|  |
| --- |
| Project Documentation |
| INB201 Scalable Systems Development |
| An Emergency Management System for Queensland |

|  |
| --- |
| MELT  5/15/2013 |

Table of Contents

[Foreword 3](#_Toc356817034)

[Introduction 4](#_Toc356817035)

[Application Layer 5](#_Toc356817036)

[Database Layer 6](#_Toc356817037)

[Introduction to Database 6](#_Toc356817038)

[Database Tables 7](#_Toc356817039)

[Figure 2.1 EER model of database 7](#_Toc356817040)

[Services Relation 7](#_Toc356817041)

[Table 2.1 Services Relation 8](#_Toc356817042)

[Location Relation 9](#_Toc356817043)

[Table 2.2 Location Relation 9](#_Toc356817044)

[Table 2.3 Location Example 9](#_Toc356817045)

[Events Relation 10](#_Toc356817046)

[Table 2.4 Events Relation Explained 10](#_Toc356817047)

[Table 2.5 Events relation Example 11](#_Toc356817048)

[Personnel Relation 12](#_Toc356817049)

[Incidents Relation 13](#_Toc356817050)

[Event Type Relation 13](#_Toc356817051)

# Foreword

The state of Queensland is a diverse and varied state that has many unique factors that influence its functioning. It is the third most populated state in Australia and has the 2nd largest area. It is extends from the Tropics of the north to the more temperate areas of the south and is constantly aware of the severity of natural and man made disasters. Queensland has been tested many times over the past decade and it is public knowledge that the current system being using by government and local emergency management workers is not able to keep up with the rapid changing global environment information delivery and distribution.

The change in the ways in which information is retrieved and distributed is something that all Queenslanders understand has been rapid and fundamental to the growth of Queensland and Australia. The rapid accessibility to information and its distribution in more efficient and relevant means then those currently implemented is something that Queensland and indeed every government needs to be researching.

The application that team 5 has developed aims to suite after extensive consultation with government and other stakeholders. After this consultation the needs and expectations of Queenslanders where learnt and developed into this application.

This documentation aims to explain how to use team fives new application has been developed for consideration as a replacement state emergency management system for Queensland. It hopes to explain the applications strengths and weaknesses and the improvements it makes over current systems both domestically to Queensland and nationally.

Thank you

Team 5

# Introduction

This documentation will explain how the system team 5 has developed works and the ways in which the application will be used to deliver upon the user requirements. The system is built on the n-tier application layer architecture and the documentation is separated accordingly.

The documentation will be split into 3 sections:

* The Application Layer
* The Database Layer
* Help documentation

The application layer will document the interface design reasons, the form functions and the interface options.

The Database Layer documentation describes the rationale behind the tables in the database and the values in each column along with the constraints that apply to the tables and their relations.

The help documentation will be an external documentation listing to be sent along with the application package when the zip file is downloaded this documentation will give a brief overview as to how to use the help documentation as well as how to find it.

# Application Layer

# Database Layer

## Introduction to Database

The database layer of the application is where the real information is stored. The Database software implemented with this application is the MySQL community server. MySQL was chosen for various reasons but mainly:

- Large industry support base

- Many skilled administrators capable of working with the Database.

- MySQL server is continually being updated and security patches are readily available.

- Works with all major operating systems and can be run on all operating systems indifferent to the application.

- Can be obtained for free and industry licences are relatively cheap but not compulsory.

- Easily developed with and using a strong robust syntax and supports complete range of standard SQL syntax rules and options.

The database used with this application consists of several tables, these tables have been designed with data redundancy in mind and the database is normalised to industry standard level 3.

## Database Tables

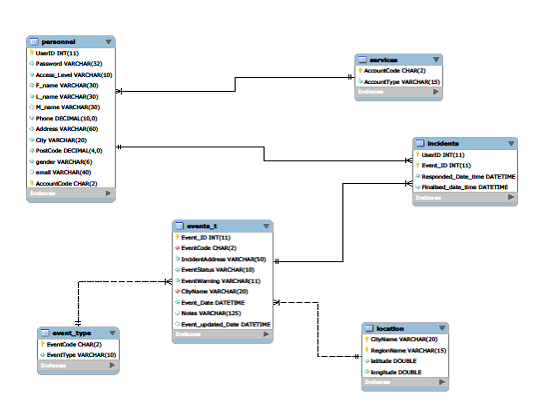
The database consists of 6 relations to help maximise data efficiency and reduce redundancy. The Database consists of the relation:

* Services
* Location
* Events\*
* Personnel
* Incidents
* event type

Note Events relation is called events\_t in database as event / events is a predefined function in SQL.

These relations all join together through a series of parent child relationships and foreign keys. The relations join together in the database to give the following model (figure2.1).

### Figure 2.1 EER model of database



Note: That the database varies slightly to the model shown but due to limitations in MySQL EER modelling this is the best representation that can be shown at this point.

## Services Relation

This Relation comes from the Normalisation process and also enforcing data integrity process. This relation is a parent to the personnel relation in the relationship involving a person’s Account Code.

The relation has two Attributes:

* AccountCode
* AccountType

The AccountCode attribute in the relation acts as a partial primary key with AccountType. AccountCode is referenced by the relation personnel with a foreign key. The Attribute was chosen as we defined it as a character with a limit size of 2. This enables the attribute to reduce the database size and help with processing speed while enabling most organisations that have had their name contracted/abbreviated to keep some meaning (ie Fire & Rescue is FR, PL is Police)(for a full list see table 2.1). Neither of the attributes are in the relation are allowed to be null.

The AccountType attribute is different to the AccountCode in two main ways. It has a Varchar length set to a max 15 characters as it gives the full name of the organisation to which the user has been assigned. This allows for the search queries to return values quicker as the larger the string the more processing power is used to return the values. The Varchar type has been chosen as it has the ability to keep the database smaller by making the rows with smaller value’s like fire only recording the four characters. In contrast if character 15 had been used the four characters would have been recorded plus the empty space after it filled with zeros or special characters to make the string up to a length of 15.

### Table 2.1 Services Relation

|  |  |
| --- | --- |
| AccountCode | AccountType |
| FR | Fire |
| AM | Ambulance |
| PL | Police |
| ML | Military |
| GV | Government |
| DM | Disaster Management |
| HR | Human Resources |
| SE | SES |

Note: In the relation the Disaster Management in the relation actually has the value of DISASTER\_MANGE. Human Resources value is also a different in the relation with Human\_Resource.

## Location Relation

This relation is designed to record the locations of the major towns and Cities in Queensland, along with the latitude and longitude co-ordinates of each of the cities/towns. We have also allocated each city into a region (This is to help if with cities and towns with the same name (ie Kawana waters is both on the South East and in Central)).

The four attributes of this relation are:

* City Name
* Region
* Latitude
* Longitude

Note: In the relation City Name is CityName.

In the relation the primary key is a composite key including CityName and Region. These attributes are both referenced as foreign keys as well with the events (events\_t) relation. This is so an incident can be tracked by a location and an area can have an incident attached to it. This enables for better and easier data distribution by allowing for easier information accessibility and keeping the events relation attribute city name column dependant on the location enables for data integrity. The values in these attributes have been restricted by defining a data type for each of them and each value have also been restricted to not allow null values. If an update or deletion rule is applied to an incident or event, a restrict rule has been placed on the foreign key so the removal of all information is not cascaded through.

The attributes have the allowable values and restriction found in Table 2.2. An example of the data in the relation is seen in table 2.3.

### Table 2.2 Location Relation

|  |  |  |
| --- | --- | --- |
| Attribute | Value | Restriction |
| City Name | Varchar | Max length 15 characters |
| Region | Varchar | Max length 20 characters |
| Longitude | Double | Must be in range 100-160 |
| Latitude | Double | Must be in range -10 to -40 |

Note: explanation of the location relation and its attributes.

### Table 2.3 Location Example

|  |  |  |  |
| --- | --- | --- | --- |
| CityName | Region | Longitude | Latitude |
| Brisbane | South East | -27.4710107 | 153.0234489 |

Note: underlined indicated primary key attribute/s.

## Events Relation

This relation holds the actual disaster events information. It references several other tables and is referenced by the incidents relation. In the database this table is called events\_t as event/events is a predefined function in the SQL query language. The relation contains several attributes:

* Event ID
* Incident Address
* Event Status
* Event Warning
* City Name
* Region Name
* Event Date
* Notes
* Event Updated date

Each of the attributes in this relation has constraints and a data type associated with it, this described in table 2.4.

### Table 2.4 Events Relation Explained

|  |  |  |  |
| --- | --- | --- | --- |
| Attribute | Data Type | Constraint | Other |
| Event\_ID | INT | Not allowed to be null,  Max length 9 numbers.  Is a primary key attribute. | This value is not put in or defined by the user. It has an Auto increment function assigned so every time a row is added the value of Event ID assigned increases by 1. Increment starts at 100000. |
| Incident Address | Varchar | Not allowed to be null,  Max length 50 characters. |  |
| Event Status | Varchar | Not allowed to be null  Max length 10.  Predefined check statements have been included. | Allowed values are:  normal, Safe, on-alert,  evacuate, emergency |
| Event Warning | Varchar | Not allowed to be null,  Max length 11 characters. Predefined check statement only allows certain values in this attribute. | Allowed values are: Definite,  High, Moderate, Low |
| City Name | Varchar | Not allowed to be null,  Max length 20. | References the Location table as a foreign key and is set to restrict if incident or event is deleted. |
| Region Name | Varchar | Not allowed to be null,  Max length 15. | References the Location table as a foreign key and is set to restrict if incident or event is deleted. |
| Event Date | Date Time | Not allowed to be null if a time is not provided the time at which the insert statement was run is the default time. |  |
| Notes | Varchar | Is allowed to be null.  Intended to be used to add information about an incident that cannot be contained in other fields. |  |
| Event Updated Date | Date Time | Is allowed to be null but by design the date time value when the initial query is run is used. |  |

Note: Attribute names differ slightly in database as spaces aren’t allowed convention of using a capital letter on the second or third word/s that has been used. Also date time is called datetime in SQL.

### Table 2.5 Events relation Example

|  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| EventID | Event  Code | Incident Address | Event Status | Event Warning | City  Name | Region  Name | Event  Date | Notes | Event Updated |
| 100001 | FE | 44 Wallaby way | Emergency | Moderate | Brisbane | South East | 2012-02-02-15:01:22 | 2 homes on fire spreading near oil refinery. | 2012-03-03 16:01:01 |

Note: By the end of event the notes warning filed will be updated to Low and the event status field to normal. When or as these values are changes the event updated time will be changed. The event date field will stay the same as it’s a record of the original add date.

## Personnel Relation

The Personnel relation keeps information about individual users of the systems and everyone that may need to use the emergency management service. It provides for identifying all users in the system and determining the access level of each of the users determined by looking at the information the row contains about each individual. This relation like events is both the parent and child of several relations. The attributes contained in this relation are:

* UserID
* Password
* Access level
* First Name
* Last Name
* Middle Name
* Phone
* Address
* City
* Post code
* Gender
* email
* Account code

## Incidents Relation

## Event Type Relation