SYSTEM PROPOSAL DOCUMENT

for

UR Covered, Inc., Auto Insurance Claims Mobile Application

Charles Adkins

JANURARY 5, 2020

TABLE OF CONTENTS

UR Covered, Inc. System Proposal Document

1.0 INTRODUCTION SUMMARY

UR Covered, Inc. is a large auto insurance company in North America. The IT Department has determined that a mobile application would enhance business relations and customer satisfaction by providing customers with an extension to their current system (Rosenblatt, 2017). However, the managers want to ensure that, in doing so, the application and its functions do not erode the personal connection between claim case managers and customers.

1.1 Background

In current standing, UR Covered, Inc., has an excellent rapport with their customers, but lack the ability to communicate claim information in a diversified and streamlined fashion. Case managers provide customers with updates regarding their accounts status, but relations would improve, if customers, themselves, had access to more detailed information.

A desktop web application is being utilized, but without accessing it through a computer, customers are limited in their knowledge of their accounts. With the addition of the mobile application user interface, case managers will continue to remain involved throughout each stage of the claims lifecycle, while also adding a layer of independence for the customers. The user interface (UI), which is how users interact with a computer system, is the key to usability, including user satisfaction, business function and effectiveness (Rosenblatt, 2017).

1.2 Problem Statement

Currently, the claims management system relies heavily upon the direct involvement of case managers and their interactions with the customers. As workloads increase, the demand for claim resolution increases. This results in an increased workload for the case managers, as customers revert to contacting their case managers with questions directly. The mobile application will provide customers with an extension to their current means of communication and will add transparency towards the claims process.

Within the interface, customers will have the ability to view claim status, message their case managers and will be updated on new information regarding their account (Rosenblatt, 2017). This application will integrate with the current claim management system database information and will uphold the same quality of security and authentication protocols.

1.3 Target Audience

The target audiences include UR Covered, Inc. IT Department Management members, claims case managers and customers. Management members want interactions with customers to remain personal and efficient, in being so, the implementation of the mobile application must showcase its ability to enhance customer satisfaction, while also displaying its ability to communicate detailed account information accurately and safely. Claims case managers want a workload that is comparable to their work hours, yet would still enjoy interacting with their clients on a consistent and personal basis. The application will not only streamline the workload productivity, but will also connect them to their customers in a more diversified manner. Customers are requesting that their account and its' associated information be more accessible and detailed. With the introduction of the application, customers will have the ability to view and understand their claims, alongside their case managers, every step of the way.

2.0 SCOPE

2.1 Purpose

The purpose of this system requirements document is to propose the creation of the UR Covered, Inc. Claims account mobile application. This documentation provides an overview of the requirements for such an application and its layout, design and functionality characteristics and capabilities.

2.2 System Description

The UR Covered, Inc Claims Mobile Application project will provide an iOS mobile application that enables customers to access their claims account information, update account information, submit documentation, receive account updates and alerts and chat message with their Case Manager. The following report presents the proposed system requirements and the solutions to such requriements.

2.2.1 Current System

Currently, UR Covered, Inc. does not have a mobile application. Their current structure involves several databases, servers and an organizational network that manages company data and the web application.

2.2.2 New System

The mobile application will provide the customer with access to their account, the ability to submit documentation and specific application feature requests, such as chat message.

3.0 SYSTEM ANALYSIS

3.1 System Requirements

Outputs:

- Mobile application providing customers with access to claims account data and information
- Account alerts – new information, opt in or opt out feature
- Chat message capabilities
- Case Manager contact info contained within the application
- Actual iOS application, not a mobile version of the web application

Inputs:

- Customers must download the application and agree to the terms and conditions of the application
- Customers must enter their ID and Password prior to entering the landing page of the user interface.
- Customers must utilize and complete the data entry windows, when searching or filtering through account information, updating/submitting personal/claim information and during chat interactions and emails with their Case Manager
- Case Managers must enter relevant account information into the system, for customers to be alerted by messages regarding the new information

Processes:

- As Customers update their account and claims information, notification must be sent to respective Case managers
- Customers and Case managers can engage in chat messaging when requested by the customer
- As Customers respond to feedback surveys, IT management members should be notified

Performance:

- The system must be operational seven days a week, 365 days a year

- Mobile application must interface properly with the current claims management system database

- System must be incorporated into the capacity and robustness of the organizational network, to maintain high speed functionality during all hours of operation

- User experience survey feedback will be consolidated into Quarterly reports

Controls:

- The system must provide the same level of security as the web application

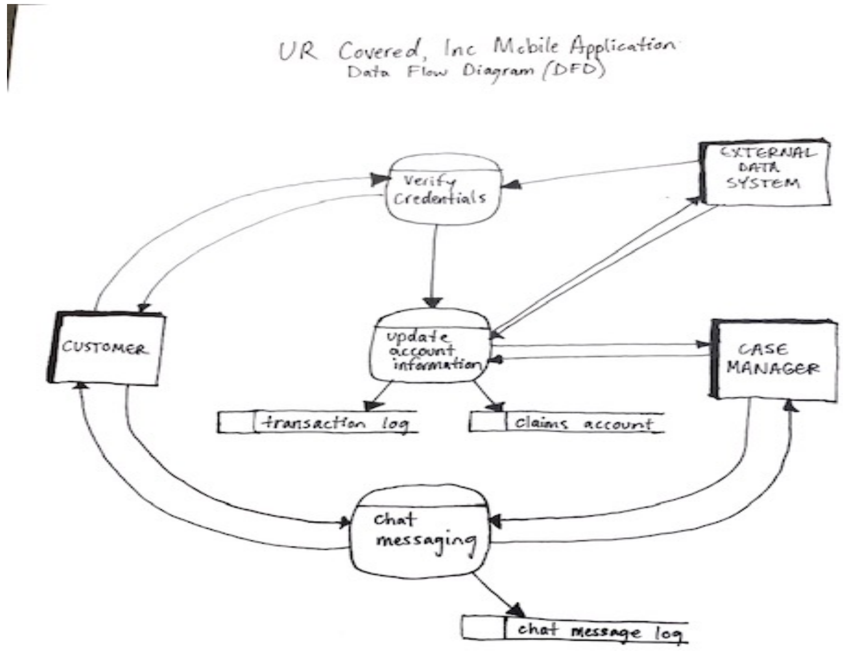
- All updates and submissions must have audit trails (Rosenblatt, 2017).

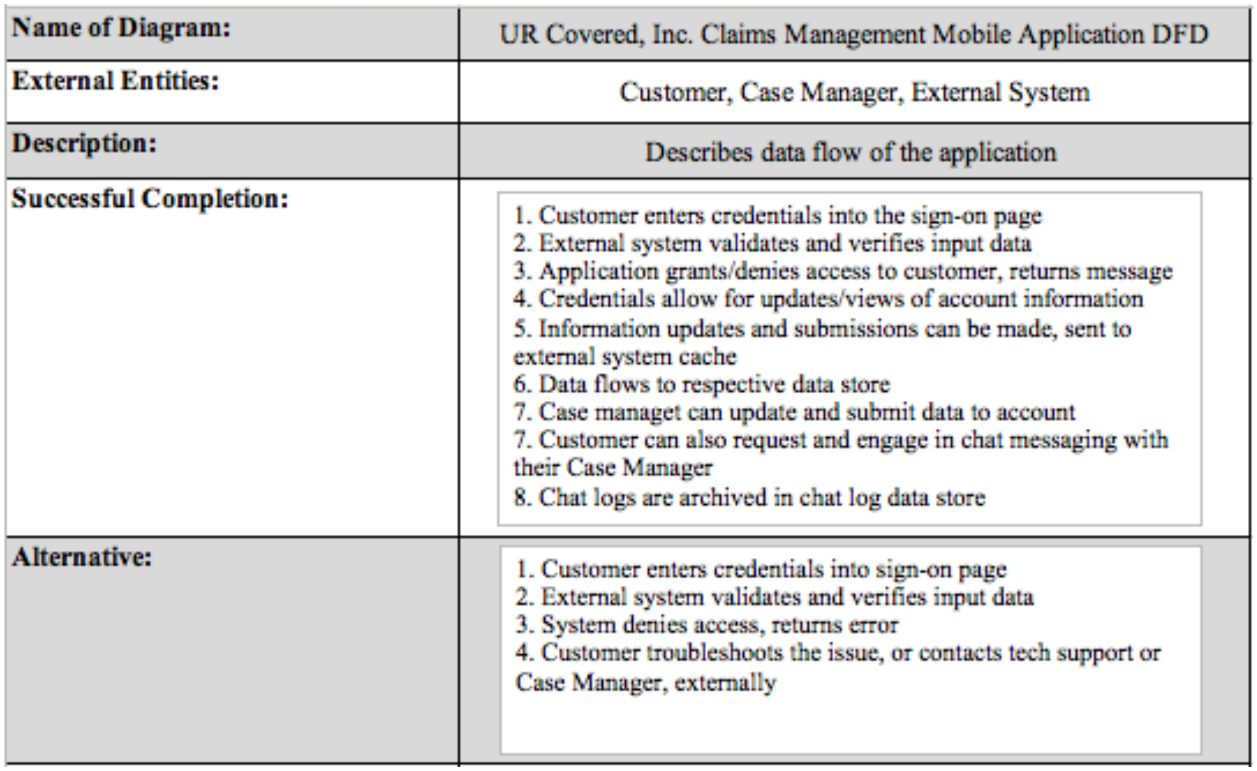
- Data entry (login credentials) must create an error log file that includes error type, description (Rosenblatt, 2017) and a threshold value for locked account purposing

- System must produce a daily activity report and log member account status

3.2 Data Processing Model

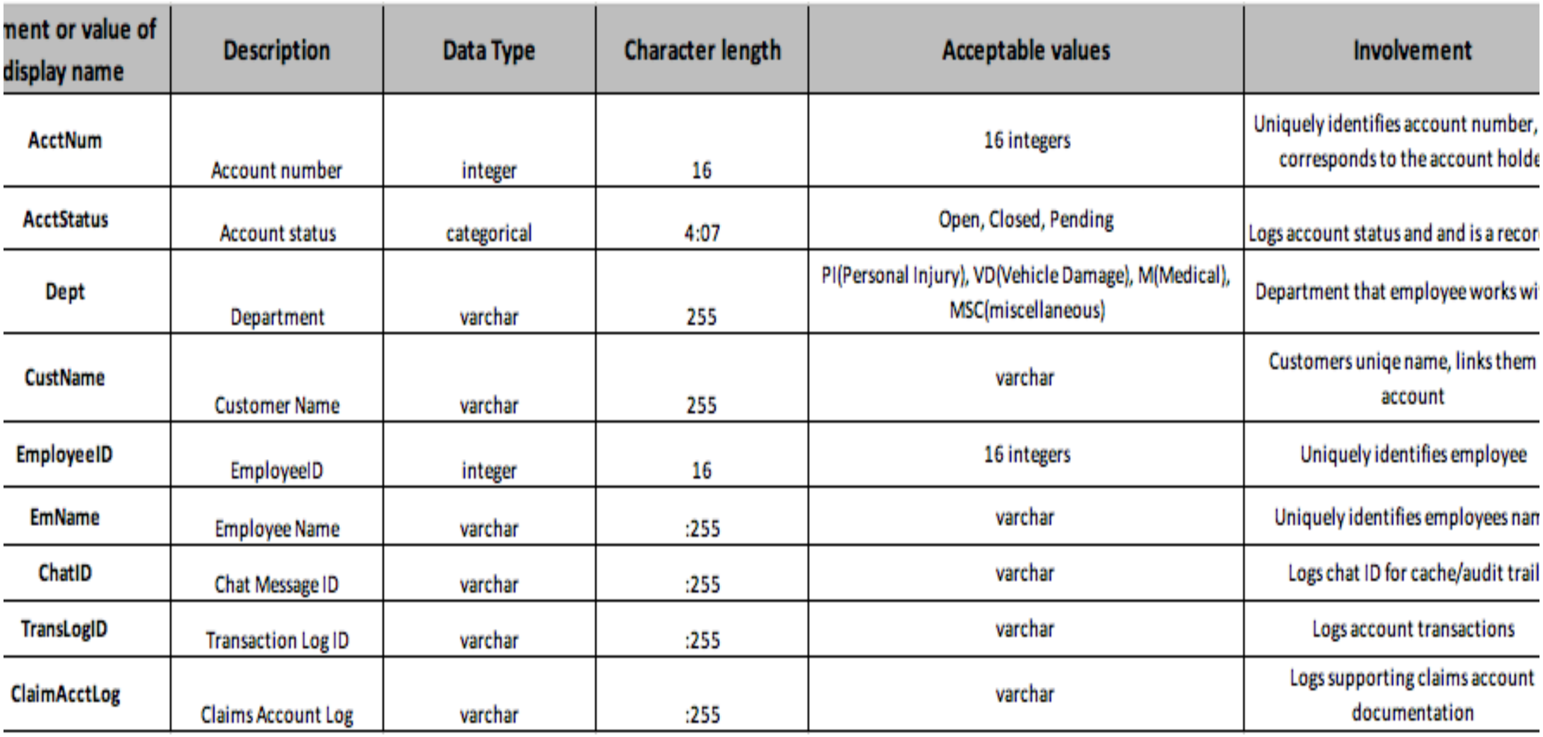
3.2.1 Data Flow Diagram (DFD)





Name of Diagram:	UR Covered, Inc. Claims Management Mobile Application DFD
External Entities:	Customer, Case Manager, External System
Description:	Describes data flow of the application
Successful Completion:	1. Customer enters credentials into the sign-on page 2. External system validates and verifies input data 3. Application grants/denies access to customer, returns message 4. Credentials allow for updates/views of account information 5. Information updates and submissions can be made, sent to external system cache 6. Data flows to respective data store 7. Case managet can update and submit data to account 7. Customer can also request and engage in chat messaging with their Case Manager 8. Chat logs are archived in chat log data store
Alternative:	1. Customer enters credentials into sign-on page 2. External system validates and verifies input data 3. System denies access, returns error 4. Customer troubleshoots the issue, or contacts tech support or Case Manager, externally

3.3 Data Dictionary



ment or value of display name	Description	Data Type	Character length	Acceptable values	Involvement
AcctNum	Account number	integer	16	16 integers	Uniquely identifies account number, corresponds to the account holde
AcctStatus	Account status	categorical	4:07	Open, Closed, Pending	Logs account status and and is a recor
Dept	Department	varchar	255	PI(Personal Injury), VD(Vehicle Damage), M(Medical), MSC(miscellaneous)	Department that employee works wi
CustName	Customer Name	varchar	255	varchar	Customers uniqe name, links them account
EmployeeID	EmployeeID	integer	16	16 integers	Uniquely identifies employee
EmName	Employee Name	varchar	:255	varchar	Uniquely identifies employees nam
ChatID	Chat Message ID	varchar	:255	varchar	Logs chat ID for cache/audit trail
TransLogID	Transaction Log ID	varchar	:255	varchar	Logs account transactions
ClaimAcctLog	Claims Account Log	varchar	:255	varchar	Logs supporting claims account documentation

Figure 3.3 Data dictionary. The data dictionary above presents the proposed, or possible utilization of data and its metadata information. Specific to claims accounts and activity, this data dictionary is simple and limited to the scope of its use. The metadata is mirrored after currently established data dictionary information within the current, external UR Covered, Inc. databases.

3.4 Object Modeling

3.4.1 Class Diagram

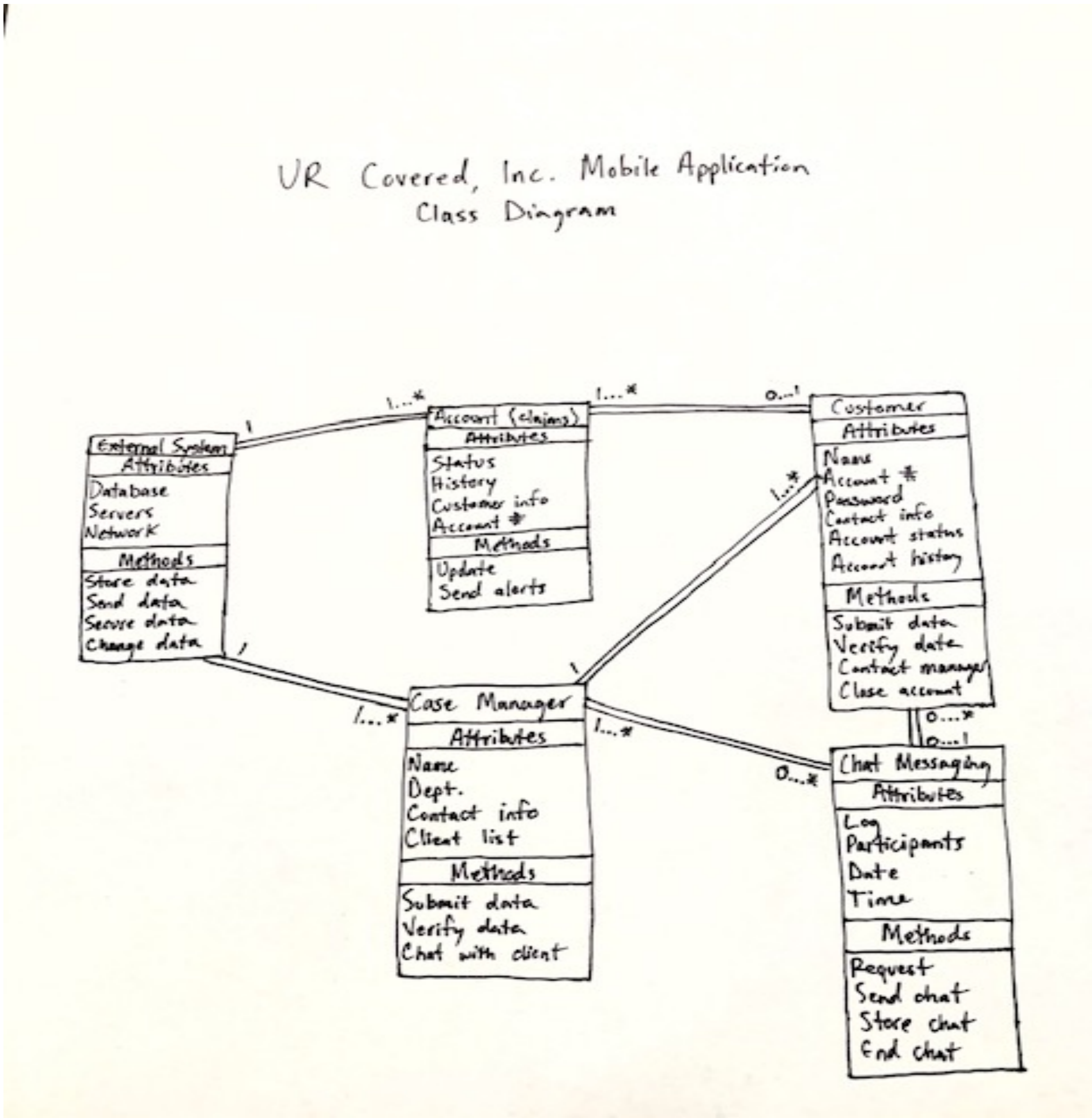
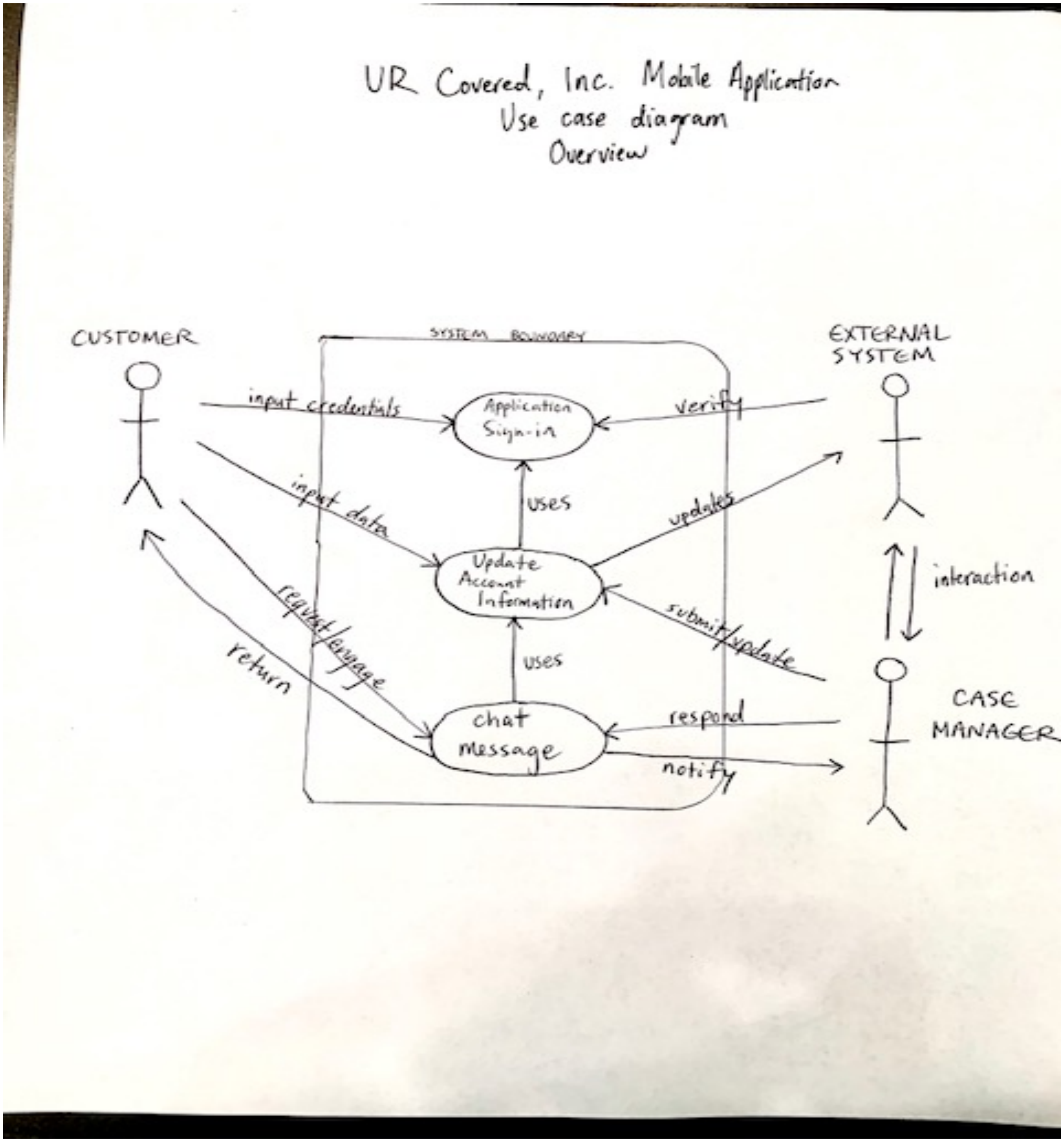
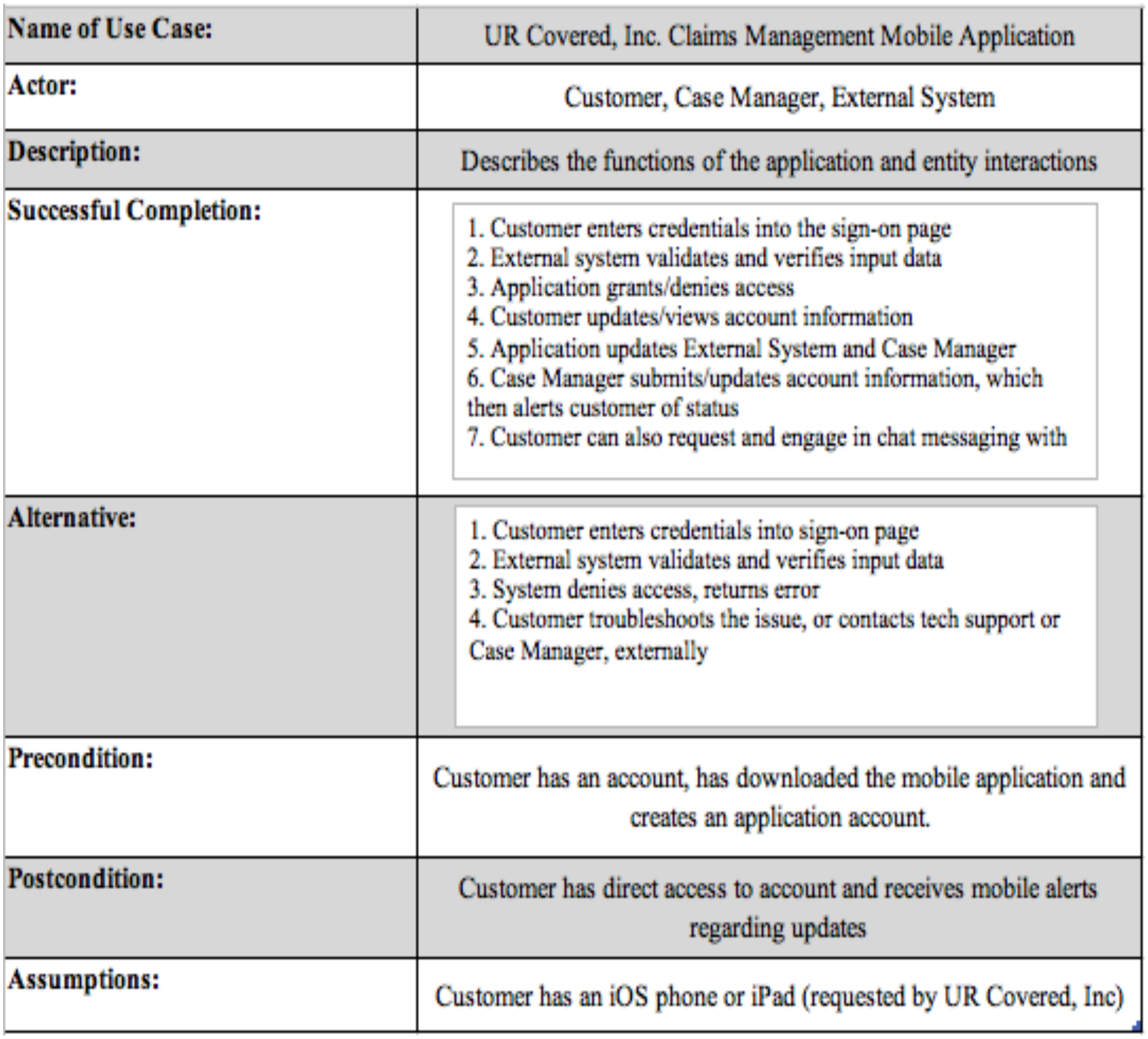


Figure 3.4.1 Class Diagram. This class diagram displays the specific attributes and methods of objects involvement in the utilization and functionality of the mobile application. For example, take the Case Manager object, with attributes (Name, Dept., contact info, client list) and methods (submit data, verify data, chat with client) and its relationship with the Customer object, its' with attributes (Name, Account #, Password, contact info, account status, account history) and methods (submit data, verify data, contact manager, close account). A Case Manager can have one to many customers ('1…*'), while the Customer can only have one Case Manager ('1').

3.4.2 Use Case Diagram





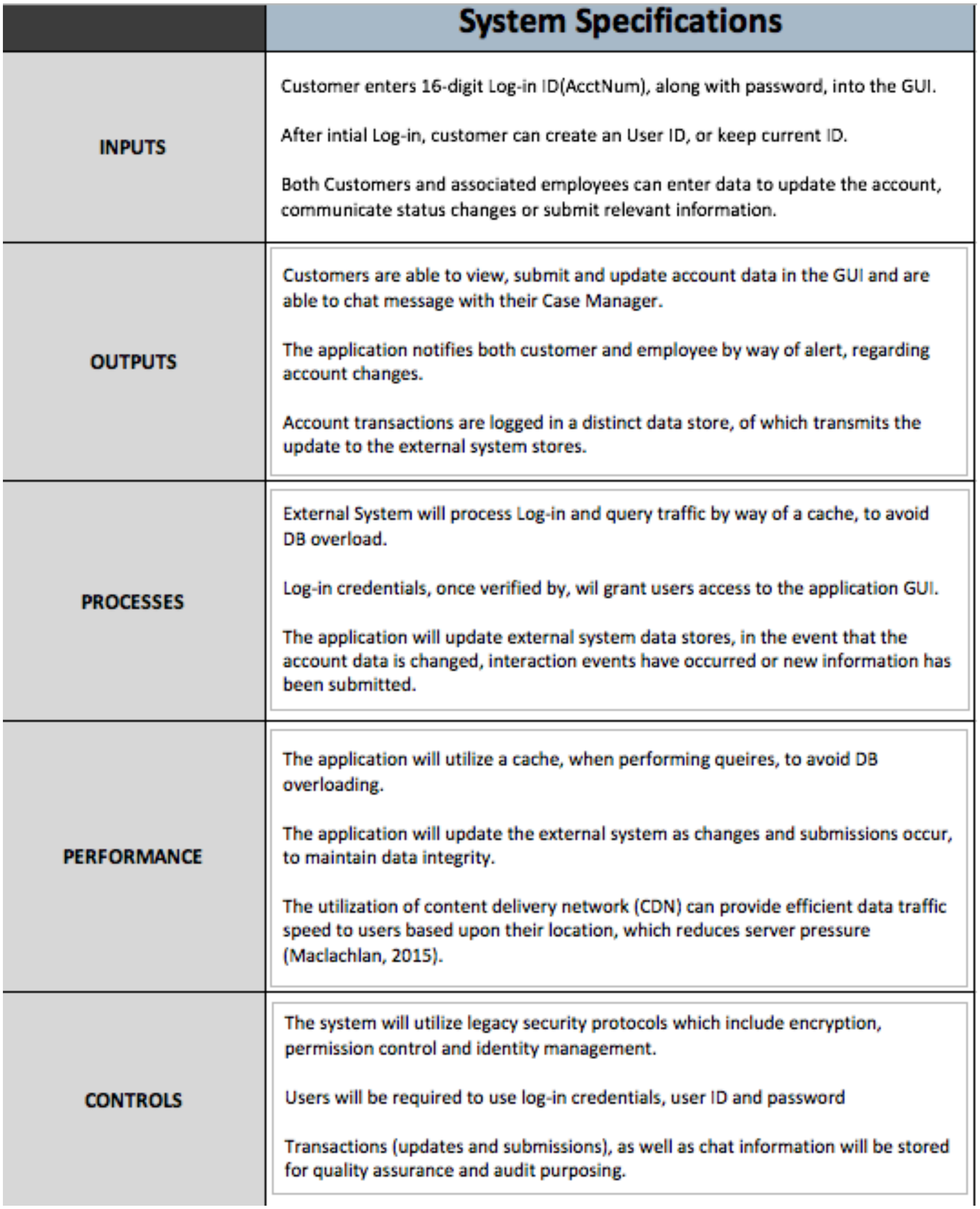
Name of Use Case:	UR Covered, Inc. Claims Management Mobile Application
Actor:	Customer, Case Manager, External System
Description:	Describes the functions of the application and entity interactions
Successful Completion:	1. Customer enters credentials into the sign-on page 2. External system validates and verifies input data 3. Application grants/denies access 4. Customer updates/views account information 5. Application updates External System and Case Manager 6. Case Manager submits/updates account information, which then alerts customer of status 7. Customer can also request and engage in chat messaging with
Alternative:	1. Customer enters credentials into sign-on page 2. External system validates and verifies input data 3. System denies access, returns error 4. Customer troubleshoots the issue, or contacts tech support or Case Manager, externally
Precondition:	Customer has an account, has downloaded the mobile application and creates an application account.
Postcondition:	Customer has direct access to account and receives mobile alerts regarding updates
Assumptions:	Customer has an iOS phone or iPad (requested by UR Covered, Inc)

Figure 3.4.2 Use Case Diagram Description. This figure displays the functions of the application and specific entity relationships as it relates to the functionality of the user interface.

4.0 SYSTEM DESIGN

The UR Covered, Inc., claims management mobile application has been developed by establishing the layout of the system, how the entities within the system interact with one another, and the scope of functionality and utility for both customers and employees. The design of the system, outlined in this document, has transformed the aforementioned logical models, into physical models. The design of the system includes the specification models, data design, user interface design, system architecture and the feasibility analysis conclusion. The specification of the design covers inputs, outputs, processes, performance and controls. Data design is displayed through the entity relationship diagram (Figure 4.2). The user interface design is displayed through a proposed form design (Figures 4.3-4.3.3) and will include three distinct graphical user interface (GUI) pages, of which can be accessed through the main Account interface, once logged in. The system architecture includes software and network support overview, corporate organization culture, enterprise resource planning, total cost of ownership, scalability, integration and interface requirements and security. The feasibility analysis includes the operational, technical, economic and scheduling feasibility of the project.

4.1 System Specifications

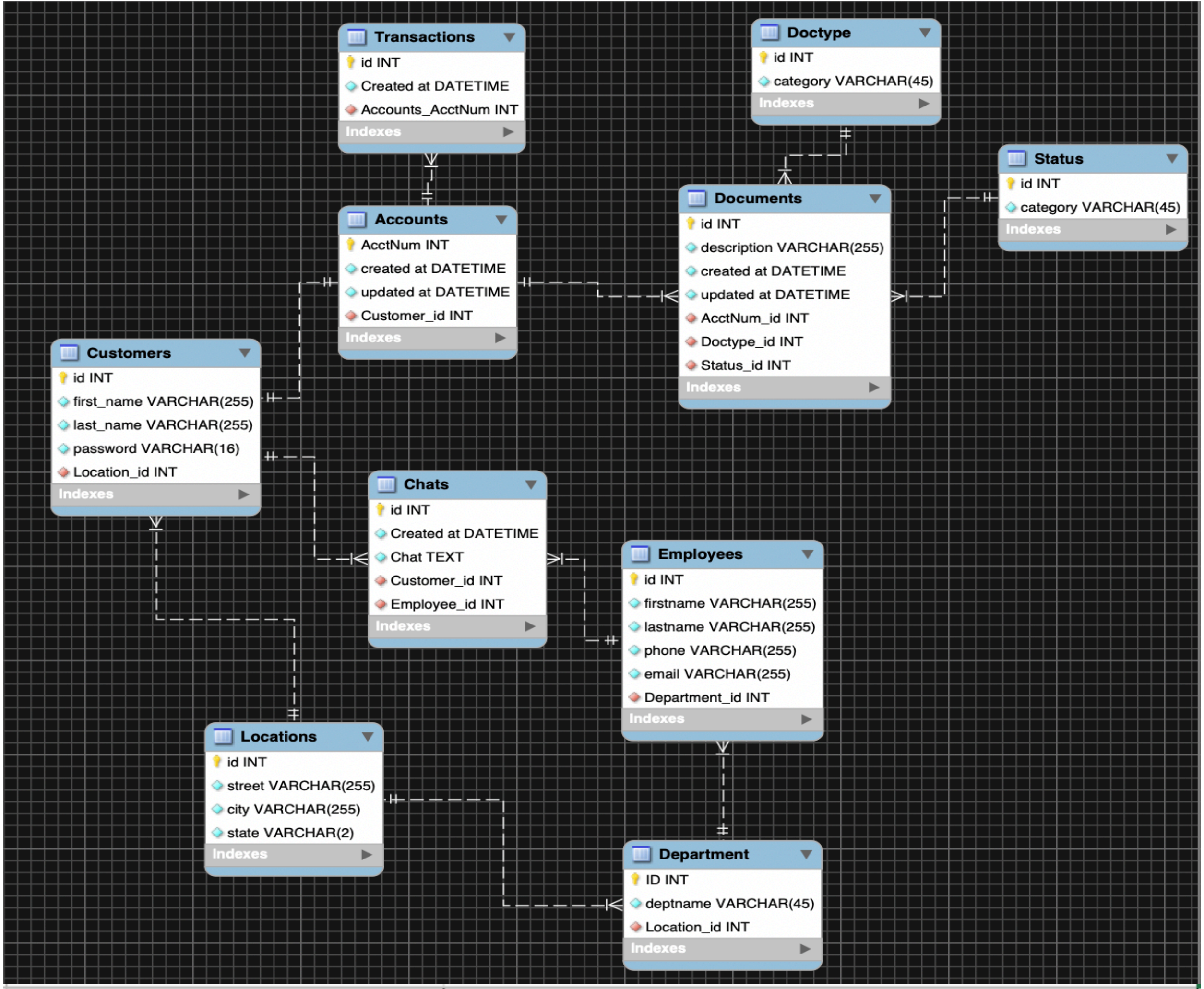


	System Specifications
INPUTS	Customer enters 16-digit Log-in ID(AcctNum), along with password, into the GUI. After intial Log-in, customer can create an User ID, or keep current ID. Both Customers and associated employees can enter data to update the account, communicate status changes or submit relevant information.
OUTPUTS	Customers are able to view, submit and update account data in the GUI and are able to chat message with their Case Manager. The application notifies both customer and employee by way of alert, regarding account changes. Account transactions are logged in a distinct data store, of which transmits the update to the external system stores.
PROCESSES	External System will process Log-in and query traffic by way of a cache, to avoid DB overload. Log-in credentials, once verified by, wil grant users access to the application GUI. The application will update external system data stores, in the event that the account data is changed, interaction events have occurred or new information has been submitted.
PERFORMANCE	The application will utilize a cache, when performing queires, to avoid DB overloading. The application will update the external system as changes and submissions occur, to maintain data integrity. The utilization of content delivery network (CDN) can provide efficient data traffic speed to users based upon their location, which reduces server pressure (Maclachlan, 2015).
CONTROLS	The system will utilize legacy security protocols which include encryption, permission control and identity management. Users will be required to use log-in credentials, user ID and password Transactions (updates and submissions), as well as chat information will be stored for quality assurance and audit purposing.

Figure 4.1: System specifications

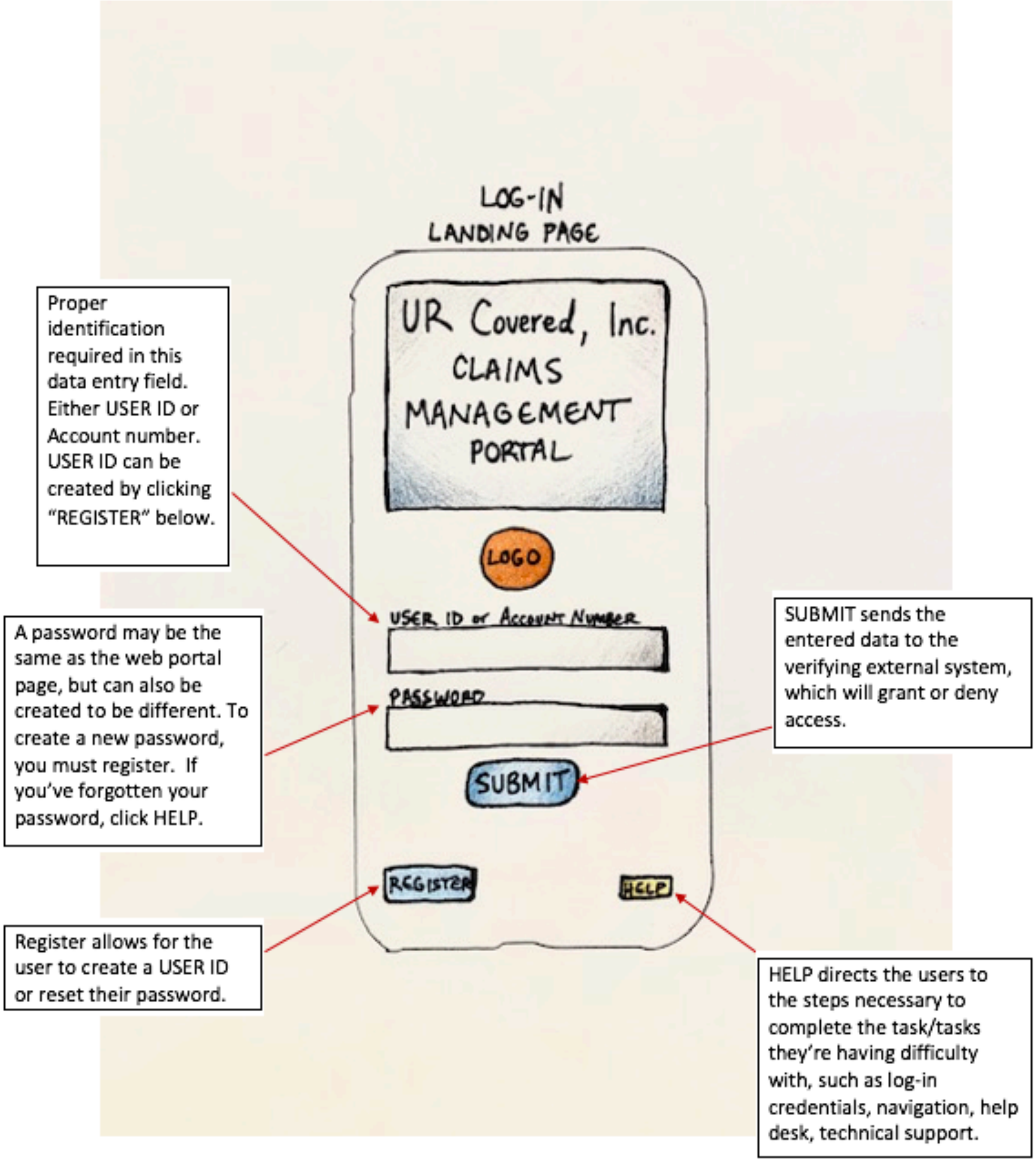
4.2 Data Design

Figure 4.2: UR Covered, Inc., Mobile Application Entity Relationship Diagram



Name of Entity Realtionship Diagram:	UR Covered, Inc. Claims Management Mobile Application
Entities:	Customers, Locations, Accounts, StatusTransactions, Chats, Employees, Department
Description:	Describes the relationships between application system entities
Entity Relationships:	1. Customers-Locations = many-to-one relationship 2. Customers-Accounts = one-to-one relationship 3. Accounts-Transactions = one-to-many relationship 4. Customers-Chats = one-to-many relationship 5. Accounts-Documents = one-to-many relationship 6. Documents-Doctype = many-to-one relationship 7. Documents-Status = many-to-one relationship 8. Chats-Employees = many-to-one relationship 9. Employees-Department = many-to-one relationship 10. Department-Locations = many-to-one relationship

4.3 User Interface Design (1/3)



Proper identification required in this data entry field. Either USER ID or Account number. USER ID can be created by clicking "REGISTER" below.

A password may be the same as the web portal page, but can also be created to be different. To create a new password, you must register. If you've forgotten your password, click HELP.

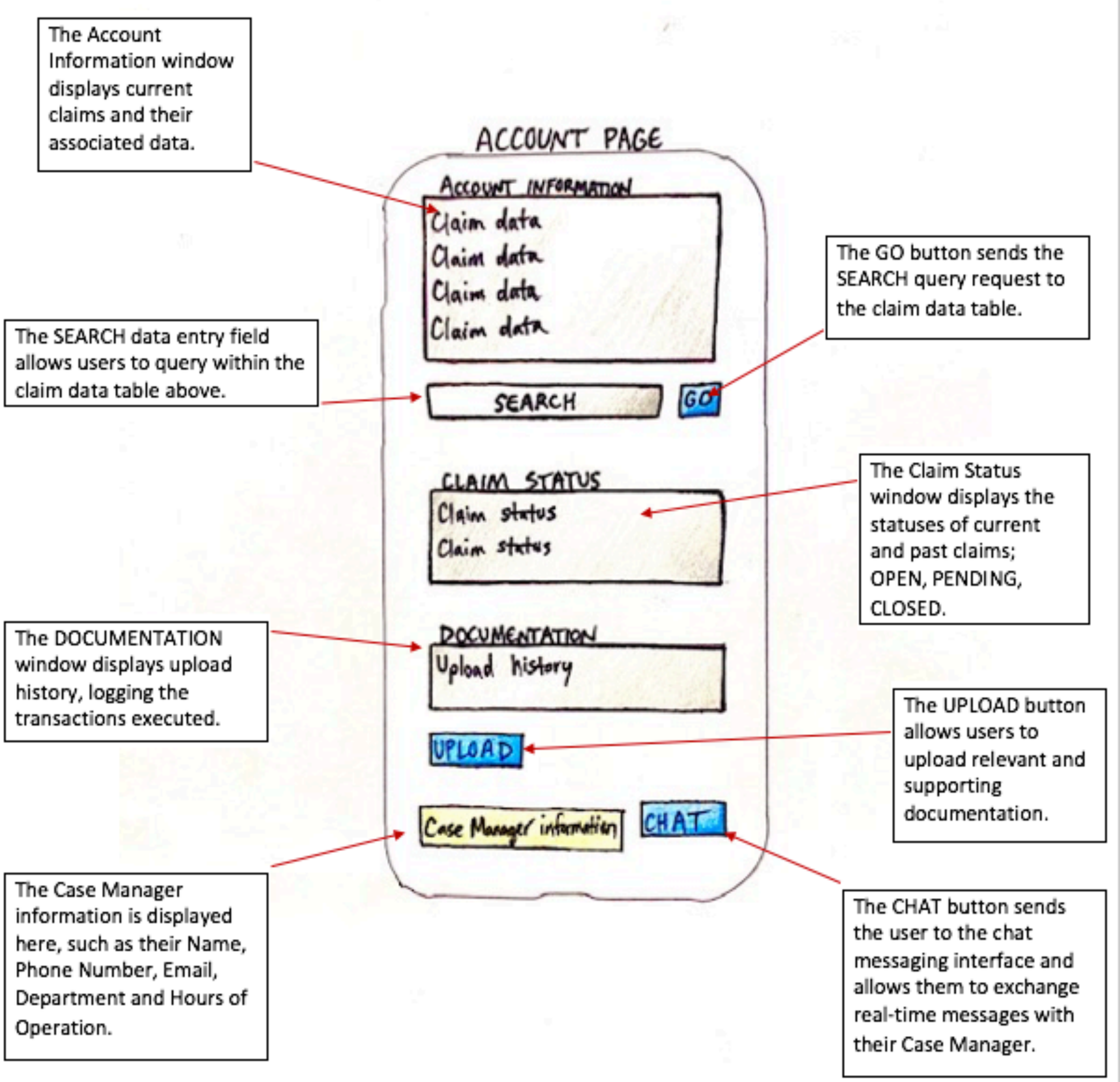
Register allows for the user to create a USER ID or reset their password.

SUBMIT sends the entered data to the verifying external system, which will grant or deny access.

HELP directs the users to the steps necessary to complete the task/tasks they're having difficulty with, such as log-in credentials, navigation, help desk, technical support.

Figure 4.3.1: Graphical User Interface: The image above shows the log-in landing page of the mobile application, as well as describes the functions of features and the intended form layout of that page.

4.3.2 User Interface Design (2/3)



The Account Information window displays current claims and their associated data.

The GO button sends the SEARCH query request to the claim data table.

The SEARCH data entry field allows users to query within the claim data table above.

The Claim Status window displays the statuses of current and past claims; OPEN, PENDING, CLOSED.

The DOCUMENTATION window displays upload history, logging the transactions executed.

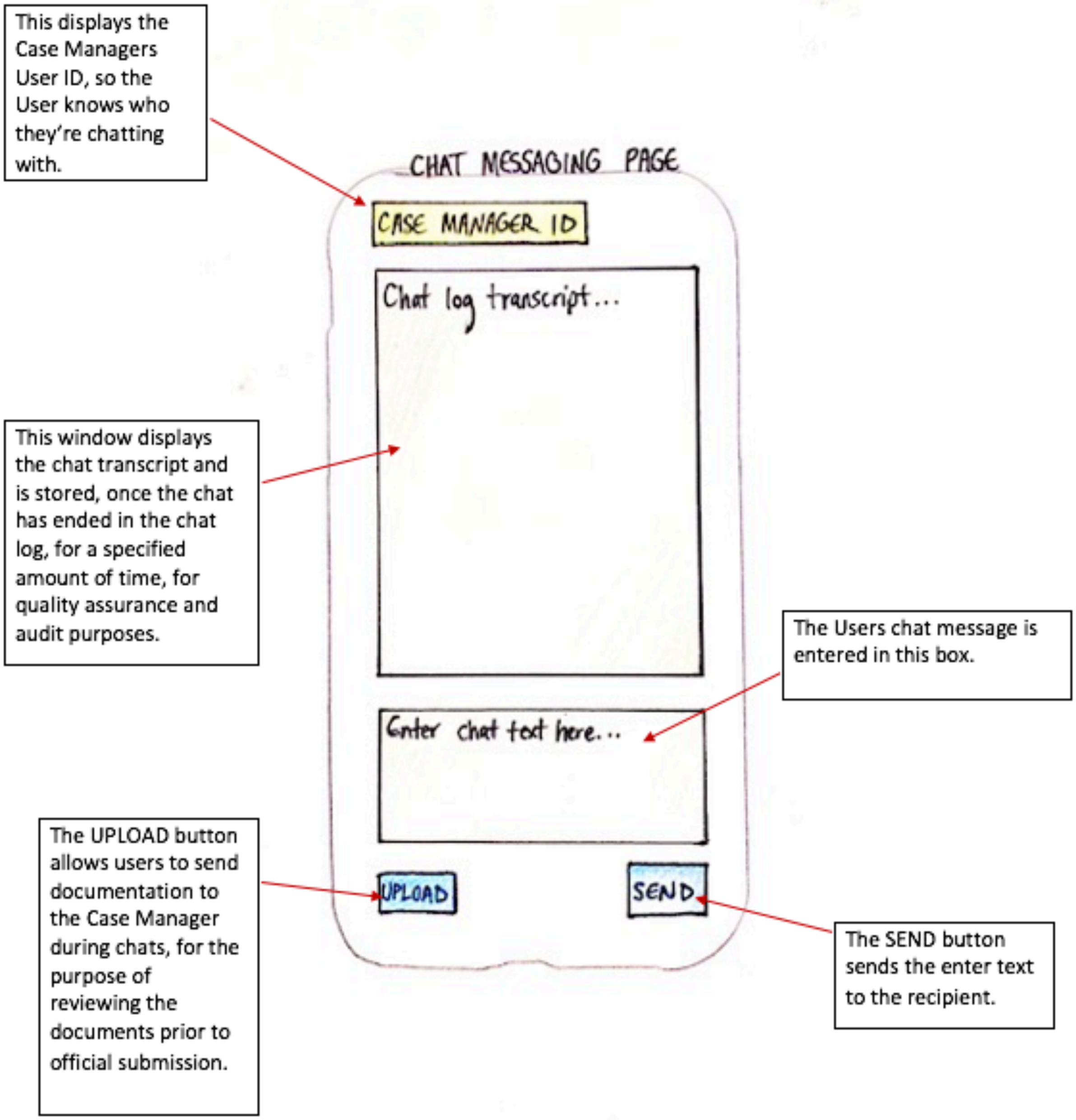
The UPLOAD button allows users to upload relevant and supporting documentation.

The Case Manager information is displayed here, such as their Name, Phone Number, Email, Department and Hours of Operation.

The CHAT button sends the user to the chat messaging interface and allows them to exchange real-time messages with their Case Manager.

Figure 4.3.2: Graphical User Interface: The image above shows the Account page of the mobile application, displayed after being granted access to the application through the log-in page. The Account page has several features and functions of which allows the user to interact with legacy data, as well as submit documentation and request chat messaging.

4.3 User Interface Design (3/3)



This displays the Case Managers User ID, so the User knows who they're chatting with.

This window displays the chat transcript and is stored, once the chat has ended in the chat log, for a specified amount of time, for quality assurance and audit purposes.

The Users chat message is entered in this box.

The UPLOAD button allows users to send documentation to the Case Manager during chats, for the purpose of reviewing the documents prior to official submission.

The SEND button sends the enter text to the recipient.

Figure 4.3.3: Graphical User Interface: The image above shows the chat messaging page, accessed through the Account page. The page displays the Case manager identifying information, the text transcript as well as the users text and the ability to upload data and documentation.

4.4 System Architecture

Software – The mobile application software requires several different phases throughout the development process and will deliver a graphical user interface (GUI), which displays the system in a visual and interaction manner. Users will download the application, log-in with authorized credentials on the landing page, then will navigate through the features of the interface based upon their needs. The software will be capable of performing data queries, documentation uploads and chat messaging.

Network support – In addition to the creation of the mobile application software, a memory cache and content delivery network (CDN) will be utilized to assist its functionality. A memory cache is an area of local memory that holds a copy of frequently accessed data that is otherwise expensive to get or compute (Shah, 2017). A CDN is a network of dispersed servers that deliver content to users based upon their geographic locations (MacLachlan, 2015). These assets currently exist within the organization and are being utilized for web application support, but incorporating them into the mobile application deployment and implementation, will bolster its ability to provide its users with the speed and reliability expected.

Corporate Organization and culture – UR Covered, Inc. has established itself as an organization that holds reputable relationships and excellent service standards with their customers. The claims management mobile application will enhance customer trust, expand their ability in terms of account processing cycle involvement and adds a layer of transparency to the organizations functions.

Enterprise Resource Planning (ERP) – The current Enterprise Resource Planning program involves a centralized organizational data base, with several database charts and caches, use to distribute database traffic loads. These same databases, with the addition of a cache specifically for the mobile application, will deliver legacy information and relevant metadata regarding specific claims account information queried through the mobile application platform.

Initial and total cost of ownership (TCO) – In-house development was initially proposed for this system and that recommendation stands. Due to this system being an organizational expansion, employee resources will need to be allocated towards the implementation, deployment and maintenance phase efforts. Due to this project being in-house developed, time will need to be spent programming and linking the system into the legacy system, of which will be carried out by the aforementioned personnel. The cost breakdown is displayed in Appendix C, shows the initial cost for application development and implementation. These initial costs, based upon current market and industry environments, remains realistic and the addition of maintenance and software update costs will need to be calculated once the utility and functionality of the application is assessed post-deployment. Such maintenance and support costs are estimated to be 15-20% of the initial investment (Mobile App Daily, 2019), in this case, based upon the $513,200 investment, that would equate to a range of $76,980 - $102,640. These costs may fluctuate based upon the significance of the updates required, such as managing application content and memory, updating desired features, analyzing usage and effectiveness, as well as maintaining the incoming traffic and security levels required.

Scalability – Due to the nature of the mobile application development costs and its foreseen implementation and nature of utility, scalability or extensibility are abilities that this application must be able to complete smoothly. Whether the utility of the application is overwhelming positive or lacking in benefit to the users, the system must be able to adjust to such conditions, in order to reduce costs, relocate resources and find suitable areas of improvement to support business models and customer expectations. For example, if customers are not utilizing the chat feature of the application, the flexibility of the design of the system, allows for the removal of the feature. If customers are utilizing the documentation upload feature more often than originally thought, and the feedback is requesting more robust and interactive features, then those expansive efforts can be made with minimal changes to the current architecture and layout. To handle such fluctuations, a content delivery network (CDN) will be imperative to distributing high speed responsiveness to customers as the user population increase. Thankfully, due to the versatility of a CDN, costs and robustness will change based upon the needs of the traffic, which minimizes unnecessary spending and unused network and application space.

Web Integration – The organization currently utilizes a web application for customer Internet traffic, however, the mobile application will be unique in nature, as it is specifically gauged to handle and include only claims content management. This application is not a mobile version of the current web application system, but is rather minimal in size and specific to its purpose to streamline processing and create an easy-to-use environment for the customers.

Legacy system interface requirements – Currently, the main organizational data system meets the requirements of their business processes, but the mobile application will offer an extension of such functions. Typically, middleware is positioned as a bridge between legacy systems and new systems, due to having to convert legacy file data information (Rosenblatt, 2017). However, the mobile application will utilize data, as well associated metadata, from the main system, which eliminates the need for middleware.

Processing options – The mobile application will utilize online processing, as opposed to batch processing, because online systems handle transactions when and where they occur and provide output directly to users (Rosenblatt, 2017). The type of processing is currently the processing method used by the organization and will extend its reach to the mobile system, as customers need quick and reliable responsiveness from the application interface and its associated data assets. Online processing avoids delays and allows a constant dialog between the user and the system (Rosenblatt, 2017).

Security – Organizational security policies in place will extend to include the mobile application system. The policy has been established upon the three elements of system security concepts, of which include confidentiality, integrity and availability (Rosenblatt, 2017). To uphold data confidentiality, the network will continue to encrypt data assets to maintain the privacy of sensitive information. In addition to encryption, the application will utilize proper permission protocol, ensuring authorized users can access key information, as well as identity management, which authenticates user credentials, such as user IDs and associated passwords (Rosenblatt, 2017). Maintaining the integrity of the system will prevent unauthorized users from creating, modifying or deleting information (Rosenblatt, 2017) while the availability of data provides users with timely and accurate access to data assets.

The security of the system will protect sensitive and valuable information by identifying threats and and following the established standards of security when deterring and mitigating any and all threats.

4.5 Feasibility Analysis

Operational feasibility – Based upon management decision to pursue the development of the mobile application, the anticipated impact of the software and its utility, in their eyes, brings great value and satisfaction to customers. Customers are well pleased with the service provided to them by UR Covered and the features included within the mobile application will extend their ability to interact and engage in the claims management process. By validating the need for the mobile application, the claims management process will incorporate a more inclusive environment, where the customers and Case managers are able to more transparently communicate and conduct the necessary steps to complete the tasks at hand. Due to the mobile application being an extension of company service, the workforce will need to be either allocated to its operational needs or workforce members will need to increase. The infrastructure required to handle the new demands has been discussed above and the changes required will help mitigate any performance issues or constraints.

Economic feasibility – The application has been proposed to be an in-house development, to reduce costs, familiarize staffing with the structuring and functionality of the system and to maintain data confidentiality. However, given the initial costs of development ($513,200) and the reoccurring yearly costs of maintenance and support ($76,980 - $102,640), compared to the functionality and benefits of the application itself, the benefits will need to be assessed and considered based upon user feedback and the rank of utility and functionality. The tangible costs for staffing, software, training, fees, as well as intangible costs, such as employee morale during the development phases and maintenance, are robust in nature and require significant consideration (Rosenblatt, 2017). The benefits of the application include reducing customer-to-support phone traffic and the intangible benefits include possible increased customer satisfaction, as they have more transparent access to their account information. In addition to the costs versus benefits analysis, the time period required for development and maintenance of such an application covers a 22-week development period, with an indefinite maintenance period, of which neither seem to match the anticipated impact of such developmental effort. However, if user feedback establishes that the application is proving to be of substantial benefit to the community and customer experience, in addition to the organization accepting the costs associated with the application, the application will prove to be of economic and operational benefit to the organization.

Technical feasibility – The organization has established that it does in fact have the necessary network and software resources and, or the ability to procure such assets. The technical expertise required will span across various levels of required skills and abilities, and those particular individuals have been allotted towards the creation of the mobile application, details are laid out in the WBS. The required capacity for the implementation and maintenance of the application will increase as the usage fluctuates and content and protocol updates are carried out. A prototype phase has been included in the projects life cycle and involves all project teams and stakeholders, as any changes or validations to be made, should be reflected in the concurrences of the respective project area team members.

Schedule feasibility – As previously mentioned, the project is projected to take 22-weeks. The Gannt chart, displayed in previous reports, has structured the project so that, if there are any tasks that can be done in conjunction with others, it has been scheduled as so. By completing the tasks at the earliest moment possible, as per the time schedule, costs are reduced and resources can be redistributed towards the needs organizational operations. Completing tasks ahead of schedule presents no issues, as long as the requirements of that task have been met prior to moving on to the next task. To ensure such standards of completion are upheld throughout each task and ultimately the entire project life, weekly reports will be generated, which cover the status of the applicable task and relevant information, and will be sent to the necessary stakeholders, the project managers specifically. Monitoring checkpoints, carried out by the project manager, will occur at the discretion of the individual, as to assess the progress of the project and address any issues or concerns that have arisen. These practices help maintain a focused workforce, and help make the necessary changes to the work being done, to deliver the task deliverable successfully, and on time. These principles and measures are more deeply explained in the project monitoring and control plan.

5.0 PROJECT PLAN

UR Covered, Inc. desires a mobile application that will not only enhance customer satisfaction, but will also improve business relations between the customers and case managers. The application will provide them with an additional resource that will be utilized for more timely account information and claim management process. The process of developing and implementing such an application will require effective design, management, project monitoring and control, resource allocation, budgeting and communication.

The final deliverable will be a User Interface (UI) mobile application. A user interface (UI) describes how users interact with a computer system, and consists of all the hardware, software, screens, menus, functions, output, and features that affect two-way communications between the user and the computer. The UI is the key to usability, which includes user satisfaction, support for business functions, and system effectiveness (Rosenblatt, 2017).

This project is broken down into five project event phases. The five stages resemble the Systems Development Life Cycle (SDLC) and includes design, development, testing, implementation and evaluation.

5.1 Work Breakdown Structure (WBS)

The work breakdown structure (WBS) communicates all project events, task sequencing and durations, costs, resource allocations and highlights specific roles and functions for each task team.

Team Roles and Functions

The project will require seven teams. The project teams consist of two User Interface (UI) teams, two Database teams, a User Support team, Security team and a Quality Assurance (QA) team. Each team has has been assigned tasks and have their own unique work schedule.

UI and Database team

The UI and Database teams will be responsible for the development and documentation of system components, form layout design, group functions and menu navigation features and data entry forms. "System components contains the complete design for the new system, including the user interface, outputs, inputs, files, databases, and network specifications. Source documents, report and screen layouts, DFDs, and all other relevant documentation should be included" (Rosenblatt, 2017). Form layout design determines the visual aesthetics and arrangement of interface zones and functions. Group functions and menu navigation features consolidate action items into a multilevel menu hierarchy (Rosenblatt, 2017), making the application more easily traversed and user intuitive. Data entry forms are the user input fields, that allow the user to enter, correct, update information into the application and how the requirements for such inputs must be understood by the source or system interpreter.

These are the tasks ad assigned responsibilities for the UI and Database teams. These tasks are the first four phase events for the project.

User Support team

The User Support team will be responsible for user assistance and technical support, both of which help maintain and improve user satisfaction and application utility. It's important that User Support team members be included during the development process of the project, as it will provide them with a detailed and in-depth awareness and knowledge of the system, ultimately strengthening their ability to carry out their technical duties.

Security team

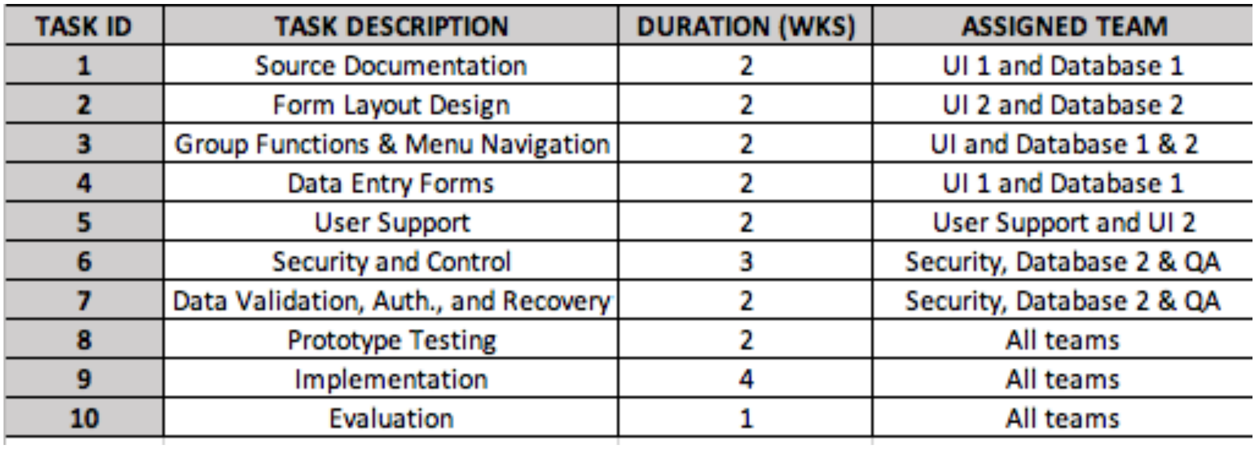
The Security teams' responsibilities are large and critical to the success of the project and the trustworthiness of the application. Security levels and protocols will mirror that of the web application. It is imperative that the team understands the organizations current security policy, of which should be based upon the three elements of security: confidentiality, integrity and availability (Rosenblatt, 2017).

Elements from the current structuring include security ID assignments and identity management, data encryption, network security, data recovery and many others. This is a robust project event, but due to this project being a transfer of infrastructure from an existing application, it will boil down to attention to detail and meticulous methodology.

Quality Assurance team

The role and function of the Quality Assurance (QA) team is identify and help avoid problems as soon as possible (Rosenblatt, 2017). The QA team will assist not only the security team during their project event tasks, but help review and amend any necessary developments completed by predecessor project events. By incorporating such a team, the project will be able to avoid inaccurate requirements, design issues, coding errors, faulty documentation and ineffective iteration testing (Rosenblatt, 2017).

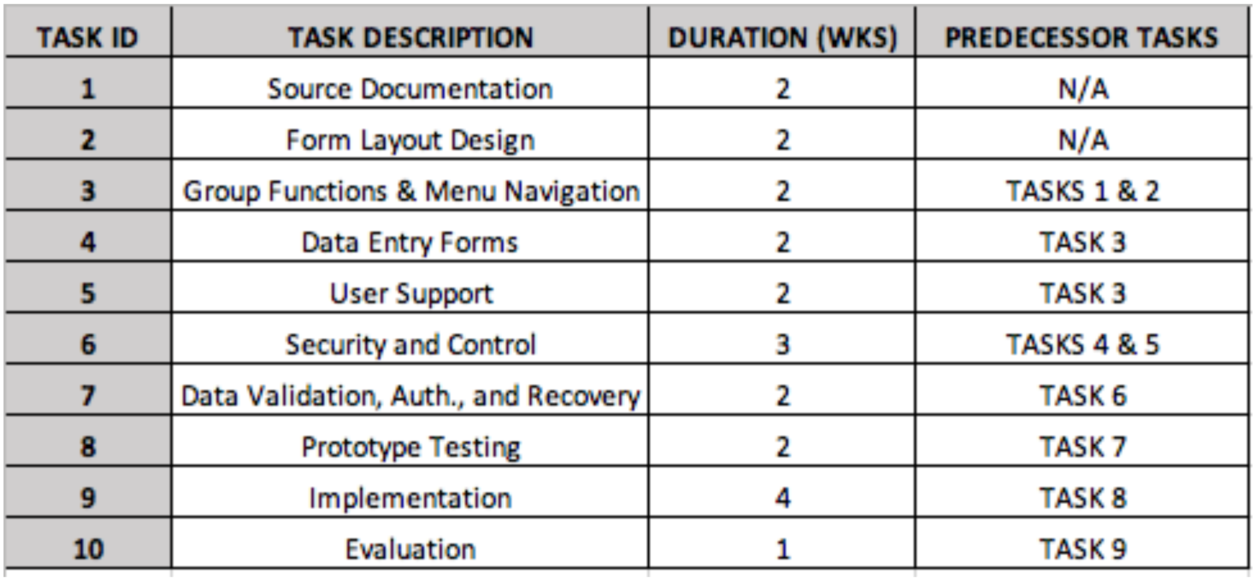
Project Duration and Task Assignments (Figure 5.1.1)



TASK ID	TASK DESCRIPTION	DURATION (WKS)	ASSIGNED TEAM
1	Source Documentation	2	UI 1 and Database 1
2	Form Layout Design	2	UI 2 and Database 2
3	Group Functions & Menu Navigation	2	UI and Database 1 & 2
4	Data Entry Forms	2	UI 1 and Database 1
5	User Support	2	User Support and UI 2
6	Security and Control	3	Security, Database 2 & QA
7	Data Validation, Auth., and Recovery	2	Security, Database 2 & QA
8	Prototype Testing	2	All teams
9	Implementation	4	All teams
10	Evaluation	1	All teams

Figure 5.1.1: Task IDs have been assigned to project tasks, along with their respective durations and the teams assigned to carry out the requirements of such tasks.

Project Task Sequencing (Figure 5.1.2)



TASK ID	TASK DESCRIPTION	DURATION (WKS)	PREDECESSOR TASKS
1	Source Documentation	2	N/A
2	Form Layout Design	2	N/A
3	Group Functions & Menu Navigation	2	TASKS 1 & 2
4	Data Entry Forms	2	TASK 3
5	User Support	2	TASK 3
6	Security and Control	3	TASKS 4 & 5
7	Data Validation, Auth., and Recovery	2	TASK 6
8	Prototype Testing	2	TASK 7
9	Implementation	4	TASK 8
10	Evaluation	1	TASK 9

Figure 5.1.2: Task sequencing has been established by evaluating which tasks will be conducted together, as well as their predecessors. Further sequencing details are included in the PERT chart, **Figure 5.2.1**.

5.2 Project Monitoring and Control Plan

Project phases have been determined and partitioned into task iterations, which ensures that each task meets the business requirements of the project, prior to solidifying its interim deliverables. Tasks are basic units of work that the project manager plans, schedules, and monitors (Rosenblatt, 2017). Each event will be subject to a task timeline, along with progress reports and monitoring checkpoints. Reports and updates will be delivered weekly, through company approved correspondence, while monitoring checkpoints will be done in person, at the discretion of the project manager, typically at the end of each event.

Teams are assigned to specific project events and their respective tasks; several events in this project have teams working together to complete those tasks and events. Each team will be assigned a team leader, whom is responsible for task management and tracking, as well as timely communication during status updates and task delivery windows. Every team member and team lead must have a solid understanding of mitigation solutions and the necessary actions required to address such issues. By adhering to established project guidelines, each event and its tasks will be completed successfully and efficiently, bringing assurance that the project will complete on time and within its proposed budget.

The projects critical path is a series of tasks that, if delayed, would affect the completion date of the overall project (Rosenblatt, 2017). The PERT chart displays the order of the proposed work tasks and as is, the critical path is identified as being all tasks being proposed. Due to multiple tasks being conducted simultaneously, each task itself is along the critical path of the project; delay in any activity will affect the completion of the project and the respective task dates and duration.

PERT chart (Figure 5.2.1)

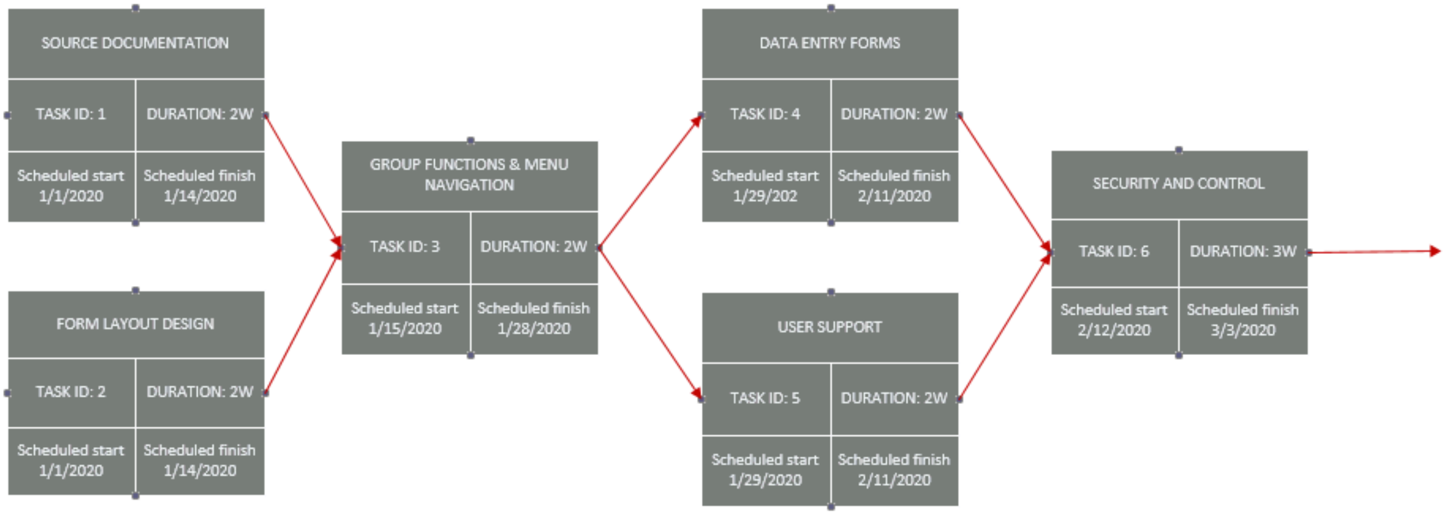


Figure 5.2.1 and 5.2.2: The PERT chart displays each project task, their unique Task IDs, scheduled start and finish dates, duration and their respective dependent and predecessor tasks. The tasks, such as Task 1 and Task 2, are stacked upon one another due to being conducted simultaneously. The same instance occurs during Task 4 and Task 4. The chart continues on to the image below.

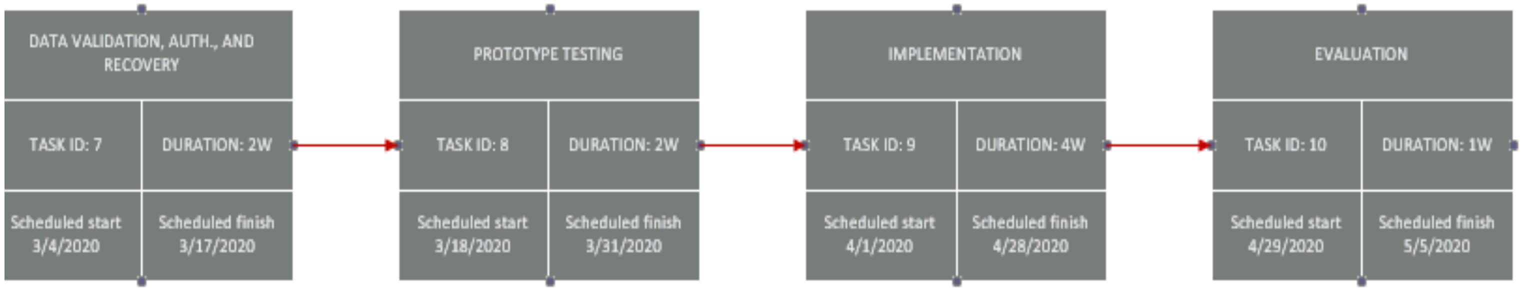
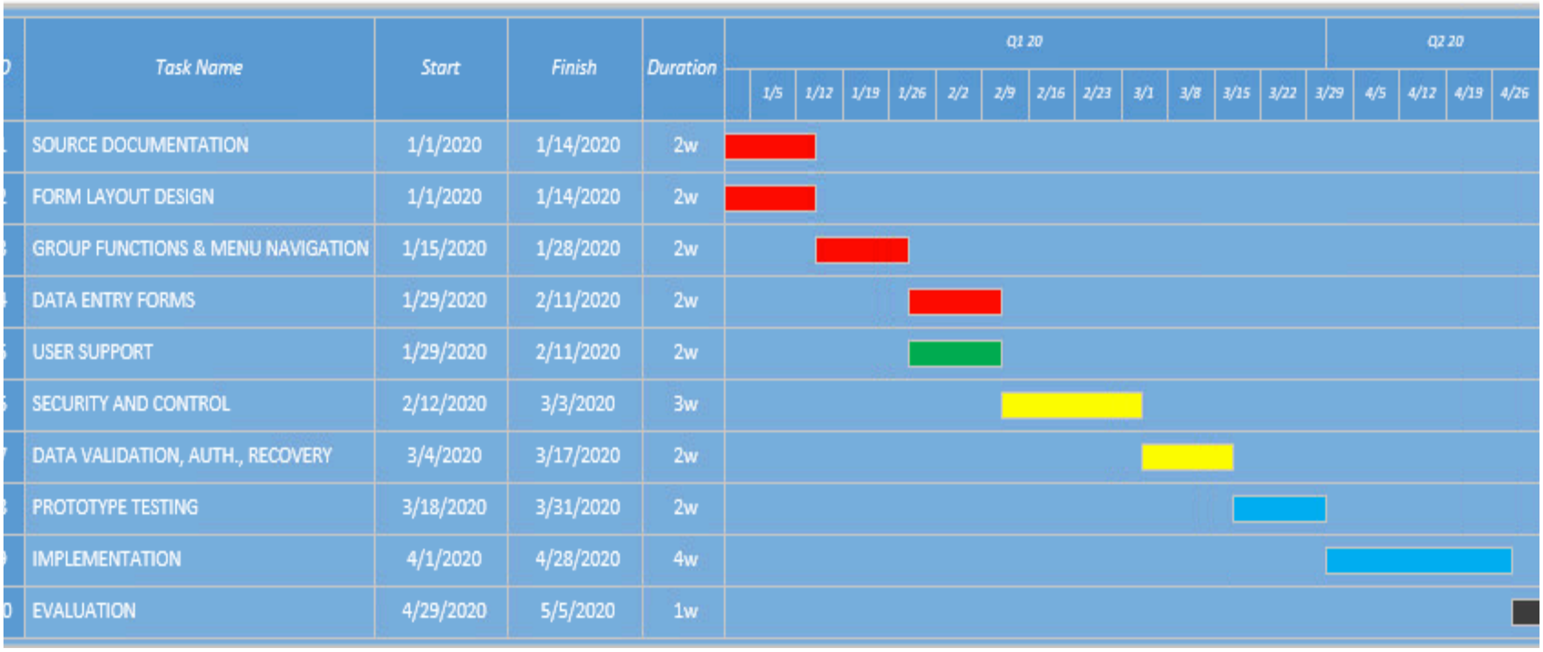

DATA VALIDATION, AUTH., AND RECOVERY
TASK ID: 7
DURATION: 2W
Scheduled start 3/4/2020
Scheduled finish 3/17/2020
PROTOTYPE TESTING
TASK ID: 8
DURATION: 2W
Scheduled start 3/18/2020
Scheduled finish 3/31/2020
IMPLEMENTATION
TASK ID: 9
DURATION: 4W
Scheduled start 4/1/2020
Scheduled finish 4/28/2020
EVALUATION
TASK ID: 10
DURATION: 1W
Scheduled start 4/29/2020
Scheduled finish 5/5/2020

Figure 5.2.2: PERT chart continued from Figure 5.2.1 above.

5.3 Project Timeline & Cost

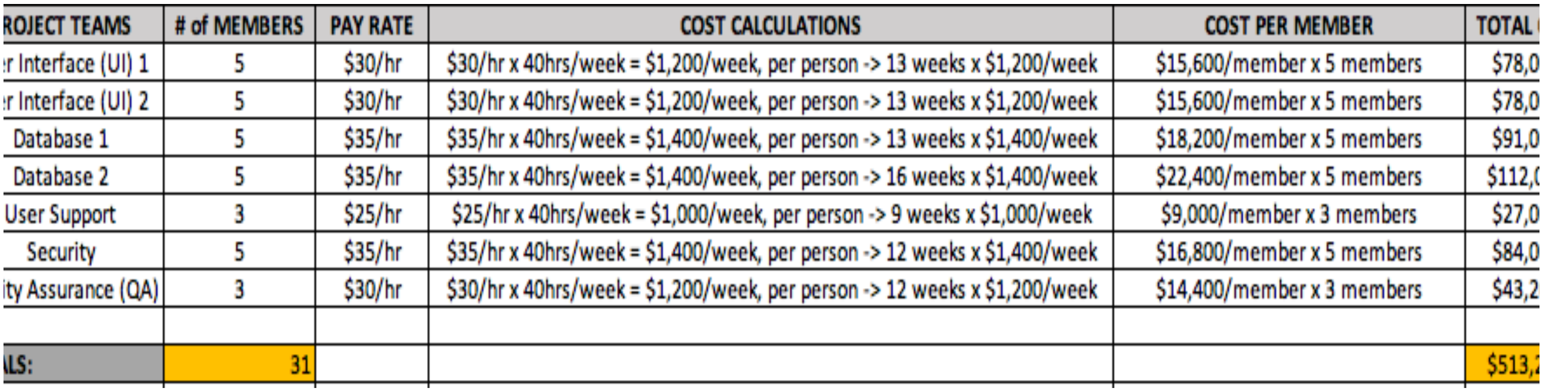
Project Gannt Chart (Figure 5.3.1)



Task Name	Start	Finish	Duration
SOURCE DOCUMENTATION	1/1/2020	1/14/2020	2w
FORM LAYOUT DESIGN	1/1/2020	1/14/2020	2w
GROUP FUNCTIONS & MENU NAVIGATION	1/15/2020	1/28/2020	2w
DATA ENTRY FORMS	1/29/2020	2/11/2020	2w
USER SUPPORT	1/29/2020	2/11/2020	2w
SECURITY AND CONTROL	2/12/2020	3/3/2020	3w
DATA VALIDATION, AUTH., RECOVERY	3/4/2020	3/17/2020	2w
PROTOTYPE TESTING	3/18/2020	3/31/2020	2w
IMPLEMENTATION	4/1/2020	4/28/2020	4w
EVALUATION	4/29/2020	5/5/2020	1w

Figure 5.3.1: The Gannt Chart communicates the project tasks, scheduled dates, duration and timeline that displays tasks that are conducted simultaneously. The color-coded feature of the task duration and progress bars, are only to provide visual contrast. As seen from the 'Duration' column, the project is expected to take 22 weeks.

Project Costs Table (Figure 5.3.2)



ROJECT TEAMS	# of MEMBERS	PAY RATE	COST CALCULATIONS	COST PER MEMBER	TOTAL
r Interface (UI) 1	5	$30/hr	$30/hr x 40hrs/week = $1,200/week, per person -> 13 weeks x $1,200/week	$15,600/member x 5 members	$78,0
r Interface (UI) 2	5	$30/hr	$30/hr x 40hrs/week = $1,200/week, per person -> 13 weeks x $1,200/week	$15,600/member x 5 members	$78,0
Database 1	5	$35/hr	$35/hr x 40hrs/week = $1,400/week, per person -> 13 weeks x $1,400/week	$18,200/member x 5 members	$91,0
Database 2	5	$35/hr	$35/hr x 40hrs/week = $1,400/week, per person -> 16 weeks x $1,400/week	$22,400/member x 5 members	$112,
User Support	3	$25/hr	$25/hr x 40hrs/week = $1,000/week, per person -> 9 weeks x $1,000/week	$9,000/member x 3 members	$27,0
Security	5	$35/hr	$35/hr x 40hrs/week = $1,400/week, per person -> 12 weeks x $1,400/week	$16,800/member x 5 members	$84,0
ity Assurance (QA)	3	$30/hr	$30/hr x 40hrs/week = $1,200/week, per person -> 12 weeks x $1,200/week	$14,400/member x 3 members	$43,2
LS:	31				$513,

Figure 5.3.2: The Project Cost table displays information regarding personnel costs. Each team has designated members, their respective pay rates, workload calculations, costs per member and totals. This table does not account for hardware, outsourcing, nor delay expenses. Those budgeting considerations can be established at a later date.

6.0 CONCLUSION

At the conclusion of the propose project, UR Covered, Inc. will have a fully functional mobile application. With the addition of this mobile resource, customers will be given more timely account alerts, which may include new information, account updates and opt-in/opt-out features. The customer will be able to live chat with their case managers, as well as find their office contact information. User experience and satisfaction are the main areas of focus once the application has been implemented. During the evaluation phase, the organization should consider issuing surveys to customers to obtain feedback on the application usability and functionality.

Glossary & Key Terms

Data design – Data design is the organization of data according to specific models, as well as classifying data assets and identifying interrelationships between such assets (Wikipedia, n.d.).

Data dictionary- A **Data Dictionary**, also called a **Data Definition Matrix,** provides detailed information about the business data, such as standard definitions of data elements, their meanings, and allowable values (Brandenburg, 2019).

Data flow diagram (DFD)- Also known as **DFD, Data flow diagrams** are used to graphically represent the flow of data in a business information system. DFD describes the processes that are involved in a system to transfer data from the input to the file storage and reports generation (Visual Paradigm, 2019).

Data processing model- The **process model** is a core diagram in structured analysis and design. Also called a **data** flow diagram (DFD), it shows the flow of information through a system.
Each **process** transforms inputs into outputs (Excel Software, n.d.).

Feasibility Analysis – feasibility analysis involves several tests that answers questions of whether the project is worthwhile to proceed to deeper levels of development; such tests include operational, economic, technical and scheduling concerns (Rosenblatt, 2017).

Object modeling- An **object model** helps describe or define a software/system in terms of objects and classes. It defines the interfaces or interactions between different models, inheritance, encapsulation and other object-oriented interfaces and features (Techopedia, 2019).

System Architecture – software architecture is the defining and structuring of a solution that meets technical and operational requirements, such elements include security, performance and manageability (Techopedia Inc, 2019).

System Specifications – system specifications cover the requirements, features and behaviors of a particular system. Examples of such requirements include, input, output, processing, performance and controls. Features and behaviors are dependent on the nature of the system and its appropriately programmed task handling capabilities.

Use case diagram- Use case diagrams model the functionality of a system using actors and use cases. Use cases are a set of actions, services, and functions that the system needs to perform (SmartDraw, LLC, 2019).

User Interface Design – UI design refers to the visual layout of the elements that a user might interact within an application (Pidoco GmbH, 2019).

Varchar- A **varchar** or **Variable Character Field** is a set of character data of indeterminate length. The term *varchar* refers to a data type of a field (or column) in a Database Management System which can hold letters and numbers.

References

Bandakkanavar, R. (2018). Krazytech. Software Requirements Specification document with example. Retrieved December 14, 2019 from https://krazytech.com/projects/sample-software-requirements-specificationsrs-report-airline-database

Brandenburg, L. (2019). Bridging the Gap. What is a Data Dictionary. Retrieved December 7, 2019 from https://www.bridging-the-gap.com/data-dictionary/

Excel Software. (n.d.). Excel Software. Process Model. Data Flow Diagrams. Retrieved December 4, 2019 from https://www.excelsoftware.com/processmodel

MINDSEA Team. (n.d.). Should You Outsource App Development (The Definitive Guide). Retrieved December 4, 2019 from https://mindsea.com/mobile-app-agency-inhouse/

Mobile App Daily. (2019). Mobile App Development. How Much Does it Cost to Maintain an App [2020]. Retrieved December 18, 2019 from https://www.mobileappdaily.com/cost-to-maintain-an-app

Nitakron, T. (n.d.). Comparison of Diagramming Tools. Information System Analysis. Retrieved December 1, 2019 from http://www.umsl.edu/~sauterv/analysis/f06Papers/Nitakorn/

Pidoco GmbH. (2019). User Interface Design (UI Design). Retrieved December 20, 2019 from https://pidoco.com/en/help/ux/user-interface-design

Rosenblatt, H. (2017). System Analysis and Design. (11th ed.).

SmartDraw, LLC. (2019). Use Case Diagram. Retrieved Decemeber 8, 2019 from https://www.smartdraw.com/use-case-diagram/

Techopedia. (2019). Object Model. Retrieved December 8, 2019 from https://www.techopedia.com/definition/8635/object-model

Techopedia. (2019). Software Architecture. Retrieved December 20 , 2019 from https://www.techopedia.com/definition/24596/software-architecture

Visual Paradigm. (2019). What is a Data Flow Diagram? Retrieved December 7, 2019 from https://www.visual-paradigm.com/guide/data-flow-diagram/what-is-data-flow-diagram/

Wikipedia. (2019). Varchar. Retrieved December 7, 2019 from https://en.wikipedia.org/wiki/Varchar

Wikipedia. (n.d.). Database Design. Retrieved December 20, 2019 from https://en.m.wikipedia.org/wiki/Database_design