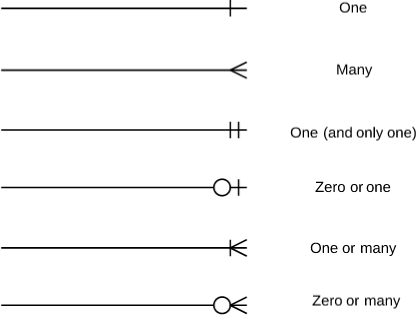
Type: The data type that the field will be (e.g. VarChar(16), Date, Char, Float etc…)

Two entities may have one or more relationships. This is shown as a line from one entity to the other. Relationships between entities have cardinality and ordinality or optionality.

Cardinality: “Cardinality refers to the maximum number of times an instance in one entity can relate to instances of another entity.” (A - Lucidchart, 2017)

Ordinality: “Ordinality, on the other hand, is the minimum number of times an instance in one entity can be associated with an instance in the related entity.” (B - Lucidchart, 2017)

Optionality: Optionality, like ordinality, refers to whether an instance of an entity needs to have another instance of an entity related to it. Ordinality and optionality both mean the same thing but are diagrammed differently. I have decided to use ordinality to diagram my Entity Relationship Diagram (ERD).



Cardinality and Ordinality

(C - Lucidchart, 2017)

Physical ERD

# 5.17 - Screen Layout

# 5.18 - Validation Control Forms

# 5.19 - Event-Handling Forms

# 5.20 - Identifier Lists

# 5.21 - Bibliography

A - Lucidchart (2017) ERD Diagram Symbols and Notation [online]. Available from the World Wide Web: <https://www.lucidchart.com/pages/ER-diagram-symbols-and-meaning> [accessed 25 November, 2017]

B - Lucidchart (2017) ERD Diagram Symbols and Notation [online]. Available from the World Wide Web: <https://www.lucidchart.com/pages/ER-diagram-symbols-and-meaning> [accessed 25 November, 2017]

C - Lucidchart (2017) ERD Diagram Symbols and Notation [online]. Available from the World Wide Web: <https://www.lucidchart.com/pages/ER-diagram-symbols-and-meaning> [accessed 25 November, 2017]

# 5.22 - Software Requirements Spec. Work Log

| Entry | Description | Person | Date | Mins Spent |
| --- | --- | --- | --- | --- |
| 1 | Formatted the document | Aidan Marshall | 25/11/17 | 6 |
| 2 | Started the Conceptual ERD | Aidan Marshall | 14/11/17 | 68 |
| 3 | Continued the Conceptual ERD | Aidan Marshall | 16/11/17 | 43 |
| 4 | Continued the Conceptual ERD | Aidan Marshall | 19/11/17 | 82 |
| 5 | Continued the Conceptual ERD | Aidan Marshall | 22/11/17 | 122 |
| 6 | Wrote the Conceptual ERD key | Aidan Marshall | 25/11/17 | 32 |
| 7 | Continued the Conceptual ERD | Aidan Marshall | 25/11/17 | 145 |
| 8 | Continued the Conceptual ERD | Aidan Marshall | 26/11/17 | 188 |
| 9 | Finished the Conceptual ERD | Aidan Marshall | 27/11/17 | 328 |
| 10 | Modified the conceptual ERD | Aidan Marshall | 28/11/17 | 120 |
| 11 |  |  |  |  |
| 12 |  |  |  |  |
| 1134 | | | | |