**Landfill Forms**

**(LF)**

**CS-4962 Senior Design**

**Software Design Specification**

Prepared by:

Andres Adame

Teky Alvarado

 14 April 2018

CALIFORNIA STATE UNIVERSITY LOS ANGELES



Los Angeles, California

**Section 1: Introduction**

* 1. ***Purpose***

This document will outline in detail the software architecture and design for the Landfill e-Forms mobile and web applications (LEF). This document will provide several views of the system’s design in order to facilitate communication and understanding of the system. It intends to capture and convey the significant architectural and design decisions that have been made for the LEF.

* 1. ***Scope***

This document provides the architecture and design of the LEF. It will show how the design will accomplish the functional and non-functional requirements detailed in the LEF Functional Requirements and Design (FRD) document.

* 1. ***Intended Audience***

This document is written on a technical level to address the City of Los Angeles, Department of Sanitation’s technical department and CSULA Computer Science department.

* 1. ***References***
* Landfill e-Forms mobile and web applications (LEF) Functional Requirements and Design (FRD) Document.
	1. ***Definitions***
* LEF – Landfill e-Forms web and mobile applications
* IDE – Integrated Development Environment

**Section 2: System Overview**

Landfill e-Forms mobile and web applications (LEF) are being built for the City of Los Angeles, Department of Sanitation in order to convert the Department of Sanitation’s landfill auditing system from paper to fully electronic. LEF is intended to reduce human error, improve efficiency, and speed up report production.

The scope of the project is to make a mobile application for the Android platform where the user can input information into an interface that displays an electronic form. The data will be stored into a database inside the mobile application that will then be transferred over to a private server via the web application. The web application will then be able to sync data and generate reports for the client based on the information in the database.

***2.1 Context Diagram (Level 0 Data Flow Diagram)***



***2.2.1 Context Diagram (Level 1 Data Flow Diagram for Mobile Application***

***2.2.1 Context Diagram (Level 1 Data Flow Diagram for Web Application)***



**Section 3: Design Considerations**

***3.1 Goals***

* Ease of usability is a top priority for the development process. The user should have little knowledge of phone and web application usage in order to operate the applications correctly.
* LEF’s mobile application will have a sidebar menu that will allow the user to choose from a selection of locations and forms, as well as have the ability to log out from the current user account.
* LEF’s web application will provide the users with utilities to upload and manage the collected data. User account management will also be included.

***3.2 Development Methods***

This project is being conducted using a modified waterfall paradigm with two implemental builds (alpha and beta). The development process is formal with document and code reviews.

**Section 4: Architectural Strategies**

***4.1 Use of a particular type of product (programming language, database, library, etc.)***

LEF’s mobile application is being developed for the Android platform using Android Studio as the Integrated Development Environment (IDE). The Android application will be developed using Java and Extensible Markup Language (XML) as its primary programming languages.

LEF’s web application is being developed using Angular 4 on the front end and will be using Visual Studio Code as the IDE. The back end will be developed in Java using Eclipse as the IDE.

***4.2 Reuse of existing software components to implement various parts/features of the system***

LEF’s database is being operated on Microsoft SQL Server Express.

***4.3 Future plans for extending or enhancing the software***

Future plans for LEF are including but not limited to:

* Improving the cosmetic look of LEF for a smoother, more pleasurable experience, on both the mobile and web application side.
* Allow the client to generate and add new forms to the mobile application, web application, and database.

***4.4 User interface paradigms (or system input and output models)***

Use of LEF will be using the graphical display paradigm. Input of data to LEF for use in functions will be either in user input textboxes or dropdown choice menus. Output of LEF will be given via queries entered into the web application.

***4.5 Error detection and recovery***

Data on LEF’s mobile application will be backed up to a local database within the application in case of data loss. Data on the web application side will be backed up to private server hosted by the client. Errors will be handled and checked by both the mobile and web application.

***4.6 External databases***

No external databases will be utilized unless specified by the City of Los Angeles, Department of Sanitation.

***4.7 Distributed data or control over a network***

The client will determine whether use of distributed data is necessary. Control over a network will be provided by the client, as well.

***4.8 Generalized approaches to control***

Approaches to control shall be achieved when the user successfully enters their personal username and password.

***4.9 Concurrency and synchronization***

LEF’s data from the mobile application will be synchronized through its local database, and the web application will be synchronized with the database in the private server.

***4.10 Communication mechanisms***

Communication from LEF’s mobile application to the web application and database will be done through hardwired sync.

**Section 5: System Architecture**

***5.1 component-1 (subsystem-1) Mobile Device (with Android mobile operating system)***

***5.1.1 –*** The mobile device will be where the Landfill e-Forms Android application will be installed. The mobile device will provide the user a touch screen interaction with the LEF. The mobile device will also provide the means of communication between the user and the web application via hardwired sync.

**Section 6: Detailed System Design**

***6.1 module-1 Android Main***

***6.1.1 –*** The Android Main Module (AMM) will be responsible for displaying data and taking input from the user, as well as access data from web application sync. It is also responsible for processing data that it receives from the web application.

***6.2 module-2 Android Login***

***6.2.1 -*** In the Android Login Module (ALM), the user enters their username and password. This is used for authentication purposes in order to prevent unauthorized users from accessing the application. Upon successful login, the user will be directed to the application. Upon unsuccessful login, the user will be redirected back to the login screen. Users that are unable to login will not be able to access the application.

***6.3 module-3 Data Transfer***

***6.3.1 -*** The Data Transfer Module (DTM) allows the Android application to convert the data taken from the application and export the information into a JSON file. The JSON file contains data that the user has inputted from the forms module (refer to Module 6.4 for more information). The JSON file will then be uploaded to the web application and the data from that file will be stored in the database.

***6.4 module-4 Forms***

***6.4.1 -*** The Forms Module (FM) in the Android application allows the user to track and input data depending on the desired site and form. The user first selects a site and an instrument. After that, depending on the instrument chosen, the user will be directed to the form that corresponds to the instrument. The types of forms include: Instantaneous, Integrated, Probe, and Instantaneous Monitored Emission (IME). Initially, the form will display a list of grids that need attention. This depends on whether the site has been completed or not. The user can then access and create a new form. The parameters for each form varies. In the case of an Instantaneous Form, it will take in the grid number, start time, end time, and CH4 reading. In the case of the forms, if the CH4 reading is within a certain threshold, a notification will pop up asking them if they want to create a new form. In the case of the instantaneous form, if the CH4 reading is above 500, the application will prompt the user and ask if they want to create a new IME form. The user can also choose to create a new IME form without the need of creating an instantaneous form. The user is able to change through to a different form by changing the instrument.

***6.5 module-11 Equipment***

***6.5.1 -*** The Equipment Module (EM) shall display a list of equipment, as well as allow the user to create new equipment, delete, as well as edit equipment.

***6.6 module-6 Users***

***6.6.1 -*** The Users Module (UM) shall allow the administrator to add, edit, and delete users. It shall allow the administrator to create user roles with permissions, as well as assign user roles to the users.

**Section 7: Graphical User Interface Design**

Android GUI





**Section 8: Glossary**

* LEF – Landfill e-Forms
* FRD – Functional Requirement and Design
* IDE – Integrated Development Environment
* XML – Extensible Markup Language
* SQL – Structured Query Language
* AMM – Android Main Module
* ALM – Android Login Module
* DTM – Data Transfer Module
* LM – Location Module
* FM – Forms Module
* SM – Sync Module
* DAM – Database Access Module
* RM – Report Module
* ECI – Environmental Compliance Inspector
* IME – Instantaneous Monitored Emission