**SYSTEM DESIGN DOCUMENT**

**INSTITUTIONS NAME AND LOGO**

**SOFTWARE DESIGN DOCUMENT FOR**

**TITLE:**

**PREPARED BY: STUDENTS NAME:**

**REG. NO.:**

**PROJECT SUPERVISOR:**

**PROJECT CORDINATOR:**

**DATES:**

**VERSION:**

**SYSTEM DESIGN DOCUMENT**

*Overview*

*The System Design Document describes the system requirements, operating environment, system and subsystem architecture, files and database design, input formats, output layouts, human-machine interfaces, detailed design, processing logic, and external interfaces.*

**SYSTEM DESIGN DOCUMENT**

**Table of Contents**

|  |  |  |  |
| --- | --- | --- | --- |
| [1](#page3) | [INTRODUCTION ..................................................................................................................](#page3) | | [4](#page3) |
|  | [1.1](#page4) | [Purpose and Scope ..........................................................................................................](#page4) | [4](#page4) |
|  | [1.2](#page4) | [Project Executive Summary ............................................................................................](#page4) | [4](#page4) |
|  | [1.2.1](#page4) | [System Overview ........................................................................................................](#page4) | [4](#page4) |
|  | [1.2.2](#page4) | [Design Constraints ......................................................................................................](#page4) | [4](#page4) |
|  | [1.2.3](#page4) | [Future Contingencies ..................................................................................................](#page4) | [4](#page4) |
|  | [1.3](#page4) | [Document Organization ..................................................................................................](#page4) | [4](#page4) |
|  | [1.4](#page4) | [Points of Contact .............................................................................................................](#page4) | [4](#page4) |
|  | [1.5](#page4) | [Project References ..........................................................................................................](#page4) | [4](#page4) |
|  | [1.6](#page5) | [Glossary ..........................................................................................................................](#page5) | [5](#page5) |
| [2](#page5) | [SYSTEM ARCHITECTURE .................................................................................................](#page5) | | [5](#page5) |
|  | [2.1](#page5) | [System Hardware Architecture .......................................................................................](#page5) | [5](#page5) |
|  | [2.2](#page5) | [System Software Architecture ........................................................................................](#page5) | [5](#page5) |
|  | [2.3](#page5) | [Internal Communications Architecture ...........................................................................](#page5) | [5](#page5) |
| [3](#page5) | [FILE AND DATABASE DESIGN ........................................................................................](#page5) | | [5](#page5) |
|  | [3.1](#page6) | [Database Management System Files ..............................................................................](#page6) | [6](#page6) |
|  | [3.2](#page6) | [Non-Database Management System Files ......................................................................](#page6) | [6](#page6) |
| [4](#page6) | [HUMAN-MACHINE INTERFACE ......................................................................................](#page6) | | [6](#page6) |
|  | [4.1](#page6) | [Inputs...............................................................................................................................](#page6) | [6](#page6) |
|  | [4.2](#page7) | [Outputs ............................................................................................................................](#page7) | [7](#page7) |
| [5](#page7) | [DETAILED DESIGN .............................................................................................................](#page7) | | [7](#page7) |
|  | [5.1](#page8) | [Hardware Detailed Design ..............................................................................................](#page8) | [8](#page8) |
|  | [5.2](#page8) | [Software Detailed Design ...............................................................................................](#page8) | [8](#page8) |
|  | [5.3](#page9) | [Internal Communications Detailed Design .....................................................................](#page9) | [9](#page9) |
| [6](#page9) | [EXTERNAL INTERFACES ..................................................................................................](#page9) | | [9](#page9) |
|  | [6.1](#page9) | [Interface Architecture .....................................................................................................](#page9) | [9](#page9) |
|  | [6.2](#page10) | [Interface Detailed Design .............................................................................................](#page10) | [10](#page10) |
| [7](#page10) | [SYSTEM INTEGRITY CONTROLS ..................................................................................](#page10) | | [10](#page10) |

**SYSTEM DESIGN DOCUMENT**

**1 INTRODUCTION**

**1.1** **Purpose and Scope**

The purpose of this document is to present a detailed description of the Home application. Firstly it is intended to guide the developer in design of the project. Equally it is intended to enable the instructor to understand the project. Lastly this document could be used by designers who try to upgrade or modify the project.

This document gives a detailed description of the software architecture of the home application system. It specifies the structure and design as described in the SRS. It describes some use cases. Class diagrams and some actual user interfaces as implemented in the system.

**1.2** **Project Executive Summary**

This section provides a description of the project from a management perspective and an overview of the framework within which the conceptual system design was prepared. If appropriate, include the information discussed in the subsequent sections in the summary.

1.2.1 System Overview

User interface

Employer

Worker

Firebase authentication

Employer database

Worker database

1.2.2 Design Constraints

This section describes any constraints in the system design (reference any trade-off analyses conducted such, as resource use versus productivity, or conflicts with other systems) and includes any assumptions made by the project team in developing the system design.

Communication Protocal limitations:

One can only post and view using the internet only. It is internet based thus without internet the application cannot show its full functionality.

Interfaces to other application limitation:

The application is limited to Android phones that means that IOS phones cannot use the application and any other phone that is not Android.

Software requirements limitation:

How fast can the phone access internet.

Assumption made was that most android phone’s were above API(Application Programming Interface) level 16 that is Jellybean.

1.2.3 Future Contingencies

The application is intended to enable prospective employers and domestic workers to communicate with each other thus at some point features like a chat room or call functionality should be implemented to enable ease of communication.

**1.3** **Document Organization**

This section describes the organization of the Systems Design Document.

**1.4** **Points of Contact**

This section provides the organization code and title of the key points of contact (and alternates if appropriate) for the information system development effort. These points of contact should include the Project Manager, System Proponent, User Organization, Quality Assurance (QA) Manager, Security Manager, and Configuration Manager, as appropriate.

**1.5 Project References**

This section provides a bibliography of key project references and deliverables that have

**SYSTEM DESIGN DOCUMENT**

been produced before this point.

<https://www.cise.ufl.edu/class/cen3031fa11/.../examples/SDD_Example_1_2011.pdf> accessed on 23/05/2018

<https://www.google.com/search?q=firebase+hardware+architecture&tbm=isch&tbo=u&source=univ&sa=X&ved=0ahUKEwj9iuTr-KDbAhWTTX0KHdyqBcEQsAQIWw&biw=1229&bih=585#imgrc=43rcr7lidAvzvM>: accessed on 23/05/2018

**1.6** **Glossary**

API-Application Programming Interface

1. **SYSTEM ARCHITECTURE**

.

Workers

Application Interface

Internet

Web browser

Firebase

Password

Email

Authentication

Charges

Prev employer

experience

location

Phone no

Name

Database

Block diagram for workers

In the home application system the workers interface and how they interact with the system is different from interface and how they interact with the system. The home application is a system combination of two applications one for the workers and one for the employers.

Employers

Application Interface

Internet

Web browser

Firebase

Password

Email

Authentication

Additional info

Duration

Task

Name

Database

Block diagram for employers

Employers

Workers

Context diagram showing major actors interacting with the system

**2.1** **System Hardware Architecture**

Mobile phone which consists of a central processing unit, memory card and touch screen. The touch screen will be used by user to interact with the application.

The internet will be used to connect to firebase.

firebase

Login

View Controller

Submit view controller

Submission view controller

## 2.2 System Software Architecture

In this section, describe the overall system software and organization. Include a list of software modules (this could include functions, subroutines, or classes), computer languages, and programming computer-aided software engineering tools (with a brief description of the function of each item). Use structured organization diagrams/object-oriented diagrams that show the various segmentation levels down to the lowest level. All features on the diagrams should have reference numbers and names. Include a narrative that expands on and enhances the understanding of the functional breakdown. If appropriate, use subsections to address each module.

### 2.2.1 User registration

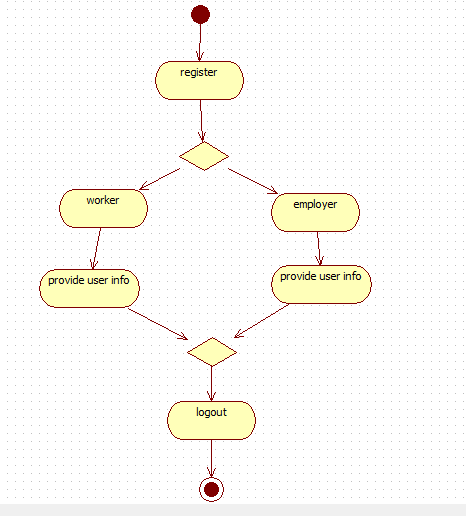


Figure 4.0 Activity diagram for user registration

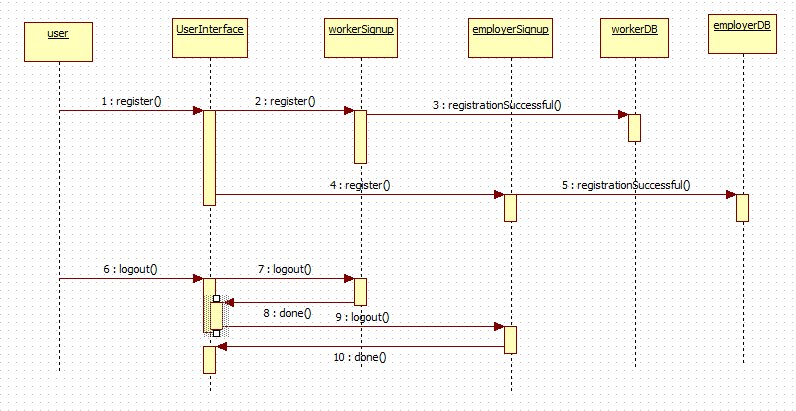


Figure 4.1 Sequence diagram for registration

### 2.2.2 Login

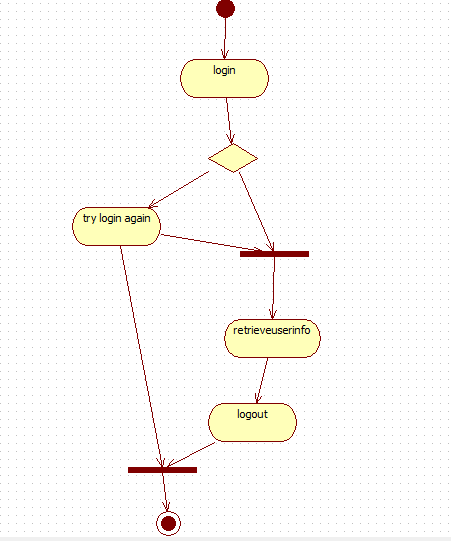


Figure 4.3 Activity diagram for login

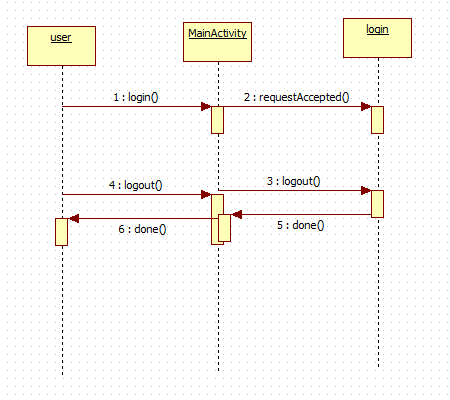


Figure 4.4 Sequence diagram for login

### 2.2.3 Worker functionality

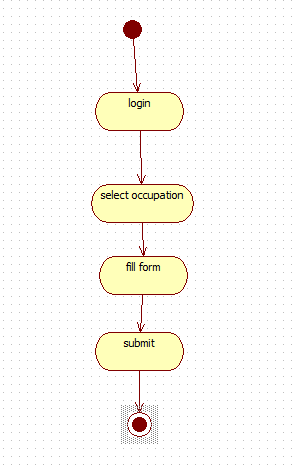


Figure 4.5 Activity diagram for a worker

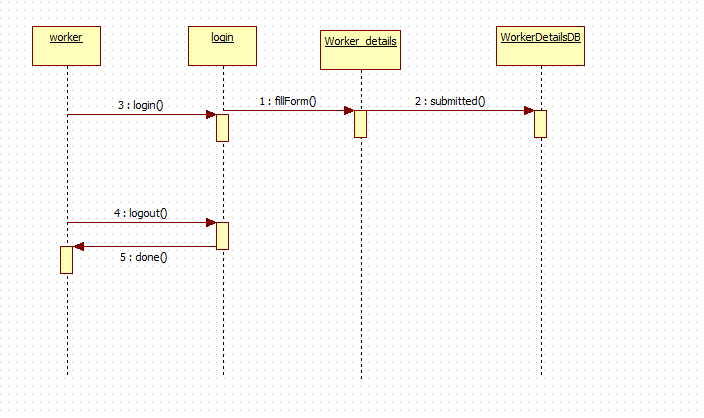


Figure 4.6 Worker Sequence diagram

### 2.2.4 Employer functionality

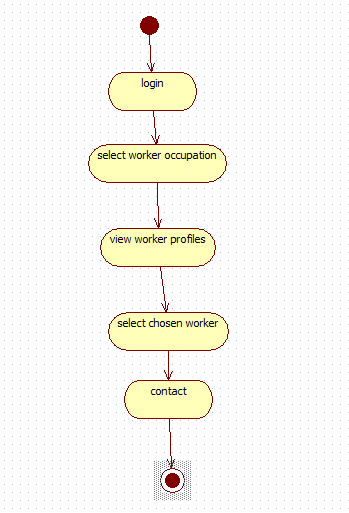


Figure 4.6 Employer activity diagram

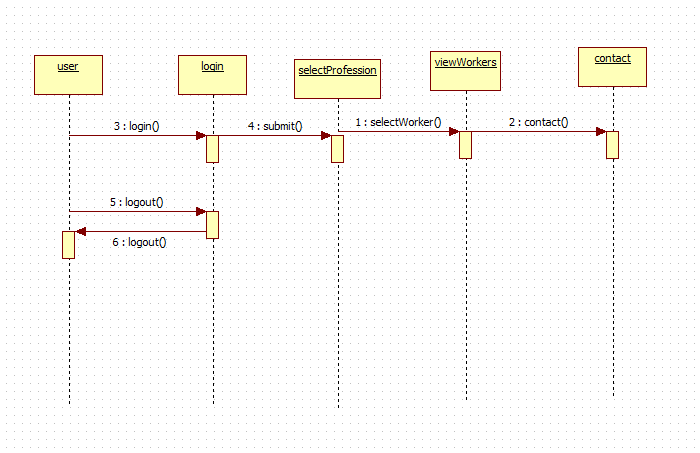


Figure 4.7 Employer sequence diagram

**Note:** The diagrams should map to the FRD data flow diagrams, providing the physicalprocess and data flow related to the FRD logical process and data flow.

**2.3** **Internal Communications Architecture**

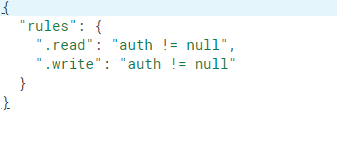
In this section, describe the overall communications within the system; for example, LANs, buses, etc. Include the communications architecture(s) being implemented, such as X.25*,* Token Ring, etc. Provide a diagram depicting the communications path(s) between the system and subsystem modules. If appropriate, use subsections to address each architecture being employed.

**Note:** The diagrams should map to the FRD context diagrams.

1. **FILE AND DATABASE DESIGN**

Interact with the Database Administrator (DBA) when preparing this section. The section should reveal the final design of all database management system (DBMS) files and the non-DBMS files associated with the system under development. Additional information may add as required for the particular project. Provide a comprehensive data dictionary showing data element name, type, length, source, validation rules, maintenance (create, read, update, delete (CRUD) capability), data stores, outputs, aliases, and description. Can be included as an appendix.

The database being used for this system is firebase database. The user needs access to the internet in order to be able to access the firebase database. The reason for using firebase database is because the database is not in a remote area and all users from anywhere can access it. The other reason is it offers large storage space for users which might be limited in a computer. One of the rules on accessing the database is that the user needs to be authenticated first in order to access the database. The rule is shown below:



In authentication the user has to enter their correct email and 8 digit character password.

The emails will be stored in the firebase authentication folder shown below:

**SYSTEM DESIGN DOCUMENT**

**3.1** **Database Management System Files**

This section reveals the final design of the DBMS files and includes the following information, as appropriate (refer to the data dictionary):

* Refined logical model; provide normalized table layouts, entity relationship diagrams, and other logical design information
* A physical description of the DBMS schemas, sub-schemas, records, sets, tables, storage page sizes, etc.
* Access methods (such as indexed, via set, sequential, random access, sorted pointer array, etc.)
* Estimate of the DBMS file size or volume of data within the file, and data pages, including overhead resulting from access methods and free space
* Definition of the update frequency of the database tables, views, files, areas, records, sets, and data pages; estimate the number of transactions if the database is an online transaction-based system

**3.2** **Non-Database Management System Files**

In this section, provide the detailed description of all non-DBMS files and include a narrative description of the usage of each file—including if the file is used for input, output, or both; if this file is a temporary file; an indication of which modules read and write the file, etc.; and file structures (refer to the data dictionary). As appropriate, the file structure information should:

* Identify record structures, record keys or indexes, and reference data elements within the records
* Define record length (fixed or maximum variable length) and blocking factors
* Define file access method—for example, index sequential, virtual sequential, random access, etc.
* Estimate the file size or volume of data within the file, including overhead resulting from file access methods
* Define the update frequency of the file; if the file is part of an online transaction-based system, provide the estimated number of transactions per unit time, and the statistical mean, mode, and distribution of those transactions

1. **HUMAN-MACHINE INTERFACE**

This section provides the detailed design of the system and subsystem inputs and outputs relative to the user/operator. Any additional information may be added to this section and may be organized according to whatever structure best presents the operator input and output designs. Depending on the particular nature of the project, it may be appropriate to repeat these sections at both the subsystem and design module levels. Additional information may be added to the subsections if the suggested lists are inadequate to describe the project inputs and outputs.

**4.1 Inputs**

This section is a description of the input media used by the operator for providing

**SYSTEM DESIGN DOCUMENT**

information to the system; show a mapping to the high-level data flows described in Section 1 .2.1, System Overview. For example, data entry screens, optical character readers, bar scanners, etc. If appropriate, the input record types, file structures, and database structures provided in Section 3, File and Database Design, may be referenced. Include data element definitions, or refer to the data dictionary.

Provide the layout of all input data screens or graphical user interfaces (GUTs) (for example, windows). Provide a graphic representation of each interface. Define all data elements associated with each screen or GUI, or reference the data dictionary.

This section should contain edit criteria for the data elements, including specific values, range of values, mandatory/optional, alphanumeric values, and length. Also address data entry controls to prevent edit bypassing.

Discuss the miscellaneous messages associated with operator inputs, including the following:

* Copies of form(s) if the input data are keyed or scanned for data entry from printed forms
* Description of any access restrictions or security considerations
* Each transaction name, code, and definition, if the system is a transaction-based processing system

**4.2 Outputs**

This section describes of the system output design relative to the user/operator; show a mapping to the high-level data flows described in Section 1.2.1. System outputs include reports, data display screens and GUIs, query results, etc. The output files are described in Section 3 and may be referenced in this section. The following should be provided, if appropriate:

* Identification of codes and names for reports and data display screens
* Description of report and screen contents (provide a graphic representation of each layout and define all data elements associated with the layout or reference the data dictionary)
* Description of the purpose of the output, including identification of the primary users
* Report distribution requirements, if any (include frequency for periodic reports)
* Description of any access restrictions or security considerations

1. **DETAILED DESIGN**

This section provides the information needed for a system development team to actually build and integrate the hardware components, code and integrate the software modules, and interconnect the hardware and software segments into a functional product. Additionally, this section addresses the detailed procedures for combining separate COTS packages into a single system. Every detailed requirement should map back to the FRD, and the mapping should be presented in an update to the RTM and include the RTM as an appendix to this design document.

**SYSTEM DESIGN DOCUMENT**

**5.1** **Hardware Detailed Design**

A hardware component is the lowest level of design granularity in the system. Depending on the design requirements, there may be one or more components per system. This section should provide enough detailed information about individual component requirements to correctly build and/or procure all the hardware for the system (or integrate COTS items).

If there are many components or if the component documentation is extensive, place it in an appendix or reference a separate document. Add additional diagrams and information, if necessary, to describe each component and its functions, adequately. Industry-standard component specification practices should be followed. For COTS procurements, if a specific vendor has been identified, include appropriate item names. Include the following information in the detailed component designs (as applicable):

* Power input requirements for each component
* Signal impedances and logic states
* Connector specifications (serial/parallel, 11-pin, male/female, etc.)
* Memory and/or storage space requirements
* Processor requirements (speed and functionality)
* Graphical representation depicting the number of hardware items (for example, monitors, printers, servers, I/O devices), and the relative positioning of the components to each other
* Cable type(s) and length(s)
* User interfaces (buttons, toggle switches, etc.)
* Hard drive/floppy drive/CD-ROM requirements
* Monitor resolution

**5.2** **Software Detailed Design**

A software module is the lowest level of design granularity in the system. Depending on the software development approach, there may be one or more modules per system. This section should provide enough detailed information about logic and data necessary to completely write source code for all modules in the system (and/or integrate COTS software programs).

If there are many modules or if the module documentation is extensive, place it in an appendix or reference a separate document. Add additional diagrams and information, if necessary, to describe each module, its functionality, and its hierarchy. Industry-standard module specification practices should be followed. Include the following information in the detailed module designs:

* A narrative description of each module, its function(s), the conditions under which it is used (called or scheduled for execution), its overall processing, logic, interfaces to other modules, interfaces to external systems, security requirements, etc.; explain any algorithms used by the module in detail

**SYSTEM DESIGN DOCUMENT**

* For COTS packages, specify any call routines or bridging programs to integrate the package with the system and/or other COTS packages (for example, Dynamic Link Libraries)
* Data elements, record structures, and file structures associated with module input and output
* Graphical representation of the module processing, logic, flow of control, and algorithms, using an accepted diagramming approach (for example, structure charts, action diagrams, flowcharts, etc.)
* Data entry and data output graphics; define or reference associated data elements; if the project is large and complex or if the detailed module designs will be incorporated into a separate document, then it may be appropriate to repeat the screen information in this section
* Report layout

**5.3** **Internal Communications Detailed Design**

If the system includes more than one component there may be a requirement for internal communications to exchange information, provide commands, or support input/output functions. This section should provide enough detailed information about the communication requirements to correctly build and/or procure the communications components for the system. Include the following information in the detailed designs (as appropriate):

* The number of servers and clients to be included on each area network
* Specifications for bus timing requirements and bus control
* Format(s) for data being exchanged between components
* Graphical representation of the connectivity between components, showing the direction of data flow (if applicable), and approximate distances between components; information should provide enough detail to support the procurement of hardware to complete the installation at a given location
* LAN topology

1. **EXTERNAL INTERFACES**

External systems are any systems that are not within the scope of the system under development, regardless whether the other systems are managed by the State or another agency. In this section, describe the electronic interface(s) between this system and each of the other systems and/or subsystem(s), emphasizing the point of view of the system being developed.

**6.1** **Interface Architecture**

In this section, describe the interface(s) between the system being developed and other systems; for example, batch transfers, queries, etc. Include the interface architecture(s) being implemented, such as wide area networks, gateways, etc. Provide a diagram depicting the communications path(s) between this system and each of the other systems, which should map to the context diagrams in Section 1.2.1. If appropriate, use subsections to address each interface being implemented.

**SYSTEM DESIGN DOCUMENT**

**6.2** **Interface Detailed Design**

For each system that provides information exchange with the system under development, there is a requirement for rules governing the interface. This section should provide enough detailed information about the interface requirements to correctly format, transmit, and/or receive data across the interface. Include the following information in the detailed design for each interface (as appropriate):

* The data format requirements; if there is a need to reformat data before they are transmitted or after incoming data is received, tools and/or methods for the reformat process should be defined
* Specifications for hand-shaking protocols between the two systems; include the content and format of the information to be included in the hand-shake messages, the timing for exchanging these messages, and the steps to be taken when errors are identified
* Format(s) for error reports exchanged between the systems; should address the disposition of error reports; for example, retained in a file, sent to a printer, flag/alarm sent to the operator, etc.
* Graphical representation of the connectivity between systems, showing the direction of data flow
* Query and response descriptions

If a formal Interface Control Document (ICD) exists for a given interface, the information can be copied, or the ICD can be referenced in this section.

1. **SYSTEM INTEGRITY CONTROLS**

Sensitive systems use information for which the loss, misuse, modification of, or unauthorized access to that information could affect the conduct of State programs, or the privacy to which individuals are entitled.

Developers of sensitive State systems are required to develop specifications for the following minimum levels of control:

* + Internal security to restrict access of critical data items to only those access types required by users
  + Audit procedures to meet control, reporting, and retention period requirements for operational and management reports
  + Application audit trails to dynamically audit retrieval access to designated critical data
  + Standard Tables to be used or requested for validating data fields
  + Verification processes for additions, deletions, or updates of critical data

Ability to identify all audit information by user identification, network terminal identification, date, time, and data accessed or changed.