|  |  |
| --- | --- |
| **Team:** |  |
| **Team Members:** |  |
| **Date:** |  |
| **Project Title:** |  |
| **Team Leader:** |  |

**Note: All diagrams should be clearly labeled. Remove all text that is shown in RED.**

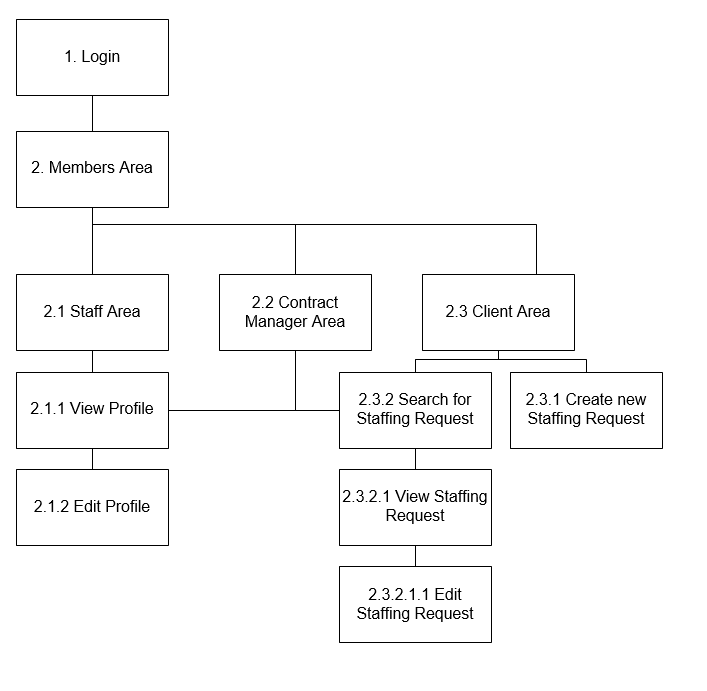
# Scope

This section will define the scope of the project by defining and describing the System, the major functions of the application, and the database.

## System Description

Describe your application – this can / should be a copy from the Requirements Document

## Major Software Functions



1. Login

The login function will accept a user name and password. The user name will be a client user name, a staff member employee id number, or a contract manager user name. The password is the password associated with the account.

If the login information is valid, we retrieve the user access level from the database and store it inside the current session.

1. Members Area

When the members are is loaded, the contents of the page will be based on the user access level gotten for the currently logged in user. If the user has not logged into the database, we will redirect them back to the login page.

* 1. Staff Area

The contents of the staff area changes based on the user access level which accesses the page. Clients are redirected to the client area if they try to access the staff area.

* + 1. View Profile

A contract manager which logs into the staff area is given the ability to search for a staff member based on their employee id number. A contract manager has the ability to view the staff member information and check their availability for staffing requests.

* + 1. Edit Profile

A staff member which logs into the staff area is given the ability to edit their information, upload a new picture, upload a new resume, or update their availability.

* 1. Contract Manager Area

The contract manager area gives the contract manager the ability to access the staff area or the client area. Along with the ability to access the staff or client areas the contract manager is given a list of the new staffing requests so they may validate and close them out.

* 1. Client Area

The contents of the client area changes based on the user access level which accesses the page. Staff members are redirected to the staff area is they try to access the client area.

* + 1. Create new Staffing Request

A client which logs into the client area is given the ability to create a new staffing request based on type of work, education, experience, location and salary. They have the ability to select up to three (3) potential candidates based on these criteria for the request.

* + 1. Search for Staffing Request

A client may search for the staffing request based on the confirmation number they received after creating the request.

* + - 1. View Staffing Request

After searching for the request, clients are given the ability to view the staffing request information.

* + - * 1. Edit Staffing Request

Contract managers have the ability to validate and close out staffing requests based on staff member availability.

## Database Description

What type of DB will you use for the application?

## Design Constraints and Limitations

Describe any and all design constraints and limitations

# Design Description

This section will provide an overview of the application design by providing a description of the data to be used and managed by the application, define the flow of data through the system, describe the architecture of the programs within the application and define the how the components will interface.

## Data Description

Provide a normalized ERD that contains all of the primary and alternate key. This is the final version of the ERD you created in the Requirements Specification and at this point it is now a SCHEMA. Then provide a description of each table in the ERD and list the fields and their characteristics

You should also provide a description of each relationship (it helps to understand the data dependencies.)

## Data Flow

In this section you will define and describe how the processes interface with agents/actors and/or datastores. This can be done with either a Context Level DFD (SAD) or a Package Diagram (OOAD). You should also provide as much narrative as necessary to describe each component of the diagram. The goal is to provide the reader with an outline of the system you are designing. Each component will be defined in detail in section 3.0.

## Program Architecture

This section is best completed with either a Physical DFD (SAD) or a Class Diagram (OOAD). The purpose is to define the physical architecture of the programs as they interface with each other. The diagram should include every module that will be part of the application.

## Component Interfaces

Provide a screen navigation diagram and an Event chart/list. If it needs to be described by a narrative then you should provide that as well.

# Detailed Design

This section will define and describe the design details for each of the components in the application.

(The intent of this section is to prepare for the build phase. These detailed design specifications will become a program in your system)

## *Component/Module One* (Change the name “Component/Module One to be the actual name of the module you are designing. Repeat section 3.1 for ea. component)

### Processing Description

Provide a Primitive DFD (SAD) or a Use Case Diagram (OOAD) and a brief narrative

### Interface Description

There are several items that should be covered here:

* If the component has a GUI, display the image of the GUI and define each field, button, link on the GUI.
* if the component receives data, describe the data coming in and where it should go once it is processed (Input/Output)
* If there is data to be passed to another component that should be listed

This section will have a detailed written description of what each button, link, data entry field etc., does on the page, with an image of the proposed page layout, for each page.

### Pseudocode

The processing rules for the component – in logical order.  But remember, language independent. (link to a great example) <http://www.wiley.com/college/busin/icmis/oakman/outline/chap05/slides/pseudo.htm>

Put things here if it is needed to help explain the business rule functionality of the screen. For example if you need to check to make sure a customer has no outstanding balance past 60 days in order for them to generate another order identify what data values need to be used to do this and the math functions that need to be used.

If Customer Balance > 0 and Last Purchase Date < (current date – 60 days) then   
 Deny Purchase  
Else  
 Allow Purchase

You can break this down into functions or procedures if that is the best way to approach this for your project.

### Modules Uses

If this component is a function that is called by other functions, such as a DB call to retrieve data, insert data, update data or delete data, or some type of standard calculation used by multiple modules, you will describe that here. If this section does not apply to the component, simply delete the section.

**Document Work Log:**

*To assist in assessing the contributions made by the individual team members, the team must complete the table below:*

|  |  |  |
| --- | --- | --- |
| ***Section*** | ***Team Member - Primary*** | ***Team Member - Secondary*** |
| 1. *Scope* |  |  |
| * 1. *System Description* |  |  |
| * 1. *Major Software Functions* |  |  |
| * 1. *Database Description* |  |  |
| * 1. *Database Constraints and Limitations* |  |  |
| 1. *Design Description* |  |  |
| * 1. *Data Description* |  |  |
| *Schema* |  |  |
| *Table and Relationships – Data Dictionary* |  |  |
| * 1. *Data Flow* |  |  |
| * 1. *Program Architecture* |  |  |
| * 1. *Component Interfaces* |  |  |
| 1. *Detailed Design Section -* ***One section (3.1 – 3.x) for each individual component or module*** | | |
| *Detailed Processing Description* |  |  |
| *Interface Description – with UI* |  |  |
| *Pseudo code, Flowcharts or Storyboards* |  |  |
| *Module Uses* |  |  |