Fall 2010 Capstone

**Mobile Client Query Application**

Software Design Document

Dru Luce-Edwards and Jeremy Smith

Date: (2/8/2011)

Table of Contents

1. **INTRODUCTION** 3
   1. Purpose 3
   2. Scope 3
2. **SYSTEM ARCHITECTURE** 3
   1. Architectural Design 3
   2. Decomposition Description 4
   3. Design Rationale 5
3. **COMPONENT DESIGN** 5
   1. Component Description 5

3.1.1 Mobile Client Subsystem.......................................................................................5

* 1. Data Dictionary 7

1. **HUMAN INTERFACE DESIGN** 8
   1. Overview of User Interface 8
   2. Screen Images 8

4.2.1 Mobile Client UI 9

* 1. Screen Objects and Actions 9

1. **REQUIREMENTS MATRICIES** 9

**1. INTRODUCTION**

1.1 Purpose

This software design document describes the architecture and system design of the iOS ICTIS Application and is intended to inform stakeholders of the details of the design and the design process.

1.2 Scope

The main objective for this project is to provide a mobile application, the ICTIS, to our Client, on the iOS platform, that will give user department based information. The Mobile Client Query Application will empower inspectors with the ability to view and update data about the items they are inspecting while they are in the field, thus reducing office time to digitize their findings and upload them. To accomplish those goals the project will also include a central server that hosts the database, processes data requests, and has a graphical user interface to access the server/database directly.

Folder Structure

Create the initial project using the GOOD Project template. Then we will create different folders and groups according to the standards followed in Fidelity. Every classes name will be begin with a prefix ‘FIL’.

|  |  |  |
| --- | --- | --- |
| SI No | Folder | Purpose |
| 1 | ICTIS | This folder contains Main storyboard, Image Assets and Info.plist.  In this folder include some initial classes, AppDelegate and FILContainerViewController class. AppDelegate class is the application’s first launching classes. Here we are handling the OS events and GOOD authentication mechanisms.  Application\_Resources/Images contains all images required for application. Group/Folderise the images with in the main folder for better maintainability  In this folder also contains all the modularized folders and FILUtility Folder. |
| 2 | ICTIS/FILUtility | FILUtility folder contains application utility classes and reusable component classes. |
| 4 | ICTIS/LeftSlideMenu |  |
| 5 | ICTIS/ICTISCollectionMenu |  |
| 3 | ICTIS/Home |  |
| 4 | ICTIS/DataAnalysis |  |
| 5 | ICTIS/Middleware |  |
| 6 | ICTIS/ServiceManagement |  |
| 7 | ICTIS/ |  |

Basic Class Structure

Every ViewControllers should be derived from the FILBaseViewController.

Some common Functionality implemented in the FILBaseVieController.

We will override the methods in the child classes based on the requirements.

Design each module

Left Slide Menu

UI elements details

|  |  |  |  |
| --- | --- | --- | --- |
| **SI No** | **Field Name** | **Type** | **Description** |
| 1 | Profile Image |  |  |
| 2 | User Name | UILabel in FILSlideMenu | Name of the user is getting connected. Retrieved from GOOD Control |
| 3 | User Email | UILabel in FILSlideMenu | Email of the connected used. Retrieve from GOOD Control. |
| 4 | Menu Title | UILabel in UITableView header view | Header menu of the slide menu. Retrieve from FILLeftSlideMenu api. |
| 5 | Sub Menu Title | UILabel in UITableViewCell | SubMenu of the slide menu, coming under MenuTitle. Retrieve from FILSlideMenu api. |
| 6 | Settings | UIButton in FILSlideMenu | This will enable the user to change the Preference of the APP |
| 7 |  |  |  |

Action of UI elements

|  |  |  |  |
| --- | --- | --- | --- |
| **SI No** | **Field Name** | **Action** | **Description** |
| 1 | UserInformationSection(User image, user email and profile image including section) | Touch up inside | On tapping the user information section, the user information page is displayed. |
| 2 | Settings | Touch up inside | Navigate to the preference page. |
| 3 | Menu Title | Tap on the UITableView header | UITableView header row gets expanded to show the submenu Titles coming under the Menu Title. Also display the Collection Menu on ContainerView. |
| 4 | Sub Menu Title | Tap on the UITableView Cell | Naviagte to the submenu connected viewcontroller. |

Validations

If the user hasn’t set any profile image, a default image should be displayed.

If Slide Menu response is null, don’t show the menu.