

1. INTRODUCTION

This document specifies the software requirements specification document for the Direct 6 Software Solutions project sponsored by ITCC. This project is being undertaken by the Direct 6 Software Solution development team. The team is comprised of undergraduate students majoring in Computer Science at California State University, Sacramento. The team members are enrolled in a two-semester senior project course required for all undergraduate majors. Successful delivery of the desired software product will fulfill the senior project requirement for the student team members.

PROJECT SPONSOR

Name: Mr. Steven Archer

Title: Planning and Development

Organization name: Inter-Tribal Council of California (ITCC)

Contact information: Cell: (916) 320-0233

email: archers@cwnet.com

DIRECT 6 SOFTWARE SOLUTION DEVELOPMENT TEAM

Jared Arbaugh	Cell: (916) 771 – 3501	Email: arbaughj@gmail.com
Dave Diel	Cell: (916) 402 – 8640	Email: diel@surewest.net
Ben Mackin	Cell: (408) 646 – 3521	Email: bmackin@mac.com
Khae Saetern	Cell: (916) 208 – 1964	Email: khaes1421@hotmail.com
Benjamin Tovar	Cell: (209) 712 – 1417	Email: tovarb@gaia.ecs.csus.edu
Kou Vang	Cell: (916) 233 – 5051	Email: vangkou@gmail.com

1.1 Purpose

The purpose of this document is to introduce the reader to Direct 6 Software Requirement Specification document. Software Requirement Specification document will explain in detail the requirement specifications to be use by development group to complete the desired software product.

1.2 Scope

This document will cover detailed information about the management plan used for the project.

- The project overview is a statement of what our project is.
- The project organization is combination of flow charts and uses cases used to illustrate the organization of the project.
- The project management and control is a documentation of how Direct 6 will enforce the required scheduling for the completion of the project.
- The technical process will inform the reader of hardware, software, and programming language components used for the completion of the desired product.
- The activities, schedule, and budget will contain a WBS of all the documentation for this project.

1.3 Definitions, Acronyms and Abbreviations

Baseline - A baseline is a work product that has been formally reviewed and accepted by the involved parties. A baseline is changed only through formal configuration management procedures.

Milestone - A scheduled event used to measure progress.

MySQL - is an open source relational database management system (RDBMS) that uses Structured Query Language (SQL) for adding, accessing, and processing data in a database.

PHP - Is a programming language that allows web developers to create dynamic content that interacts with relational databases.

Project Deliverable - A work product that is delivered to the project sponsor, advisor, team or project manager. .

Task - The smallest unit of work subject to management accountability.

1.4 References

Document was prepared using the following references below.

Buckley, R. Guide to Preparing the Software System Proposal Document. Version 8.24.03

Buckley, R. Guide to Preparing the Software Requirements Specification Document. Version 7.3.03

Pressman, Roger S. 2001. *Software Engineering: A Practitioner's Approach*, McGraw Hill.

1.5 Overview of Contents of Document

- Section 2 of this document describes the types of users that will be using the service of the Elder Health Care System. Section two will also introduce the features which will be implemented in the final system. Any constraints, assumptions, and dependencies relating limiting the design of the final system will be noted here as well.
- Section 3 provides specific functional and non-functional requirements of the different components of the system.
- Section 4 contains a signature page to certify the approval of SRS by the sponsor and the members of Direct 6 Software Solutions team. Appendix A of this document contains a description of data store, data elements, and data structures. An Entity Relational Diagram (ERD) is given in appendix B.

2. GENERAL DISCRPTION

This section contains information that will make the rest of the documents easier to understand.

2.1 Product Perspective

The ITCC Database System being developed by Direct 6 will be completely independent and self-contained. This system will be used to collect the required data in order to provide Inter-Tribal Council of California with information. ITCC database system is being planned in two phases. Phase one will contain an online elder health care survey and a planning and development database for information on reservations. Phase two will hold information about violence clients and child care program.

This project proposes a system for use on a Personal Computer using a Visual Interface. The interface will be designed using the Programming Language PHP. Once the interface is designed, the data will be stored in Tables and Schemas which the team will create using MySQL. Major functions that will be provided by the system include: logging into the system, storing registrant information, elder healthcare questionnaires results, reports which generate statistical data using the data stored in the database.

2.2 Use Case Model of the system's features

This section describes the entities that shall interact with the ITCC system and the use cases for each entity involved.

Below are the actors that will interact with ITCC Database software system.

- Web User
- Simple User
- Database Editor
- Administrators

ITCC Software shall provide the following functions for each of the actors involved.

Web Users will be allowed to:

- Login
- Input Survey
- Logout

Simple Users will be allowed to:

- Login
- Input Survey
- Print Report
- Logout
- Generate Report
- Create Custom Report

Super User will be allowed to:

- Login
- Input Survey
- Logout
- Generate Report

- Print Report
- Create Custom Report
- Remove Entry
- Reset Table
- Create Data
- Read Data
- Update Data
- Delete Data

Administrators will be allowed to:

- Login
- Input Survey
- Print Report
- Create Custom Report
- Remove Entry
- Reset Table
- Remove user
- Update user
- Backup
- Logout
- Generate Report
- Create Data
- Read Data
- Update Data
- Delete Data
- Reset Passwords
- Create user
- Restore

2.3 User Characteristics

There are four user types for the ITCC Software system. Web users will only be able to login and logout of the system, and input surveys. He only needs the most basic of computer skills, such as operation of a mouse and the ability to fill out simple forms. Simple users will be able to generate and print reports, as well as create custom reports. Simple users shall be able to act out any use case intended for the Web user to act out. A simple user should know how to operate a web browser. Database editors will be able to create, update, and remove tables from the database. They will also be able to add, edit, or remove entries from individual tables, as well as reset any temporary tables. Database editors shall be able to act out any use case intended for the Simple user to act out. A database editor need not know any SQL queries, rather he only needs to know how to operate within a web browser. Administrators will be able to backup and restore the database. They will also be able to add, edit, or remove users from the system. Administrators shall be able to act out any use case intended for the Database editor to act out. Administrators should have a decent knowledge of how the system operates, and should have sufficient knowledge of how to navigate around a computer. The following use cases specify the interaction between different users and the ITCC Software system. A general use case model is described in Figure 2.1 below.

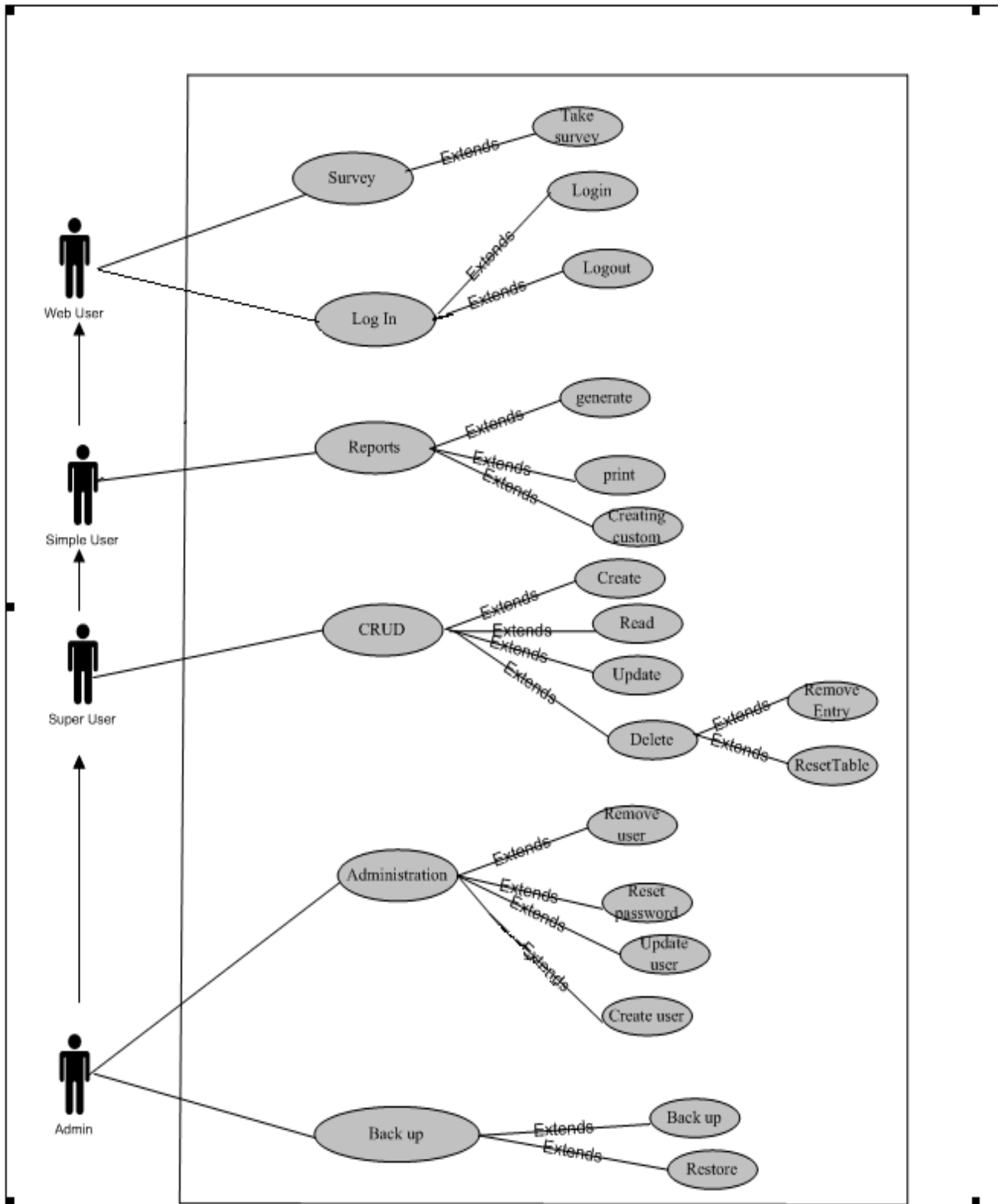


Figure 2.1 - Overview of User Characteristics, Use Case Diagram

2.4 General Constraints

The following is a list of general constraints that may affect the delivery of the final product.

The constraints are defined as follows:

- Elder Health Care system must run on a Windows NT server.
- Windows NT server must be able to support MySQL.
- Passwords will be at least 5 characters long.

2.5 Assumptions and Dependencies

Direct6 Software Solutions assumes that ITCC server will be compatible with MySQL and PHP.

3. SPECIFIC REQUIREMENTS

3.1 Use Case Specifications

This section documents each of the Use Cases identified in the system Use Case Model in section 2.

3.1.1 Take Survey

Users of the system can input surveys online.

Actor(s) Name: Web User, Simple User, Super User, or Administrator

Priority: High

Precondition(s): Authenticated user is logged into the system.

Postcondition(s): New survey information is stored in the database.

Primary Scenario:

1. User selects survey type.
2. System presents user with selected survey.
3. User enters the requested information and submits form.
4. If user provided valid input, system displays confirmation message indicating that survey has been successfully added.

Secondary Scenario:

1. If in step 3 of primary scenario, user enters invalid data, system displays error message and requires entering valid data.
2. At any time before submitting the form the user can cancel this operation and return to main menu or logout of the system.

3.1.2 Login

User logging into the system.

Actor(s) Name: Web user and above.

Priority: Low

Precondition(s): User must be a valid user in the system

Postcondition(s): Access to the data.

Primary Scenario:

1. User links to interface.
2. Interface gives a prompt to login with username and password and accept button.
3. User inputs username, password and hits the accept button.
4. Interface performs a check on the given inputs.
5. Inputs found and interface continues to appropriate sections.

Secondary Scenario:

1. Repeat steps 1-4 in primary Scenario.
2. Inputs not found and interface returns an error message.
3. User given a second chance but if the inputs are wrong again the program ends.

3.1.3 Logout

User logging out of the system

Actor(s) Name: Web user and above

Priority: Low

Precondition(s): User logged into the system.

Postcondition(s): No connection left opens.

Primary Scenario:

1. User finished with the system.
2. User hits link for logout.
3. System disconnects user.
4. System sends last message to user confirming logout.

Secondary Scenario:

1. User accidentally closes program without hitting the link for logout or user has not given input to the system.
2. System will wait for 5 minutes.
3. If no activity System will logout automatically

3.1.4 Generate Reports

Users of the system can generate reports based on data stored in the database.

Actor(s) Name: Simple User, Super User, or Administrator

Priority: High

Precondition(s): Authenticated user is logged into the system.

Postcondition(s): Report is generated and displayed on screen.

Primary Scenario:

1. User selects Reports from main menu.
2. System presents user with a list of options based on user rights.
3. User selects a report type, and then clicks on Generate Report.
4. System displays generated report.

Secondary Scenario:

1. At any time before clicking Generate Report the user can cancel this operation and return to main menu or logout of the system.

3.1.5 Print Reports

Users of the system can print reports once they have been generated.

Actor(s) Name: Simple User, Super User, or Administrator

Priority: High

Precondition(s): Authenticated user is logged into the system, and a report has been generated.

Postcondition(s): Current report is printed to the selected printer.

Primary Scenario:

1. User selects Print from the File menu.
2. System presents user with print setup screen.
3. User enters the requested information and chooses Print.
4. If user provided valid input, system begins printing the report.

Secondary Scenario:

1. If in step 3 of primary scenario, user enters invalid data, system displays error message and requires entering valid data.
2. At any time before submitting the form the user can cancel this operation and return to the screen with the report still showing.

3.1.6 Creating Custom Reports

Users of the system can create custom reports based on limited query strings.

Actor(s) Name: Simple User, Super User, or Administrator

Priority: High

Precondition(s): Authenticated user is logged into the system.

Postcondition(s): Report is generated and displayed on screen.

Primary Scenario:

1. User selects Reports from main menu.
2. System presents user with a list of options based on user rights.
3. User enters a query string, and selects Generate Custom Report.
4. If user provided valid input, system displays the custom report.

Secondary Scenario:

1. If in step 3 of primary scenario, user enters an invalid query string, system displays error message and requires entering of a valid query string.
2. At any time before submitting the form the user can cancel this operation and return to main menu or logout of the system.

3.1.7 Create Data

This use case enables the Administrator or Super User to create table and added to the database.

Actor(s) Name: Administrator or Super User

Priority: High

Preconditions(s): Authenticated user is an administrator or a Super User

Postcondition(s): A new table is created and is stored in the database

Primary Scenario:

1. Administrator or Super User selects Create Table
2. System request to enter name for table, field names, and data type for each field
3. Administrator or Super User entered requested information and submit form
4. If Administrator or Super User provided valid input, system displays confirmation message indicating that table has been successfully created

Secondary Scenario:

1. If in step 3 of primary scenario, Administrator or Super User enters invalid data, system displays error message and requires entering valid data.
2. At anytime before submitting the form the user can cancel this operation and return to main menu

3.1.8 Read Data

This use case enables the Administrator or Super to read the table

Actor(s) Name: Administrator or Super User

Priority: High

Preconditions(s): Authenticated user is an administrator or a super user

Postcondition(s): Read from a table in the database

Primary Scenario:

1. Administrator or Super User selects Read Table
2. System request to enter name for table, field names, and data type for each field
3. Administrator or Super User entered requested information and submit form
4. If Administrator or Super User provided valid input, system displays result read from table

Secondary Scenario:

1. If in step 3 of primary scenario, Administrator or Super User enters invalid data, system displays error message and requires entering valid data.

3.1.9 Update Data

This use case enables user to update the table.

Actor(s) Name: Administrator and Super User

Priority: High

Precondition(s): Authenticated user is an administrator or a super user

Postcondition(s): Updated table with new data

Primary Scenario:

1. Administrator or Super User selects Update Table
2. System request to enter the name of the table where data is to be update
3. Administrator or Super User entered requested information and submit form
4. If Administrator or Super User provided valid input, system displays form where new data can be added
5. Administrator or Super User entered requested information and submit form
6. System displays confirmation message indicating that table has been successfully updated with new data

Secondary Scenario:

1. If in step 3 of primary scenario, Administrator or Super User enters invalid data, system displays error message and requires entering valid data.
2. If in step 5 of primary scenario, Administrator or Super User enters invalid data, system displays error message and requires entering valid data.
3. At anytime before submitting the form the user can cancel this operation and return to main menu

3.1.10 Remove Entry

This use case enable user to remove or delete an entry from a table

Actor(s) Name: Super user, administrator

Priority: High

Preconditions: Must be logged on with at least Super user rights.

Postconditions: An entry in the table is removed from the database

Extended use case: Delete

Use case used: CRUD (create read update delete) --> Delete

Primary Scenario:

1. User select remove entry
2. System prompt user to enter entry to be delete
3. User enter data and submit form
4. System display confirmation regarding the deleted data

Secondary Scenario:

1. Any data that is no longer valid is deleted

Other requirements: None.

3.1.11 Reset Table

This use case enable user to reset the table

Actor(s) Name: Administrator and Super User

Priority: Medium

Precondition(s): Must be logged on with at least Super user rights.

Postcondition(s): Deleted all rows that were setup for reset.

Extended use case: Delete

Use case used: CRUD (create read update delete) --> Delete

Primary Scenario:

1. Clear out database for another survey in x amount of years

Secondary Scenario:

1. Reset the whole database (maybe give away computer w/hard drive.

Other requirements: None.

3.1.12 Remove User

Administrator can delete users from the system.

Actor(s) Name: Administrator

Priority: High

Precondition(s): Authenticated user is an administrator and is logged into the system.

Postcondition(s): User is removed from the system.

Primary Scenario:

1. Administrator selects Delete User.
2. System request to enter a user ID.
3. Administrator enters the requested information and submits form.
4. If Administrator provided valid input, the System displays that users info and presents the Administrator with a choice to delete the user.
5. Administrator submits form.
6. System displays confirmation message indicating that user has been successfully deleted.

Secondary Scenario:

1. If in step 3 of primary scenario, Administrator enters invalid data, system displays error message and requires entering valid data.
2. At any time before submitting the form the Administrator can cancel this operation and return to main menu or logout of the system.

3.1.13 Reset Password

Reset password for users.

Actor(s) Name: Administrator

Priority: High

Precondition(s): Administrator must verify user's Identification before reset password.

Postcondition(s): Password is reset. New password is stored in the database.

Primary Scenario:

1. User has changed original password.
2. Next time the user tries to log in, the system does not let him/her because of invalid password. The user forgot their new password.
3. User requested the Administrator to reset password.
4. Administrator verifies the users account with proper identification.
5. The password is reset.

Secondary Scenario:

1. If the user does not provide valid Identification.
2. The user will not reset password and then delete account.

3.1.14 Update User

Administrator can update user information.

Actor(s) Name: Administrator

Priority: High

Precondition(s): Authenticated user is an administrator and is logged into the system.

Post condition(s): New user information is stored in the database.

Primary Scenario:

1. Administrator selects Update User.
2. System request to enter a user ID.
3. Administrator enters the requested information and submits form.
4. System request to enter first name, enter middle initial, enter last name, enter user ID, enter password, enter the type of user to be changed to.
5. Administrator enters requested information and submits form.
6. If Administrator provided valid input, system displays confirmation message indicating that user has been successfully added.

Secondary Scenario:

1. If in step 3 of primary scenario, Administrator enters invalid data, system displays error message and requires entering valid data.
2. If in step 5 of primary scenario, Administrator enters invalid data, the system displays an error message and requires entering valid data.
3. At any time before submitting the form the user can cancel this operation and return to main menu or logout of the system.

3.1.15 Create User

Administrator can add more users to the system.

Actor(s) Name: Administrator

Priority: High

Precondition(s): Authenticated user is an administrator and is logged into the system.

Postcondition(s): New user has been created and is stored in the database.

Primary Scenario:

1. Administrator selects Add User.
2. System request to enter first name, enter middle initial, enter last name, enter user ID, enter password, enter what type of user for security purposes.
3. Administrator enters requested information and submits form.
4. If Administrator provided valid input, system displays confirmation message indicating that user has been successfully added.

Secondary Scenario:

1. If in step 3 of primary scenario, Administrator enters invalid data, system displays error message and requires entering valid data.
2. At any time before submitting the form the user can cancel this operation and return to main menu or logout of the system.

3.1.16 Back up

This use case enable administrator to back up database.

Actor(s) Name: Administrator

Priority: Medium

Preconditions(s): Must be logged on with administration rights.

Postcondition(s): Create data restore in the database.

Extended Use Case: None

Primary Scenario:

1. Administrator will back up at will.

Secondary Scenario: None

Other Requirements: None

3.1.17 Restore Data

This use case enable administrator to restore data.

Actor(s) Name: Administrator

Priority: Medium

Precondition(s): Must be logged on with administration rights.

Postcondition(s): Database successfully restored.

Extended Use Case: None

Primary Scenario:

1. Administrator will restore data.

Secondary Scenario: None

Other Requirements: None

3.1.18 Database

Data Store: Reservations

Composition (Elements and/or Data Structures)

Rname+

Affiliation+

Acres+

LandCondition+

Street+

City+

County+

Zip+

Land+

Data Store: Water

Composition (Elements and/or Data Structures)

Rname+

WaterName+

Type+

milesTo+

note+

Data Store: Plan

Composition (Elements and/or Data Structures)

Rname+

PlanName+

Note+

Data Store: Roads

Composition (Elements and/or Data Structures)

Rname+

RoadName+

Type+

milesTo+

condition+

Data Store: Contact

Composition (Elements and/or Data Structures)

Rname+

Lname+

Fname+

Title+

Phone+

Email+

Street+

City+

County+

Zip+

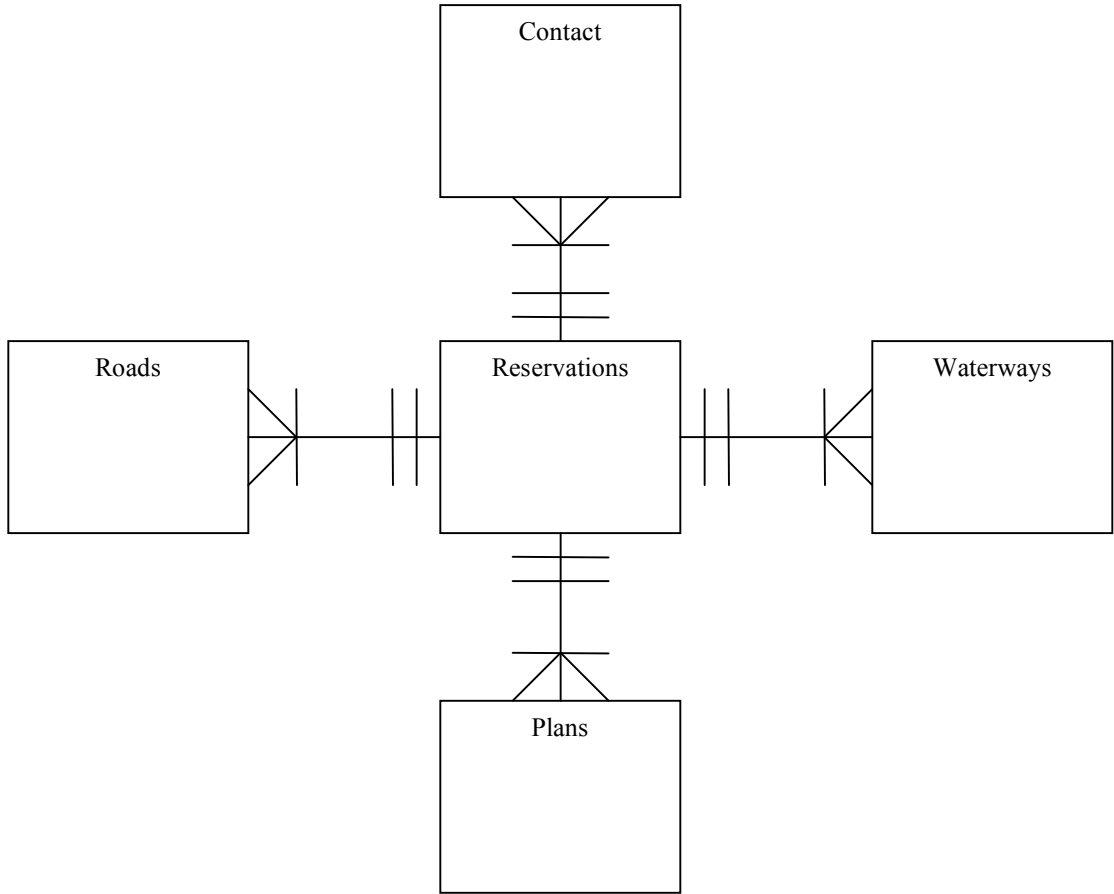
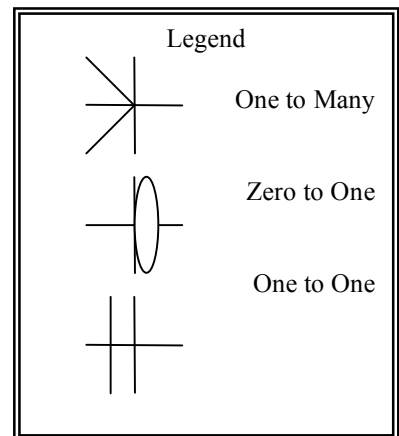


Figure 2.1 Entity Relationship Diagram



3.2 Performance Requirements

- ITCC server must be able to run PHP file successfully and MySQL.
- Users must be able to navigate on the internet using internet browser.
- Data Stores include a MySQL database for the aspects of the planning and development part of the project. See section 3.1.n+1 for full description of planning and development tables. There will be a separate table for users, user passwords, and privilege level. If the online survey is implemented then a table design will be created for that purpose. Sponsor has not at this time given a yes or no on survey(s).
- Performance and number of users depends on ITCC server hardware.

3.3 Design Constraints

- The Elder survey is meant to be a spot check of Native American elder population here in California. For this kind of temporary data there needs not be any unique keys. The surveys will be taken anonymously. The data generated will give a general idea of how the Native American population is surviving and where their needs are.

3.4 Quality Attributes

ITCC is a small office and as such there are no foreseen limitations or excessive user factors which apply. If the Elder survey is implemented online, then the amount of users will be limited by the server capacity. PHP and MySQL can handle a fair amount of traffic, and this amount of traffic will never be reached in the lifetime of the software.

This section contains the required system attributes including

- reliability
- maintainability
- program quality
- security
- transferability/conversion
- operational quality

3.4.1 Reliability

The system shall be stable. Server side software shall run on a server and not fail unless the server is unavailable.

3.4.2 Maintainability

ITCC Software system shall be easy to maintain and modify to correct faults, improve performance and adapt to a changing environment.

3.4.3 Program Quality Attributes

System shall satisfy the following program quality attributes related to readability, understandability, comprehensibility, robustness, and the speed of execution.

- The code for the program shall be fully documented so that it shall be easy to read and understand logic.
- Program shall handle all incorrect input data and be protected against misuse. Error messages will be displayed if any input data is incorrect.
- The speed of the program execution will mostly depend on database management system due to use of SQL queries. SQL queries will be used as a means of accessing data in the ITCC database and retrieving result to the user interface.

3.4.4 Security

The main DB has information which should be viewable by all leading members of ITCC. The specific sections of the DB, which include but are not limited to Elder health care, and Domestic Violence should be private and only accessible by the specific ITCC member in charge of that information. To solve this problem we are going to create access rites for each user. There level of accessibility will determine whether the user can input, view, or modify the data in the DB.

	Level	Frequency
Input	Low	Hi
View	Mid – Hi	Mid
Modify	Hi	Low

Unauthorized users are any user with out permission from ITCC administration. With out passwords and usernames these users will have no access rites and should not be able to view the information in the DB. ITCC will receive a physical copy of the project and a user manual on how to install and run the project on there Windows NT server system in case of physical destruction of the software occurs. If data is lost the ITCC will have to manually input the information back into the DB, since most of the data for the main DB is information about specific tribes the information should be accessible if lost from the DB. For the specific sections of the DB if information is lost then it retrievable through the backup system already implemented by ITCC. Computer viruses and hackers are outside the scope of our project and ITCC existing systems should have security already in place for these contingencies.

3.4.5 Transferability/ Conversions

ITCC is currently using a Windows NT server. Our project will be confined to running on this configuration. But since we are using PHP and MySQL there are means to transfer the project to other systems. There would have to be some modification to PHP if system specific configurations are made but we will try to limit these to a minimum. A list of these concerns can be found at http://linuxalpha1.eicn.ch/OREilly_books/books/webprog/php/ch15_02.htm.

As for MySQL there are migration tools to transfer already existing DB's to other systems. Information on this can be found at <http://dev.mysql.com/downloads/migration-toolkit/1.0.html>.

3.4.6 Operation Quality Attributes

- Ergonomic aspects: Our sponsor runs Windows NT and is used to the look and feel of windows programming so a goal of ours should be to produce a user interface that will be easy for our sponsor to learn.
- User interface language: The project will be written in PHP and it is possible to build the GUI with this language.
- Help system: will be in the form of a user manual.
- Program Quality: The program must be running for the majority of the day without crashing.
- Program Readability: Program will be implemented in standard text size (12 point font size).
- Program robustness: Direct 6 elected to write this program in the language of PHP and MySQL on a Windows NT server. While there are ways to convert the program to other systems with tools we are not responsible for making these changes.
- Program speed of execution: will be determined by the amount of data that is in the system

3.4.7 Operations

There are four main modes of operation:

- Web User
- Simple User
- Super User
- Admin User

The administrator will be able to perform back up and administration tasks such as create user, remove user, reset password, update user, and modify user rights. The Super User mode allows user to create table, remove table, and update table. The simple user mode is a read-only mode with permissions to generate and print reports. The website user mode allows the user to take survey online by inputting information onto the website using a graphical user interface.

3.4.8 Site Adaptations

Direct6 Software Solutions will provide the setup of this platform that utilizes and establishes open source framework. This framework will require limited, if any, modification to run on the hosted platform. The platform is expected to be Window NT derivative. Any modifications to the program to accommodate the hosting environment's platform will be made during the design and coding.

4. APPROVALS

This section contains signatures used to indicate approval of and agreement to the conditions and commitments contained in the software proposal.

Project Sponsor:

Stephen Archer, Sponsor ITCC

Team Members:

Dave Diel, Project Manager

Jared Arbaugh, Chair

Khae Saetarn

Ben Tovar

Kou Vang

Ben Mackin

Faculty Advisor:

Dr. Jiang, Computer Science Department

Appendix A Data dictionary

- **TABLES**

Data Table: ROAD

Description: This table keeps the reservation's roads information

How it is set: The table is set by an administrator

How it is used: The table is used to describe the reservation's roads

Data Table: PLAN

Description: This table keeps the reservation's plans information

How it is set: The table is set by an administrator

How it is used: The table is used to describe the reservation's plans

Data Table: RESERVATION

Description: This table keeps the reservation information

How it is set: The table is set by an administrator

How it is used: The table is used to identify reservations

Data Table: WATERWAY

Description: This table holds the fields for the water way.

How it is set: It is created by the Database Administrator.

How it is used: The table is used to describe the reservation's water way.

Data Table: ROADS

Description: This table holds the fields for a road

How it is set: It is created by the Database Administrator.

How it is used: The table is used to describe the road condition.

Data Table: CONTACTS

Description: This table holds the fields for contact person

How it is set: It is created by the Database Administrator.

How it is used: The table is used to identify the contact information.

- **ELEMENTS**

Data Element: **TName**

Description: Field for holding tribe names.

How it is set:

How it is used: Field is used for identifying tribes' names in relation to the registered.

Included in Data Store: CONTACT, TRIBES

Value Type: String

Data Element: **Address**

Description: This field holds a location address.

How it is set:

How it is used: The field is used for identifying the current registered address.

Included in Data Store: CONTACT, TRIBES, RESERVATIONS

Value Type: String

Data Element: **Street**

Description: Field used for holding street address.

How it is set:

How it is used: The field displays address of the current registrar.

Included in Data Store: CONTACT, TRIBES, RESERVATIONS

Value Type: String

Data Element: **City**

Description: Field for holding city names

How it is set:

How it is used: Used for identifying the current registered city name.

Included in Data Store: CONTACT, TRIBES, RESERVATIONS

Value Type: String

Data Element: **County**

Description: Field used for holding county names

How it is set:

How it is used: the field displays the name of the county the current registered is in

Included in Data Store: CONTACT, TRIBES, RESERVATIONS

Value Type: String

Data Element: **Zip**

Description: field used for holding zip codes

How it is set:

How it is used: The field is use for entering the zip code the current registered is in

Included in Data Store: CONTACT, TRIBES, RESERVATIONS

Value Type: String

Data Element: **EmailAdd**

Description: Field for holding email addresses

How it is set:

How it is used: Used for storing the current registered email address for contact.

Included in Data Store: CONTACT

Value Type: String

Data Element: **Phone**

Description: Field used for storing phone numbers

How it is set:

How it is used: The field is used for entering the current registered phone number.

Included in Data Store: CONTACT

Value Type: String

Data Element: **Title**

Description: Field used for holding registered titles

How it is set:

How it is used: Used for identifying the registered title in the reservation

Included in Data Store: CONTACT

Value Type: String

Data Element: **LName**

Description: Field used for holding last names

How it is set:

How it is used: The field displays last name of the current registered.

Included in Data Store: CONTACT

Value Type: String

Data Element: **FName**

Description: field used for holding first names.

How it is set:

How it is used: Used for displaying or entering the first name of the current registree.

Included in Data Store: CONTACT

Value Type: String

Data Element: **RdName**

Description: Field holds road names.

How it is set:

How it is used: Used to identify the road name the current registered is on.

Included in Data Store: ROAD

Value Type: String

Data Element: **LandCond**

Description: Field holds condition descriptions for land/road

How it is set:

How it is used: Used for displaying the condition of the land/roads stored in the database.

Included in Data Store: RESERVATION, ROAD

Value Type: String

Data Element: **RName**

Description: Field holds the names of Reservations

How it is entered: Registered users with at minimum of super user rights enters data

How it is used: The field is used for entering the names of Reservations

Included in Data Store: RESERVATIONS, WATERWAYS, ROADS, PLAN

Value Type: String

Data Element: **WName**

Description: Field holds the name of the Waterways.

How it is entered: Registered users with at minimum of super user rights enters data

How it is used: Used for entering the names of the Water Ways located near reservations

Included in Data Store: WATERWAY

Value Type: String

Data Element: **Type**

Description: Field name used for entering the type of Water Way.

How it is entered: Registered users with at minimum of super user rights enters data

How it is used: Used for entering a short description of the water way

Included in Data Store: ROADS, PLANS, WATERWAYS

Value Type: String

Data Element: **MilesTo**

Description: Field holds the mile rage distance from a Reservation to the Water Way.

How it is entered: Registered users with at minimum of super user rights enters data

How it is used: Used for determining distance from reservations to waterways

Included in Data Store: WATERWAY, ROADS

Value Type: Double

Data Element: **Notes**

Description: Field holds comments in regards to the current Water Way entered

How it is entered: Registered users with at minimum of super user rights enters data

How it is used: Field displays surveyor notes in regards to the water ways.

Included in Data Store: ROAD, RESERVATIONS, WATERWAYS, PLAN

Value Type: String

Data Element: **Land**

Description: Field holds comments in regards to the land

How it is entered: Registered users with at minimum of super user rights enters data

How it is used: Field displays information about reservation land used.

Included in Data Store: RESERVATIONS

Value Type: String

Data Element: **Population**

Description: Field holds the population

How it is entered: Registered users with at minimum of super user rights enters data

How it is used: Field displays information amount of reservation's population.

Included in Data Store: RESERVATIONS

Value Type: String

Data Element: **Locations**

Description: Field holds the reservation's location

How it is entered: Registered users with at minimum of super user rights enters data

How it is used: Field displays information about reservation's location.

Included in Data Store: RESERVATIONS

Value Type: String

Data Element: **TribeName**

Description: Field holds name of the tribes in the reservation

How it is entered: Registered users with at minimum of super user rights enters data

How it is used: Field displays information reservation.

Included in Data Store: RESERVATIONS

Value Type: String