AN ANDROID APPLICATION FOR THE USDA STRUCTURAL DESIGN SOFTWARE

by

RAJESH KANNIKANTI

B.Tech, Acharya Nagarjuna University, 2011

A REPORT

submitted in partial fulfillment of the requirements for the degree

MASTER OF SCIENCE

Department of Computing and Information Sciences
College of Engineering

KANSAS STATE UNIVERSITY
Manhattan, Kansas

2013

Approved by:

Major Professor
Dr. Mitchell L. Neilsen

Abstract

People are more inclined to use tablets instead of other computing devices due to their portability and ease of use. A number of desktop applications are now becoming available as tablet applications, with increasing demand in the market. Android is one of the largest and most popular open source platforms that offer developers complete access to framework APIs in order to develop innovative tablet applications.

The objective of this project is to develop an Android application for the U.S. Department of Agriculture (USDA) Structural Design Software. The GUI for this software is developed to run on tablet devices powered by Android platform. The main features provided by the User Interface include:

- Allowing the input to be saved in ASCII text format and displaying the simulation results in PDF format
- Allowing the user to select the type of project or view help contents for the projects
- Allowing the user to build the simulation for the selected type of project
- Allowing the user to send the simulation results to an e-mail

The backend for this software is supposed to replace the old FORTRAN source files with Java source files. FORTRAN to Java translation is performed using the FORTRAN to Java (F2J) translator. F2J is intended to translate old FORTRAN math libraries, but was not completely successful in translating these FORTRAN programs. To accomplish successful translation, some features (such as Common Blocks, IO operations) were removed from the FORTRAN source files before translation. After successful translation, the removed features were added again to the translated Java source files.

The simulation results provided by the software are useful to design engineers to develop new structural designs.

Table of Contents

List of Figures

List of Tables

Acknowledgements

I would like to take this opportunity to thank my major professor, Dr. Mitchell L. Neilsen, for his valuable guidance and support throughout the project. Dr. Neilsen, I feel honored and privileged to have had the opportunity to work with you.

I would also like to thank Dr. Torben Amtoft and Dr. Daniel Andresen for graciously accepting to serve on my committee.

My special thanks to Dr. Neilsen for providing the Android device to test the application during the development.

I would like to thank my professors in the CIS department at Kansas State University for giving me the opportunity to take courses and work with them, which provided me with more practical guidance and experience.

I would like to thank my parents, family and friends for their continuous support and encouragement.

Finally, I would like to thank the administrative and technical support staff of the CIS department for their support in successful completion of my graduate study.

Chapter 1 - Project Description

1.1 Introduction

The soil conservation service at the U.S. Department of Agriculture [1] annually designs a number of cast-in-place rectangular conduits (Single Cell [3] or Twin Cell [4]) for use in principal and emergency spillways passing through the earth embankments, reinforced concrete rectangular channels [5], SAF stilling basins [6] and monolithic straight drop spillways [7]. Design of these structural designs by manual methods is a difficult and time consuming process.

Initially, computer programs written in Fortran 77 for IBM 360 [10] were developed to perform this design task. Later on several versions of this software [10] such as Legacy structural design software servlets (Java Servlets version), Windows .Net Version (Using Fortran executable), PC version with C# source (Using Fortran executable) and an Android version with java source were developed. The Android version available has a portion of FORTRAN to Java translation (for Single Cell [3], Twin Cell [4], CCHAN [5] and CBASIN [6]) performed and a portion of the user interface developed targeting Android Gingerbread Operating System.

This project is an Android application for the USDA structural Design software targeting Android tablet devices. In this project I worked on FORTRAN to Java translation for DRPWS3E [7] project and developed a user friendly Graphical User Interface which is capable of handling all the five structural design projects in a single application. This Android application is developed targeting the Android Honeycomb (3.1) operating System.

1.2 Motivation

The main motivation behind this project was to develop an Android Application for the USDA Structural Design Software which will assist the design engineers in coming up with new structural designs. With the availability of their own RAM and processing unit, tablets can perform complex operations faster enough when compared with high performance desktops and laptops available today. Thus the necessity for an android application for the USDA structural design software has increased, as people prefer tablets instead of other computing devices due to their portability and ease of use.

The primary factor that motivated me to work on this project is to learn Android Application development. With the fairly decent practical experience I have on the translator design concepts, the necessity to work on a practical tool (F2J) to perform the FORTRAN to Java Translation is another factor that motivated me. Also, this project requires a basic understanding of FORTRAN [13] (First ever High level programming language) for debugging purpose during FORTRAN to Java translation. The Android Application for the USDA structural Design software is an application which provides the users with an easy to use Graphical User Interface for performing the Structural Design tasks for all the five Structural Design project types and optionally emailing the results of the Structural Designs.

1.3 FORTRAN to Java Translation

The backend for this application is supposed to replace the old FORTRAN executable with the Java source files performing the same task of providing structural designs. For this purpose, F2J (FORTRAN To Java) [9] translator was chosen.

1.3.1 Necessity for FORTRAN to Java Translation

On many occasions we face the requirement of converting source code of software from one programming language to another. This requirement is driven mostly by the need to port the software to a different Operating system. The main intention behind FORTRAN to Java translation for this project was due to Android operating system being not capable of running FORTRAN executable. So FORTRAN to Java translation was chosen, as Android operating system has a great support for Java Programming Language.

1.3.2 FORTRAN To Java Translator (F2J)

FORTRAN to Java Translator (F2J) [9] was a research tool developed at the Innovative Computing laboratory of University of Tennessee. The F2J translator is a formal compiler that translates programs written using a subset of Fortran 77 in to a form that may be executed on JVM (Java Virtual Machines). However F2J was developed with a primary motivation of translating old FORTRAN linear algebra libraries instead of full programs. Thus the F2J translator will not guarantee the successful translation of the FORTRAN source files of this project if attempted to translate directly. To ensure successful translation of these FORTRAN source files, some features such as COMMON blocks, some IO operations and some String operations are removed from source files before translation. After the successful translation, the removed elements prior to the translation were added again.

1.3.3 Other Formatters Used

Certain other formatters were used in addition to F2J tool during the process of FORTRAN to Java translation. The other formatters used include:

1.3.3.1 CommonOut

The Fortran To Java translator (F2J) does not supports unnamed common blocks and may mistranslate named common blocks of Fortran 77 during translation. To avoid this scenario, the CommonOut formatter [10] was used. This formatter will remove the common blocks (both named and unnamed) within the FORTRAN source files. The Formatter will take a FORTRAN source file as an input, produces another FORTRAN source file and a Text file. The output FORTRAN source file produced was the original FORTRAN source file with the common blocks removed. In order to allow F2J to parse the second FORTRAN source file, DIMENSION declarations of all arrays that were in the common blocks were replaced in place of common blocks. The text file produced will contain all the common blocks removed along with the function name they belong to.

Usage: java CommonOut CO1.FOR CO2.FOR commonout.txt

Where CO1.FOR is the input source file and is not modified in the process for error checking, CO2.FOR is the output FORTRAN source file with common blocks removed, commonout.txt is a text file containing the removed common blocks

1.3.3.2 CommonIn

The CommonIn formatter [10] is used to replace variables that are common globals within a predefined common.java and commonout.txt after the successful f2j translation.

Usage: java CommonIn CI1.java Common.java CommonOut.txt   CI2.java

Where CI1.java is a java source file resulting from the F2J translation, Common.java is a pre-made file that holds of all the global variables, CommonOut.txt is a text file resulting from using the Commonout formatter; CI2.java is the resulting java file after using this formatter, where all the variables that are common globals are replaced to access common class instead. This usage will give an error if the java file it is parsing (CI1.java) is not in the commonOut.txt file or if there is any global variable in the CommonOut.txt which is not identified in the Common.java for the java file being parsed

1.3.3.3 SwapStreams

The SwapStreams formatter [10] is used on F2J translated Java source files that write to a file. This formatter will replace all instances of "Util.f77write(" with "UtilStream.f77write(out," which will allow the translated Java source file to use the f77write method of the UtilStream class. The f77write is simply the FORTRAN format parser that writes to the print stream object called out. This formatter is used to change translated programs from writing to stdout to a file of your choice.

Usage: SwapStreams SS1.java SS2.java

SS1.java is a translated Java source file writing to stdout, SS2.java is a Java source file resulting from using this formatter which will write to a file of your choice.

1.4 Graphical User Interface

The Graphical User Interface is kept simple, understandable and easy to navigate. The use of Action bar provides the user with a global navigation and consistent interface across different screens of the application.

The GUI for the software was developed to run on tablet devices powered by Android platform. The features provided by the User Interface include:

- Allowing the input to be saved in ASCII text format.
- Allowing the user to view the simulation results in PDF [15] format.
- Providing an optional feature of sending the simulation results to an e-mail with the help of android e-mail application.
- Allowing the user to select the type of project (SINGCELL, TWINCELL, CCHAN, CBASIN, DRPWS3E) for which they wish to build the structural designs.
- Allowing the user to build the simulation for the selected type of project.
- Providing Help contents in PDF format for the different type of projects (SINGCELL, TWINCELL, CCHAN, CBASIN, and DRPWS3E)

Chapter 2 - Requirement Analysis

2.1 Requirements Gathering

I have collected most of the requirements for the project from Dr. Mitchell Neilsen. Regular meetings with him at every phase of the project development helped in meeting the requirements for the project. Apart from this, this project requires a reasonable understanding of the structural design terminology which I learned with the help of documentation materials [2] of the software. As this project involves working with FORTRAN to Java translation, a decent experience with the FORTRAN programming language will be handy especially for debugging purposes. I learned the basic concepts of the FORTRAN programming language from the Fortran 77 tutorials [13]. Finally this project requires a lot of background research in building applications on android framework. Android Developers Tutorials [13] provided me with the necessary information in building this application. In order to make use of the Action Bar provided by the latest versions of Android operating system, this project was developed on Android 4.2 Jelly Bean platform.

2.2 Requirements Specification

The requirements specification for the project can be categorized in to two parts:- The Requirements Specification for FORTRAN to Java translation and the Requirements Specification for the development of the Android application.

2.2.1 Requirements Specification for FORTRAN to Java translation

2.2.1.1 Software Requirements

Operating System: Linux

Languages: Java, FORTRAN

Java Version: Java SE6

Tools: F2J (Fortran To Java) translator, Eclipse Helios IDE

Debugger: Eclipse Debugger

2.2.1.2 Hardware Requirements

Processor: Pentium IV or higher

RAM: 512MB

Disk Space: 1GB or higher

2.2.2 Requirements Specification for the development of Android Application

2.2.2.1 Software Requirements

Development end:

Operating System: Windows 7

Language: Java, Android SDK, XML

Tools: Eclipse Juno IDE

Technologies: Java, Android, XML

Debugger: Android DDMS (Dalvik Debug Monitor Service), Android Samsung Tab 10.1

User end:

Framework: Android SDK version 3.0

Network: Internet Connection

2.2.2.2 Hardware Requirements

Development end:

Processor: Pentium IV or higher

RAM: 512 MB

Disk Space: 250 MB or higher

User end:

Android Device: Any Android Tablet with Operating System Honeycomb or higher

Chapter 3 - System Architecture, Android Framework Components

3.1 System Architecture

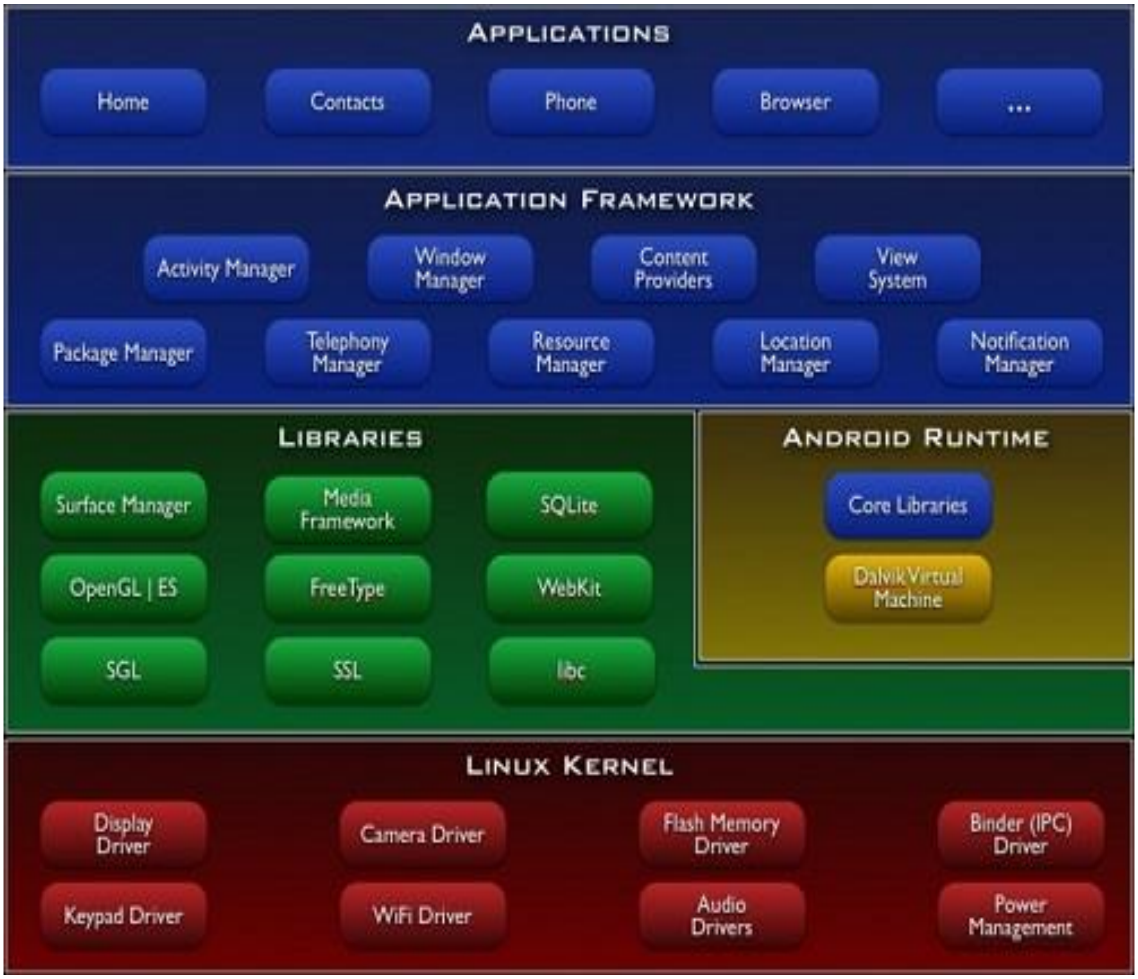


Figure 3.1 Android Architecture

(from elinux.org)

From the above Android Architecture, it can be observed that the Android Operating System is a software stack of various layers with each layer offering different services to the layer above it. The Android Layers include:

- The Linux Kernel, which is responsible for interacting with the hardware, networking, files system access and inter-process communication
- Native Libraries, which are specific to the system Hardware (written in C or C++)
- Android Runtime, which consists of core libraries and the DDVM
- Application Framework with which Android applications interact.

- Applications consist of the Android application written in java and executing in a Dalvik Virtual Machine.

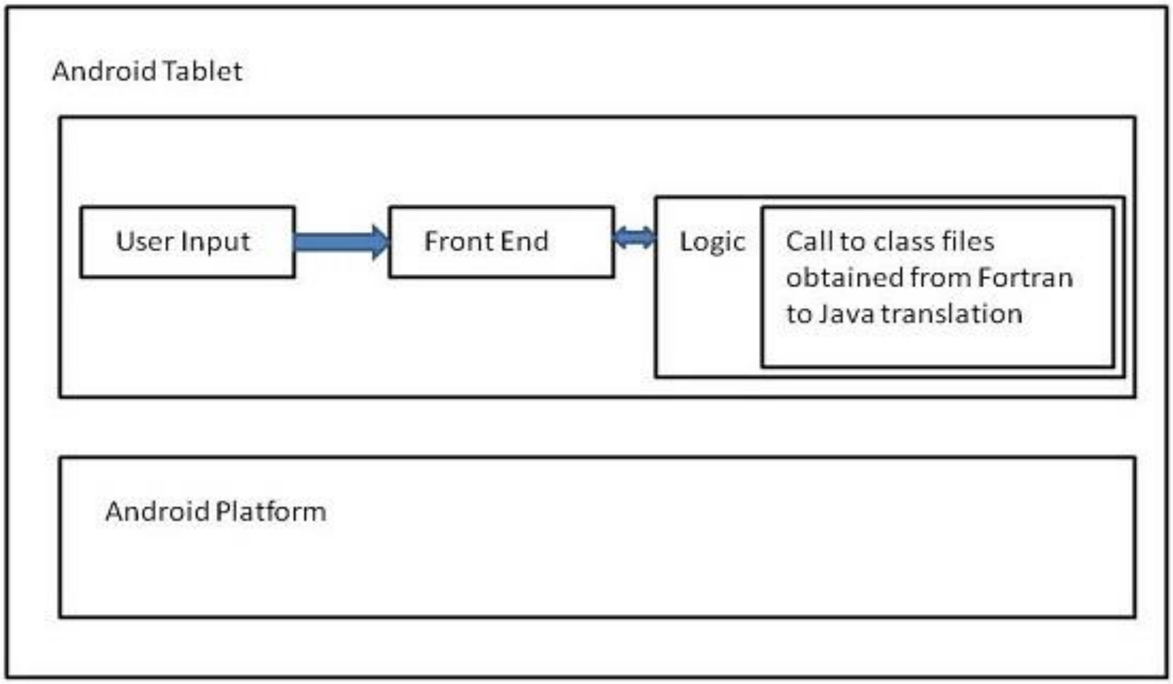

Android Tablet
User Input
Front End
Logic
Call to class files obtained from Fortran to Java translation
Android Platform

Figure 3.2 System Architecture

From the System Architecture above, the application runs on the Android platform present in the Android framework of the Android Tablet. The input to this application is via the touch event of the user. Depending on the touch input from the user, corresponding action will takes place and the user may be displayed with the necessary information or will be provided with the next screen to interact with.

Once the user touches on any of the project buttons on the home page, the user will be redirected to the respective project screen. In each of the five project screens, the user will have the option of selecting a new input form, filling out the input form, auto filling the input form by opening a text file from the external storage which contains a saved input by the user, saving the input filled in the form in a text file on the external storage, building the structural design, viewing help content for any of the five projects, selecting any of the project screen different from the current project screen or can exit the current project screen.

When the user opts to build the structural design on any of the five project screens, the application interacts with the corresponding project’s class files obtained from the FORTRAN to

Java translation process to build the structural design and the results of the structural design will be displayed to the user in a PDF format. Optionally, the user can send the results to an email of his choice.

3.2 Android Framework Components

An Android application is a package which consists of loosely coupled components which are connected to each other at run time. In this section the various components of an android application such as activity, intent are discussed.

3.2.1 AndroidManifest.xml file

Every Android application has an AndroidManifest.xml file in its root directory which contains a list of all the activities, intents and permissions which the app includes. It also includes the minimum SDK version required to run the application.

```xml
<?xml version="1.0" encoding="utf-8"?>
<manifest xmlns:android="http://schemas.android.com/apk/res/android"
    package="edu.ksu.usdastructuraldesign"
    android:versionCode="1"
    android:versionName="1.0" >

    <uses-permission android:name="android.permission.WRITE_EXTERNAL_STORAGE" />
    <uses-permission android:name="android.permission.INTERNET" />

    <uses-sdk
        android:minSdkVersion="8"
        android:targetSdkVersion="17" />

    <application
        android:allowBackup="true"
        android:icon="@drawable/ic_launcher"
        android:label="@string/app_name"
        android:theme="@style/AppTheme" >

        <activity
            android:name="edu.ksu.usdastructuraldesign.HomeActivity"
            android:label="@string/app_name" >
        </activity>

        <activity
            android:name="edu.ksu.usdastructuraldesign.InstallActivity"
            android:label="@string/app_name" >
            <intent-filter>
                <action android:name="android.intent.action.MAIN" />

                <category android:name="android.intent.category.LAUNCHER" />
            </intent-filter>
        </activity>
        <activity
            android:name="edu.ksu.usdastructuraldesign.SingcellActivity"
            android:label="@string/title_activity_singcell" >
        </activity>

        <activity
            android:name="edu.ksu.usdastructuraldesign.Twincell"
            android:label="@string/title_activity_twincell" >
        </activity>

        <activity
            android:name="edu.ksu.usdastructuraldesign.Cchan"
            android:label="@string/title_activity_cchan" >
        </activity>

        <activity
            android:name="edu.ksu.usdastructuraldesign.Cbasin"
            android:label="@string/title_activity_cbasin" >
        </activity>

        <activity
            android:name="edu.ksu.usdastructuraldesign.Drpws3e"
            android:label="@string/title_activity_drpws3e" >
        </activity>

        <activity
            android:name="edu.ksu.usdastructuraldesign.FileOpen"
            android:label="@string/title_activity_file_open" >
        </activity>

        <activity
            android:name="edu.ksu.usdastructuraldesign.SaveInputFile"
            android:label="@string/title_activity_save_input_file" >
        </activity>

    </application>
</manifest>
```

3.2.2 Activities

An Activity is a single, focused application component that the user can interact to do something. The list of activities in this application includes - HomeActivity, InstallActivity, FileOpen, SaveInputFile, and the five structural design activities: Singcell, Twincell, Cchan, Cbasin, and Drpws3e.

3.2.3 Intents

An Intent is an abstract description of an operation to be performed. Intents are used for intercommunication between activities within an application or between applications, to activate broadcasts or to communicate with a background service. The function startActivity() is used to start an intent. The two primary pieces of an intent are action and data, while there may be several secondary attributes such as category, type, extras etc. An action defines the general action to be performed such as ACTION_VIEW, ACTION_SEND etc., and data is the information that is operated when an intent is activated.

This application makes use of ACTION_VIEW intent to view the Structural Design results in PDF format and ACTION_SEND intent to send a copy of Structural Design results to an e-mail.

Intent to view the Structural Design simulation results in PDF format:

```
Intent pdfIntent = new Intent(Intent.ACTION_VIEW);
pdfIntent.setDataAndType(path, "application/pdf");
pdfIntent.setFlags(Intent.FLAG_ACTIVITY_CLEAR_TOP);
try{
      startActivity(pdfIntent);
   }
catch(ActivityNotFoundException e2){
      Toast.makeText(this, "You must have a PDF Viewer to view the  results.",
                                                             Toast.LENGTH_LONG).show();
   }
```

Intent to send a copy of the Structural Design results to an email:

```
Intent email = new Intent(Intent.ACTION_SEND);
email.putExtra(Intent.EXTRA_EMAIL,new String[]{emails});
email.putExtra(Intent.EXTRA_SUBJECT, "Single Cell Structural Design Results");
email.putExtra(Intent.EXTRA_TEXT, "SINGLE CELL");
email.setType("message/rfc822");
try{
      email.putExtra(Intent.EXTRA_STREAM, uri);
      startActivity(Intent.createChooser(email, "Choose an Email client :"));
   }
catch (Exception e){
      Toast.makeText(SingcellActivity.this, "Failed to attach",
                                                          Toast.LENGTH_SHORT).show();
   }
```

Intents are of two types namely explicit intents and implicit intents. Explicit intents are those intents that make use of components name to call an activity within a known source, whereas Implicit intents do not use a components name. Intent filters are used to provide information about implicit intents in AndroidManifest.xml file.

Chapter 4 - Implementation, Testing and Results

4.1 Implementation

The main objective of the USDA Structural Design Software application is to provide a simple, understandable and user friendly interface that will help the users in accomplishing their task with ease and use. The main features of this application include:

- Allowing the user to save the input in a text file
- Allowing the user to open the input from a text file
- Allowing the user to select a project screen of his choice from any of the five project screens.
- Allowing the user to view the help content for a project of his choice from any of the five project screens.
- Allowing the user to build the Structural Design Simulation for the project selected.

The Android application is developed on Android SDK version 4.2 using Eclipse Juno IDE. The screens were designed using XML and the business logic was written in Java. This application consists of 8380 lines of code including the Java and the XML files. The Breakdown of Lines of Code (LOC) is given as follows.

Language	LOC
Java	4520
XML (both Landscape and Portrait modes)	3860
Total	8380

Table 4.1 Lines of Code (LOC)

In order to build the Structural Design Simulation, the application makes use of the Java class files obtained from the process of FORTRAN to Java translation. While the Fortran to Java translation for the projects Single Cell [3], Twin Cell [4], Cchan [5], Cbasin [6] projects are already implemented, I have implemented the Fortran to Java translation for the Monolithic Straight Drop Spillways (DRPWS3E) [7] project. The rest of this section discusses the Implementation of the FORTRAN to Java translation for the DRPWS3E project and the Implementation of the Graphical User Interface.

4.1.1 FORTRAN to Java translation for DRPWS3E

The FORTRAN to Java translation for the DRPWS3E was performed using F2J translator [9] and the formatters [10] mentioned above. The procedure adopted to implement the FORTRAN to Java translation includes the following steps:

- The first step is to run CommonOut formatter on the DRPWS3E.FOR file to produce another .FOR file with the common blocks removed and a text file.

  Java CommonOut DRPWS3E.FOR DRPCOUT.FOR CommonOut.txt
- Next step is to run F2J on the second .FOR file

  f2java DRPCOUT.FOR

  During this step, f2java might produce some parsing errors while attempting to read the DRPCOUT.FOR file. Fix all the parsing errors that are produced. Strings, Comment Lines, Format Strings and DO loops produce most of the parsing errors. Next run the f2java on the DRPCOUT.for file.
- The next step is to create the common.java file. The easiest way to make the common.java file is to make it hold all of the global as floats and read the declarations of each java file produced. Any commons located in the files declared as int, intW, floatW, arrays should be changed in the common.java file appropriately. The declarations of the common.java should be preceded by /*Global Declarations*/ and terminated by /*End Global Declarations */ to guide CommonIn formatter. The common.java file should also be made accessible by a singleton method to simplify reintroducing commons.
- The next step is to reintroduce commons using CommonIn formatter. It should be able to run on any .java file listed in commonout.txt and reintroduce commons.

  Java CommonIn D1.java Common.java commonout.txt D1C.java
- The next step is to use SwapStreams formatter to allow the .java files write to a file instead of stdout. As DRPWS3E project contain multiple files that require file output, Printstream is placed in to the Common class.

  Java SwapStreams D1C.java D1S.java
- The next step is to replace f77 read statements with equivalent java commands in order to replace input to file input.
- The next step is to recompile the project.

- The last step is to run the javab command on .class files. With this step, the F2J translation for DRPWS3E is completed.

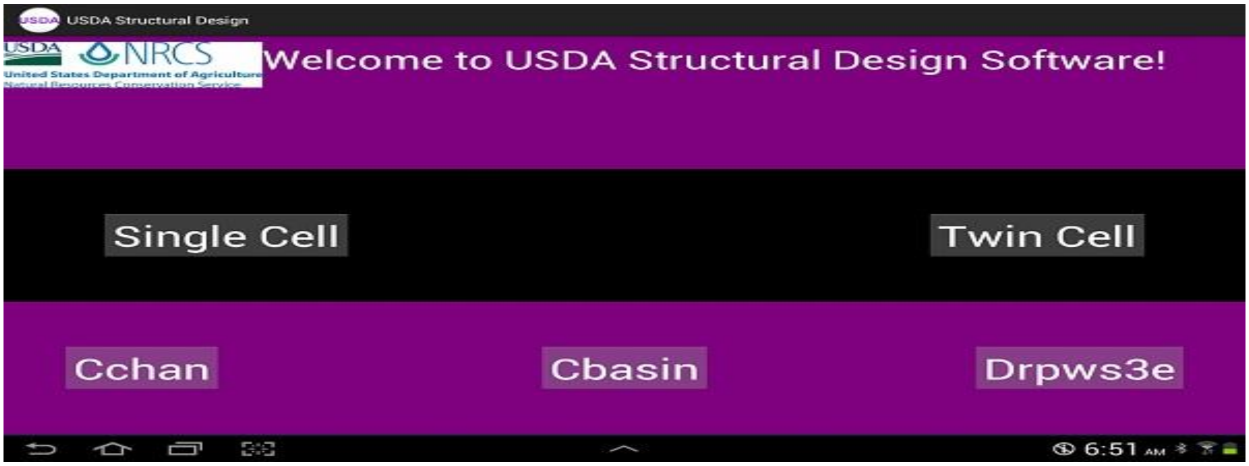
  Javab *.class

4.1.2 Graphical User Interface

The Graphical User Interface has been kept simple, understandable and easy to navigate. Attempts are made to keep the layout consistent for all the screens so that the application will be compatible in both landscape and portrait modes. The user does not require any memorization of the navigational instructions in the application. Hints are included for the input fields to help the user in giving the correct input. The GUI has been designed targeting Android tablet users rather than mobile users as this application requires lot of input entry from the user which will work best on wider screen of tablets. Hence this application is limited to tablet users.

4.1.2.1 Home Page

The Home page is the starting screen of the application displayed to the user when the application is launched. This screen displays the logo of the USDA Natural Resources Conservation Service [16] and the name of the application at the top of the screen. The user can navigate to five other screens (Single Cell [3], Twin Cell [4], CCHAN [5], CBASIN [6], and DRPWS3E [7]) from this screen. The Single Cell button takes the user to the Single Cell project screen, Twin Cell to the Twin Cell project Screen, CCHAN to the Cchan project Screen, CBASIN to the Cbasin project screen and DRPWS3E button to the DRPWS3E project Screen.



Welcome to USDA Structural Design Software!

USDA Structural Design
USDA NRCS
United States Department of Agriculture
Natural Resources Conservation Service
Single Cell
Twin Cell
Cchan
Cbasin
Drpws3e
6:51 AM

Figure 4.1 Home Page Screen [16]

4.1.2.2 Action Bar

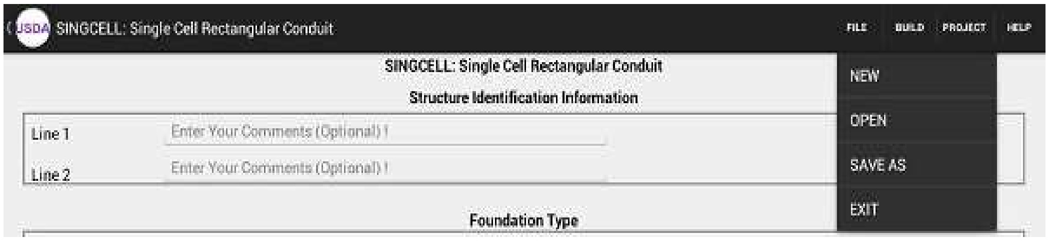
The Action Bar is a dedicated piece of space at the top of each screen that is persistent throughout the app. It provides a dedicated space for app's identity and consistent navigation throughout the app.

Including Action Bar in the application [11]:

```
    // Inflate the menu; this adds items to the action bar if it is present.
getMenuInflater().inflate(R.menu.optionmenu, menu);
```

In this application, the Action Bar is consistent throughout the five different project Screens. The Action Bar consists of the following menu items:

- File menu Item with the sub menu items New, Open, Save As and Exit.



SINGCELL: Single Cell Rectangular Conduit

SINGCELL: Single Cell Rectangular Conduit

Structure Identification Information

Line 1 Enter Your Comments (Optional) !

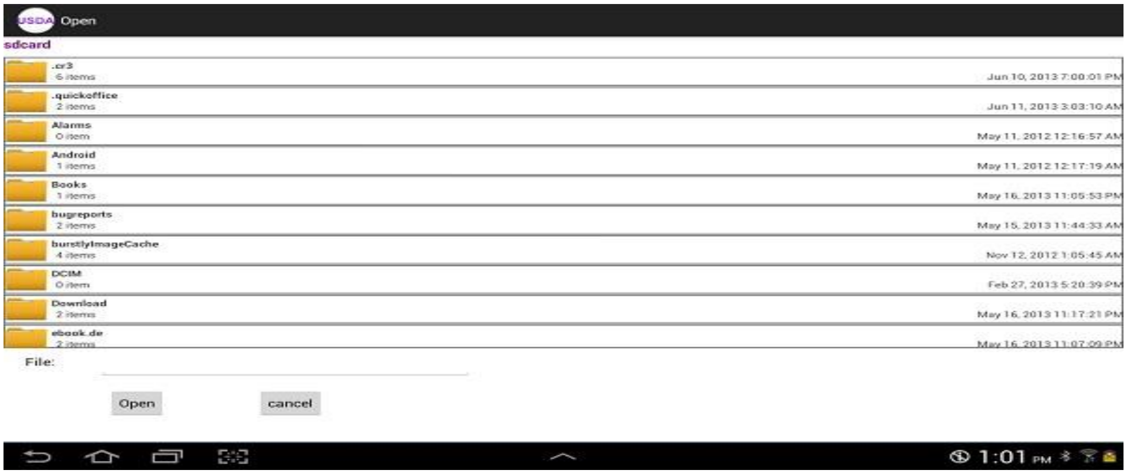
Line 2 Enter Your Comments (Optional) !

Foundation Type

Figure 4.2 File Menu Item and Its Sub Menu Items

New: On clicking this on any of the five project screens, the user will be displayed with a new Input form based on the project selected. This was implemented by using a recursive procedure to clear all the text input fields and setting Radio Buttons and Check Box fields to the default values.

Open: On clicking this sub menu item, on any of the five project screens, the user will be displayed a File Open window, from which the user can select a text input file. The input from the selected text file will be placed in the input fields.



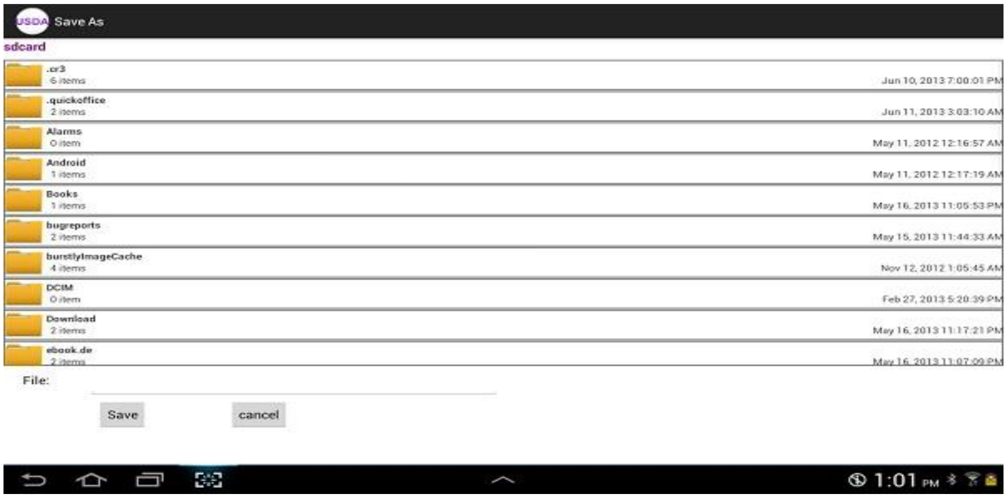
sdcard

.cr3 6 items	Jun 10, 2013 7:00:01 PM
.quickoffice 2 items	Jun 11, 2013 3:03:10 AM
Alarms 0 item	May 11, 2012 12:16:57 AM
Android 1 items	May 11, 2012 12:17:19 AM
Books 1 items	May 16, 2013 11:05:53 PM
bugreports 2 items	May 15, 2013 11:44:33 AM
burstlyImageCache 4 items	Nov 12, 2012 1:05:45 AM
DCIM 0 item	Feb 27, 2013 5:20:39 PM
Download 2 items	May 16, 2013 11:17:21 PM
ebook.de 2 items	May 16, 2013 11:07:09 PM

File:

Figure 4.3 File Open Screen

Save As: On clicking this sub menu item, on any of the five project screens, the user will be displayed a File Save As window, from which the user can save the input in a text file.

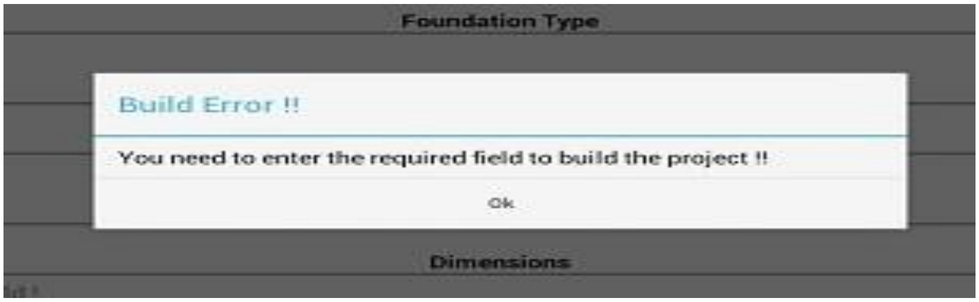


.cr3	6 items	Jun 10, 2013 7:00:01 PM
.quickoffice	2 items	Jun 11, 2013 3:03:10 AM
Alarms	0 item	May 11, 2012 12:16:57 AM
Android	1 items	May 11, 2012 12:17:19 AM
Books	1 items	May 16, 2013 11:05:53 PM
bugreports	2 items	May 15, 2013 11:44:33 AM
burstlyImageCache	4 items	Nov 12, 2012 1:05:45 AM
DCIM	0 item	Feb 27, 2013 5:20:39 PM
Download	2 items	May 16, 2013 11:17:21 PM
ebook.de	2 items	May 16, 2013 11:07:09 PM

Figure 4.4 File save Screen

Exit: On clicking this sub menu item, the user can exit from the project screen

- **Build Menu Item**: For any of the selected project screens, when the user clicks on this menu item, the structural design for the given input will be built for the respective project. Initially, it will check for all the required fields and will prompt the user with the help of an alert dialog window if any of the required input fields are left blank. If all the required input fields are filled out, it will build the structural design and the results will be displayed in a PDF format using an external PDF Intent. If the user wishes to send the results to an e-mail, he will be prompted with the email intent.



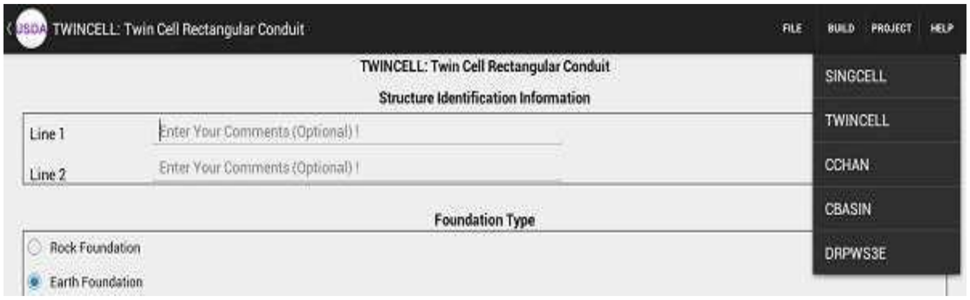
Build Error !!

You need to enter the required field to build the project !!

Figure 4.5 Build Error Alert Dialog

- Project Menu Item with the sub menu items SINGCELL, TWINCELL, CCHAN, CBASIN, and DRPWS3E

Project: On clicking this Menu item, the user will be displayed with five sub menu items (five project types). The User can switch to any of the project types from this menu item.



TWINCELL: Twin Cell Rectangular Conduit

Structure Identification Information

Line 1 Enter Your Comments (Optional) !

Line 2 Enter Your Comments (Optional) !

Foundation Type

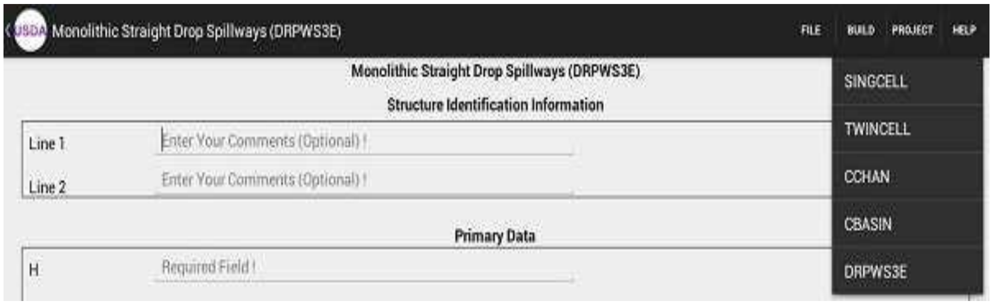
○ Rock Foundation

● Earth Foundation

Figure 4.6 Project Menu Item and its Sub Menu Items

- Help Menu Item with the Sub Menu Items SINGCELL, TWINCELL, CCHAN, CBASIN, and DRPWS3E

Help: On clicking this Menu item, the user will be displayed with five sub menu items (five project types). The User can view the Help content for any of the project types from this menu item in PDF format.



Monolithic Straight Drop Spillways (DRPWS3E)

Structure Identification Information

Line 1 Enter Your Comments (Optional) !

Line 2 Enter Your Comments (Optional) !

Primary Data

H Required Field !

Figure 4.7 Help Menu Item and its Sub Menu Items

In Android, there is no way to show a PDF file directly from the app's asset folder. Also there is no library to facilitate this process. So the procedure I used to show the Help content in PDF format consists of these steps:

- During the installation of the app, it will check if the external storage is mounted or not and whether the space will be enough for copying the files from assets to external storage or not.
- If yes, the Help content files from the assets folder are copied to a folder in external storage of the device. Else, the user will be prompted for mounting an external storage.

- As there is always a chance of user deleting the contents on external storage, the app will check for the files every time it is launched. If any of these files are missing, the app will copy those missing files from the assets to the external storage, else it will be launched in normal way without doing any copy operation.

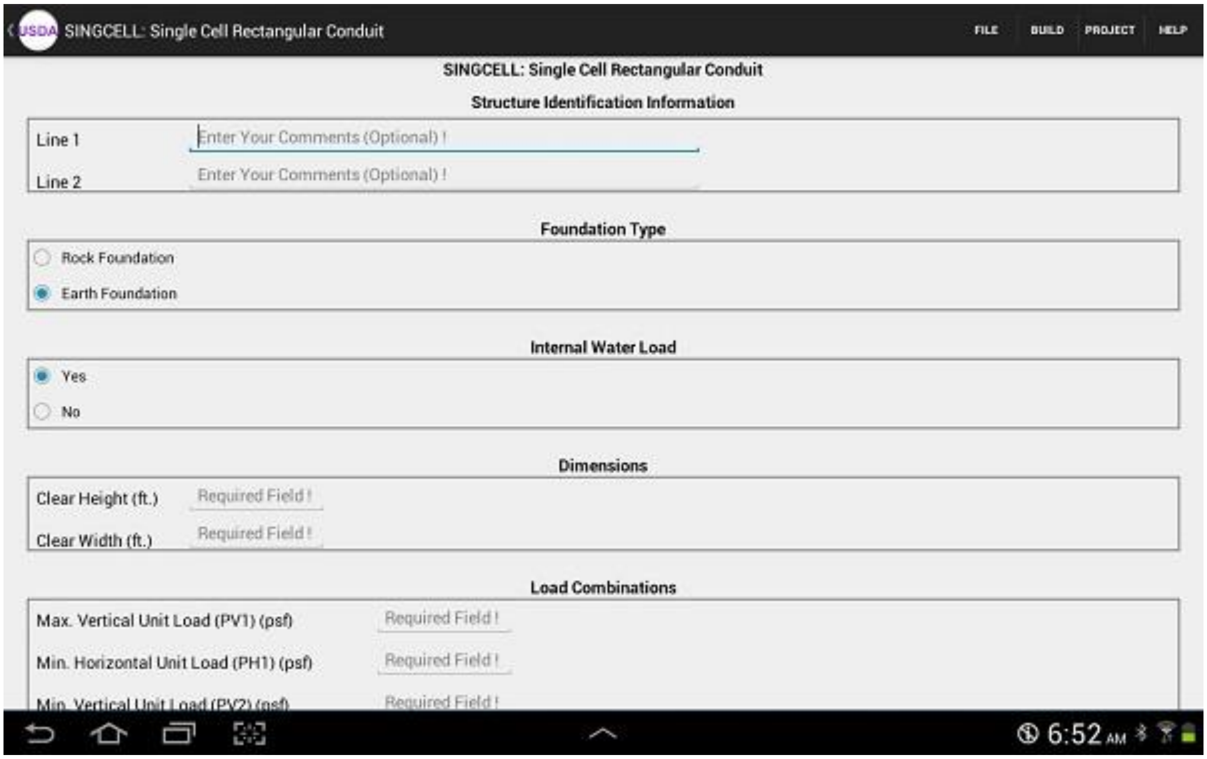
4.1.2.3 Single Cell Rectangular Conduit (SINGCELL) Project Screen

The Single Cell Rectangular Conduit Project Screen consists of the Input form for the Single Cell project [3] along with the title of the project and a dedicated space for Action Bar at the top of the screen. The input fields for this project screen include:

- Structure Identification Information (Line 1, Line 2) – Comment lines to identify the structure. These two input fields are optional.
- Foundation Type (Rock Foundation, Earth Foundation)
- Internal Water Load (Yes or No)
- Dimensions (Clear Height, Clear Width)
- Load Combinations (Max. Vertical Unit Load, Min. Horizontal Unit Load, Min. Vertical Unit Load, Max. Horizontal Unit Load)
- Email to send the Structural Design results (Optional)

Foundation Type and Internal Water Load together constitutes the Design Mode. A design Mode characterizes the conditions for which the conduit is designed. There are four design modes.

Earth Foundation, no internal water load = 00; Earth Foundation, with internal water load = 01;
Rock Foundation, no internal water load = 10; Rock Foundation, with internal water load = 11



SINGCELL: Single Cell Rectangular Conduit

Structure Identification Information

Line 1 Enter Your Comments (Optional) !

Line 2 Enter Your Comments (Optional) !

Foundation Type

○ Rock Foundation

● Earth Foundation

Internal Water Load

● Yes

○ No

Dimensions

Clear Height (ft.) Required Field !

Clear Width (ft.) Required Field !

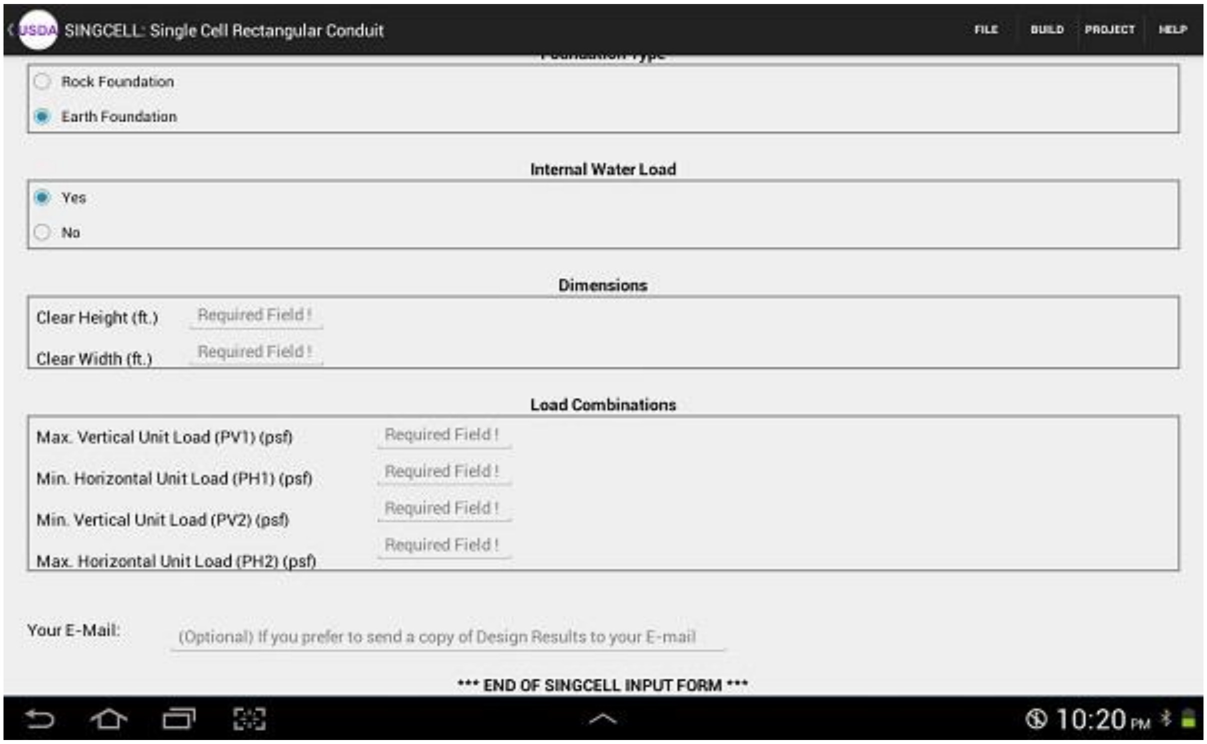
Load Combinations

Max. Vertical Unit Load (PV1) (psf) Required Field !

Min. Horizontal Unit Load (PH1) (psf) Required Field !

Min. Vertical Unit Load (PV2) (psf) Required Field !

Figure 4.8 Single Cell Project Screen - I



○ Rock Foundation

● Earth Foundation

Internal Water Load

● Yes

○ No

Dimensions

Clear Height (ft.) Required Field !

Clear Width (ft.) Required Field !

Load Combinations

Max. Vertical Unit Load (PV1) (psf) Required Field !

Min. Horizontal Unit Load (PH1) (psf) Required Field !

Min. Vertical Unit Load (PV2) (psf) Required Field !

Max. Horizontal Unit Load (PH2) (psf) Required Field !

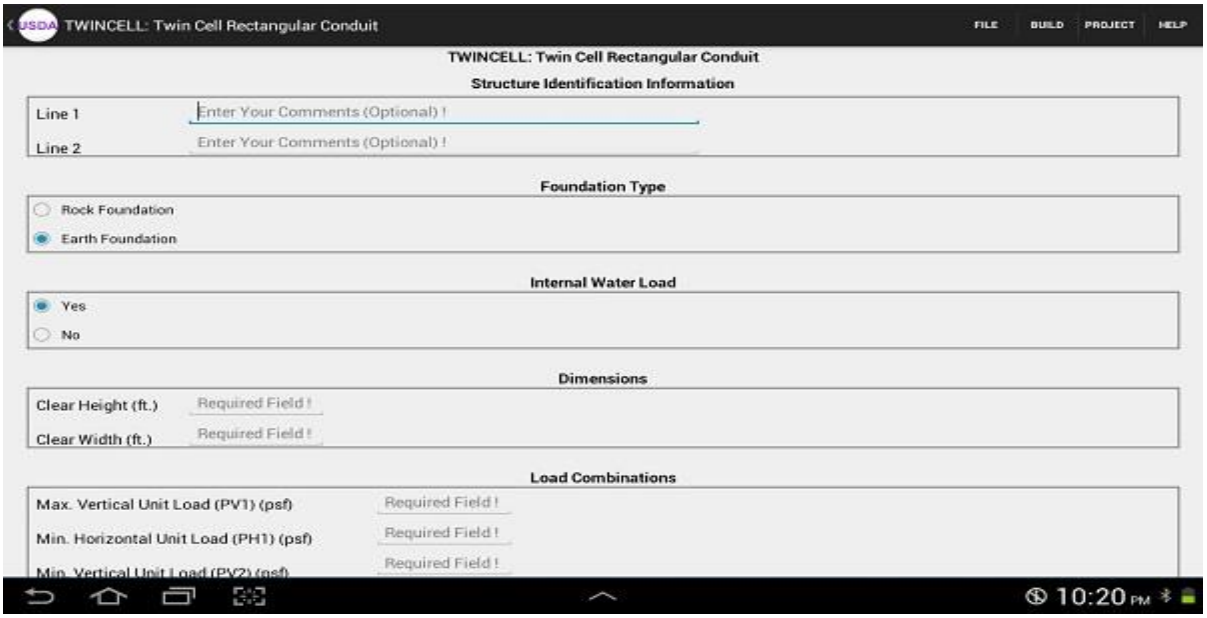
Your E-Mail: (Optional) If you prefer to send a copy of Design Results to your E-mail

*** END OF SINGCELL INPUT FORM ***

Figure 4.9 Single Cell Project Screen - II

4.1.2.4 Twin Cell Rectangular Conduit (TWINCELL) Project Screen

The Twin Cell Rectangular Conduit Project Screen consists of the Input form for the Twin Cell project [4] along with the title of the project and a dedicated space for Action Bar at the top of the screen. The input fields for this project screen and the design mode are similar to the Single Cell project Screen mentioned above.



TWINCELL: Twin Cell Rectangular Conduit

Structure Identification Information

Line 1 Enter Your Comments (Optional) !

Line 2 Enter Your Comments (Optional) !

Foundation Type

○ Rock Foundation

● Earth Foundation

Internal Water Load

● Yes

○ No

Dimensions

Clear Height (ft.) Required Field !

Clear Width (ft.) Required Field !

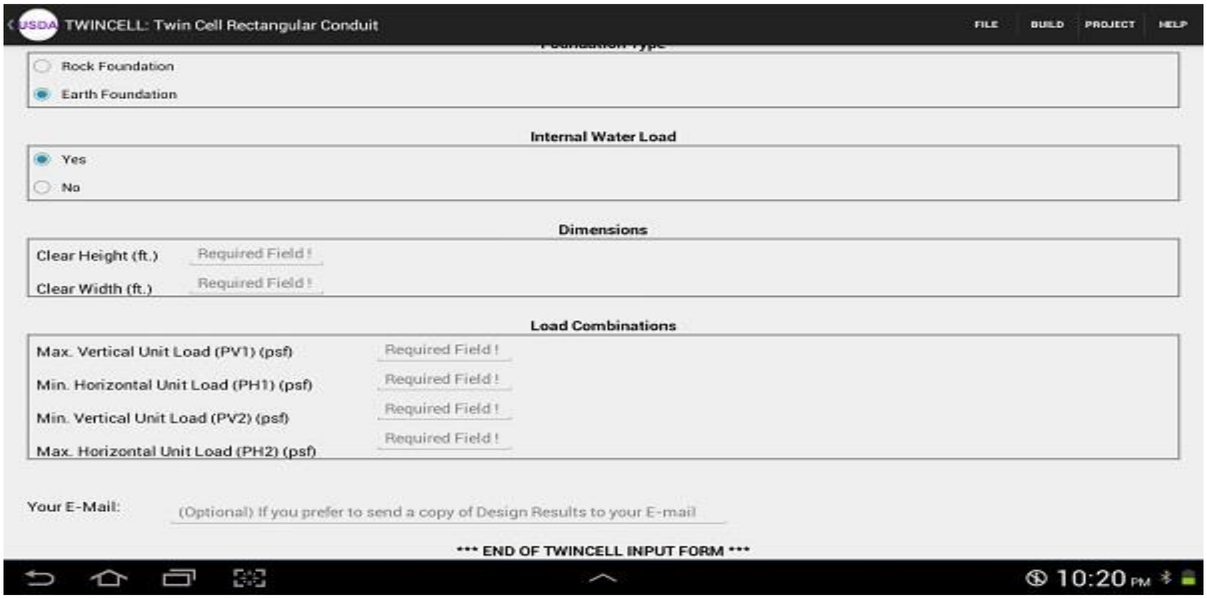
Load Combinations

Max. Vertical Unit Load (PV1) (psf) Required Field !

Min. Horizontal Unit Load (PH1) (psf) Required Field !

Min. Vertical Unit Load (PV2) (psf) Required Field !

Figure 4.10 Twin Cell Project Screen - I



○ Rock Foundation

● Earth Foundation

Internal Water Load

● Yes

○ No

Dimensions

Clear Height (ft.) Required Field !

Clear Width (ft.) Required Field !

Load Combinations

Max. Vertical Unit Load (PV1) (psf) Required Field !

Min. Horizontal Unit Load (PH1) (psf) Required Field !

Min. Vertical Unit Load (PV2) (psf) Required Field !

Max. Horizontal Unit Load (PH2) (psf) Required Field !

Your E-Mail: (Optional) If you prefer to send a copy of Design Results to your E-mail

*** END OF TWINCELL INPUT FORM ***

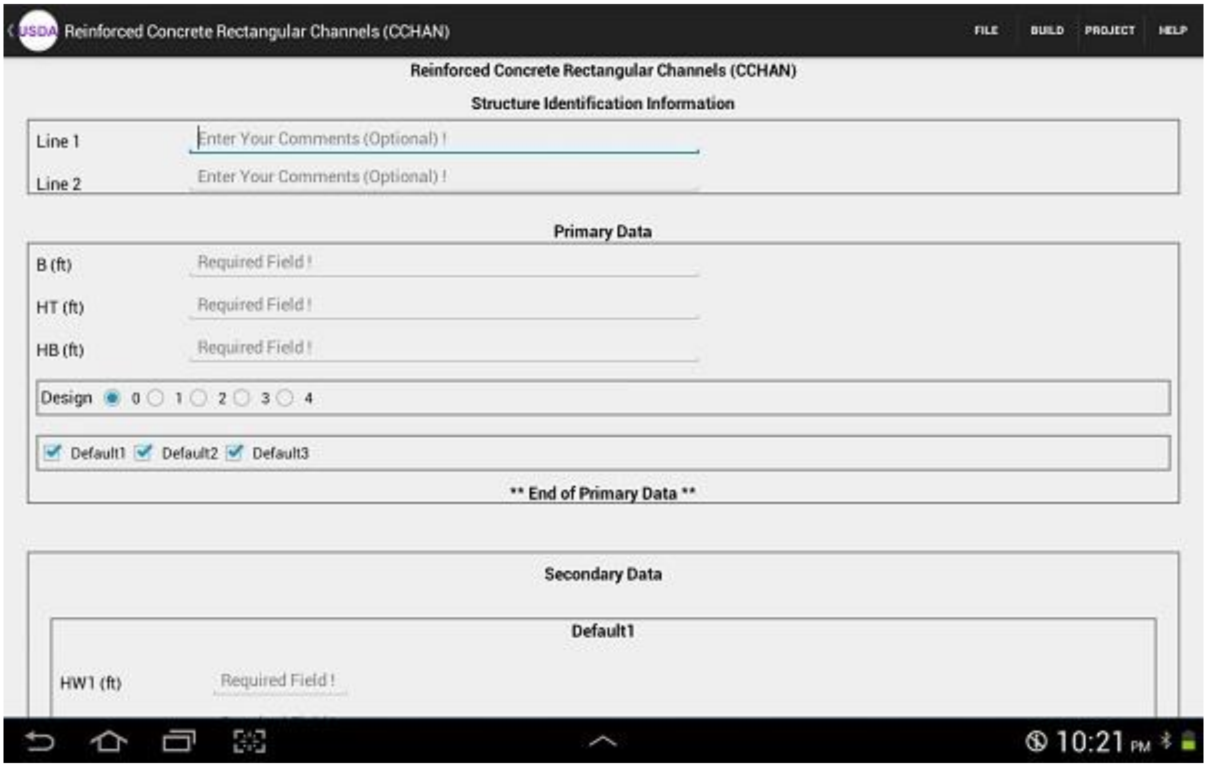
Figure 4.11 Twin Cell Project Screen - II

4.1.2.5 Reinforced Concrete Rectangular Channels (CCHAN) Project Screen

The CCHAN Project Screen consists of the Input form for the CCHAN project [5] along with the title of the project and a dedicated space for Action Bar at the top of the screen. The input fields for this project screen include:

- Structure Identification Information (Line 1, Line 1) – Comment Lines to identify the structure. These two input fields are optional.
- Primary Data (B, HT, HB, Design, Default1, Default2, Default3)
- Secondary Data (Default1, Default2, Default3)
- Default1 (HW1, HW2, HWP, GMOIST, GSAT, KO1, KO2, FLOATR)
- Default2 (MAXFTG, JOINTS, MFOUND)
- Default3 (CFSC, CFSS, KPASS)
- Email to send Structural Design results (Optional)

If Default 1 checkbox is checked, the input entry fields corresponding to Default 1 will be shown, otherwise they will not be shown. This will be similar for Default2 and Default3. If all of these checkboxes are not checked, then the Secondary Data will not be shown entirely.



Reinforced Concrete Rectangular Channels (CCHAN)

Structure Identification Information

Line 1 Enter Your Comments (Optional) !

Line 2 Enter Your Comments (Optional) !

Primary Data

B (ft) Required Field !

HT (ft) Required Field !

HB (ft) Required Field !

Design ● 0 ○ 1 ○ 2 ○ 3 ○ 4

☑ Default1 ☑ Default2 ☑ Default3

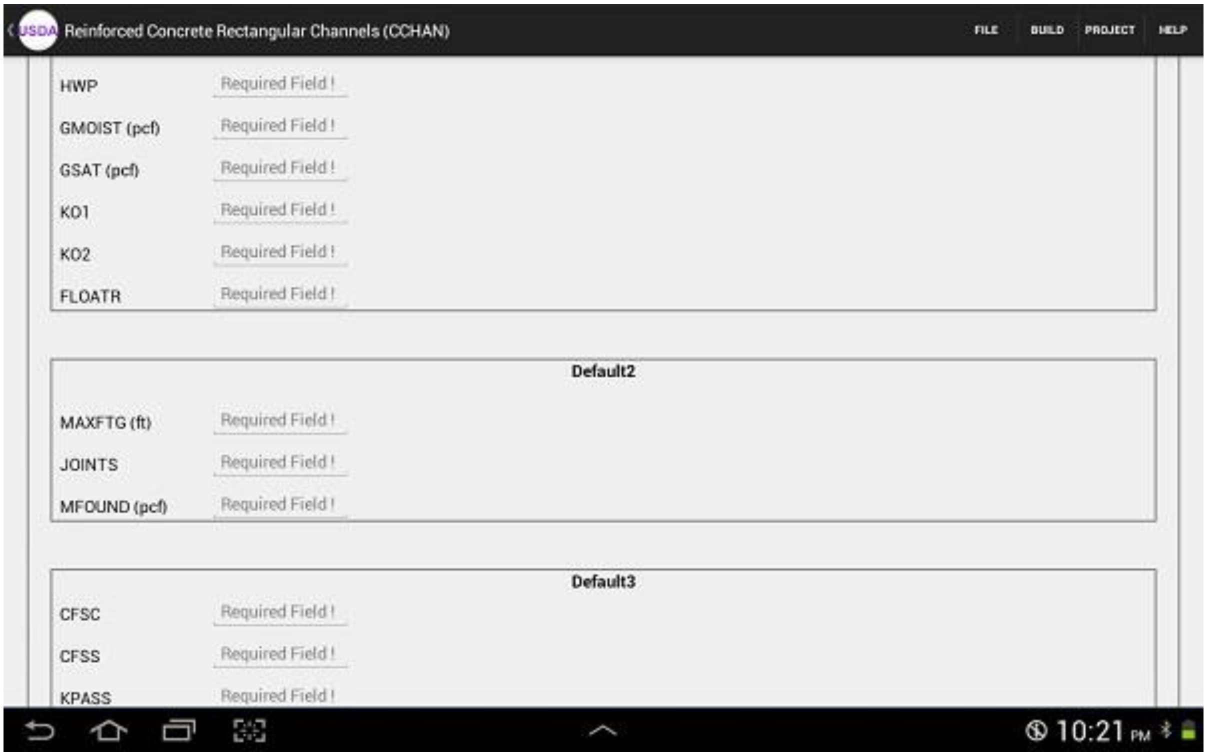
** End of Primary Data **

Secondary Data

Default1

HW1 (ft) Required Field !

Figure 4.12 CCHAN Project Screen - I



HWP	Required Field !
GMOIST (pcf)	Required Field !
GSAT (pcf)	Required Field !
KO1	Required Field !
KO2	Required Field !
FLOATR	Required Field !

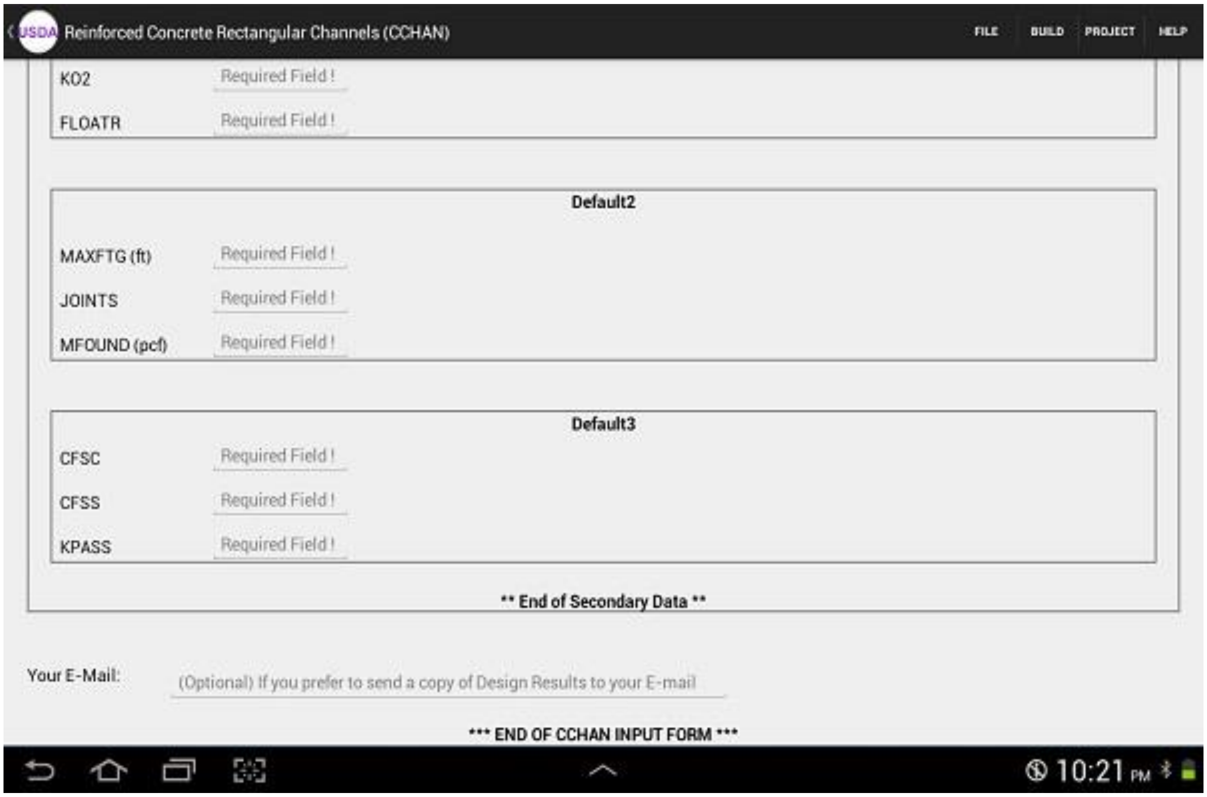
Default2

MAXFTG (ft)	Required Field !
JOINTS	Required Field !
MFOUND (pcf)	Required Field !

Default3

CFSC	Required Field !
CFSS	Required Field !
KPASS	Required Field !

Figure 4.13 CCHAN Project Screen - II



KO2	Required Field !
FLOATR	Required Field !

Default2

MAXFTG (ft)	Required Field !
JOINTS	Required Field !
MFOUND (pcf)	Required Field !

Default3

CFSC	Required Field !
CFSS	Required Field !
KPASS	Required Field !

** End of Secondary Data **

Your E-Mail: (Optional) If you prefer to send a copy of Design Results to your E-mail

*** END OF CCHAN INPUT FORM ***

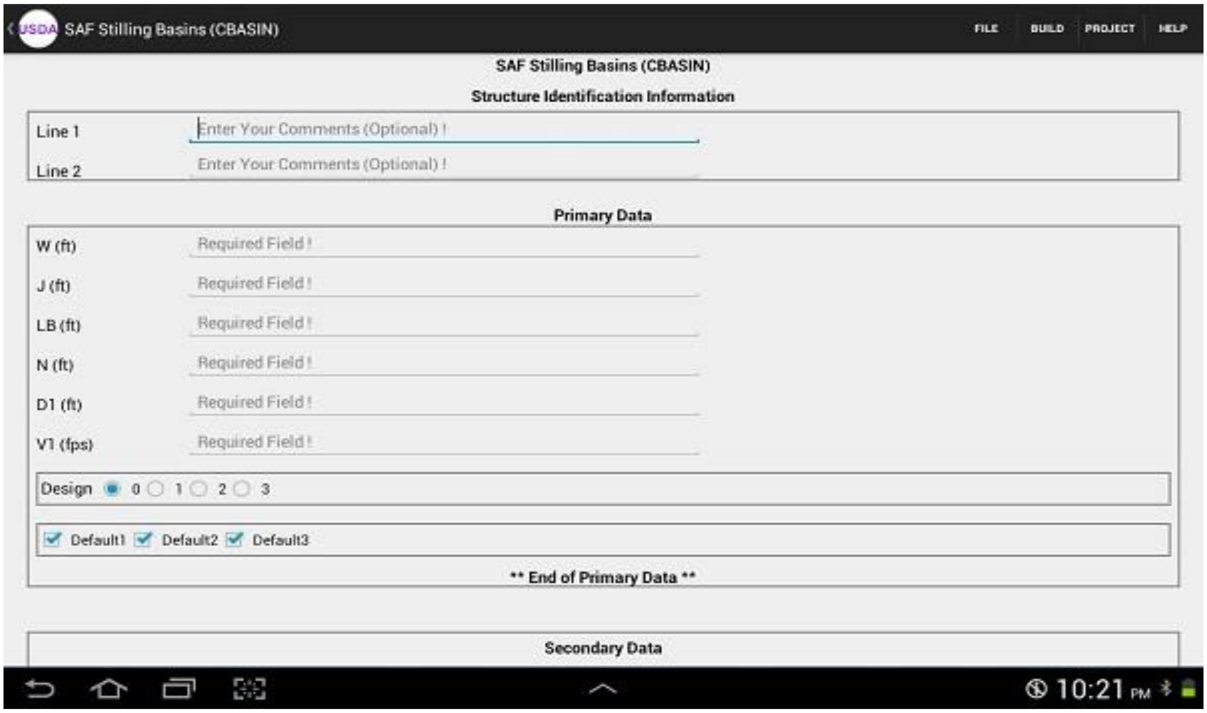
Figure 4.14 CCHAN Project Screen - III

4.1.2.6 SAF Stilling Basins (CBASIN) Project Screen

The CBASIN Project Screen consists of the Input form for the CBASIN project [6] along with the title of the project and a dedicated space for Action Bar at the top of the screen. The input fields for this project screen include:

- Structure Identification Information (Line 1, Line 1) – Comment Lines to identify the structure. These two input fields are optional.
- Primary Data (W, J, LB, N, D1, V1, Design, Default1, Default2, Default3)
- Secondary Data (Default1, Default2, Default3)
- Default1 (HB, HTW2, HUP2, HTW1, HUP1)
- Default2 (MAXFTG, FLOATR, SLIDER, ZS, BAT)
- Default3 (GM, GS, KO, CFSC, HTW, TTW)
- Email to send Structural Design results (Optional)

If Default 1 checkbox is checked, the input entry fields corresponding to Default 1 will be shown, otherwise they will not be shown. This will be similar for Default2 and Default3. If all of these checkboxes are not checked, then the Secondary Data will not be shown entirely.



SAF Stilling Basins (CBASIN)

Structure Identification Information

Line 1	Enter Your Comments (Optional) !
Line 2	Enter Your Comments (Optional) !

Primary Data

W (ft)	Required Field !
J (ft)	Required Field !
LB (ft)	Required Field !
N (ft)	Required Field !
D1 (ft)	Required Field !
V1 (fps)	Required Field !

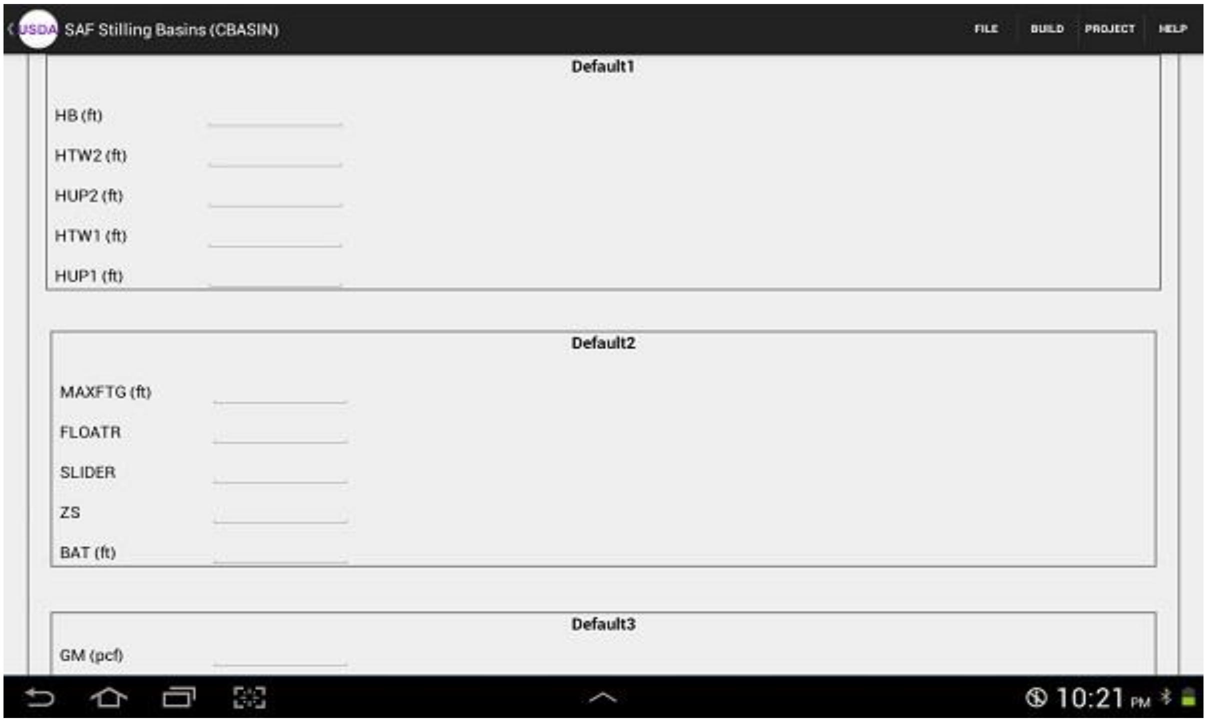
Design ● 0 ○ 1 ○ 2 ○ 3

☑ Default1 ☑ Default2 ☑ Default3

** End of Primary Data **

Secondary Data

Figure 4.15 CBASIN Project Screen - I



Default1

HB (ft)

HTW2 (ft)

HUP2 (ft)

HTW1 (ft)

HUP1 (ft)

Default2

MAXFTG (ft)

FLOATR

SLIDER

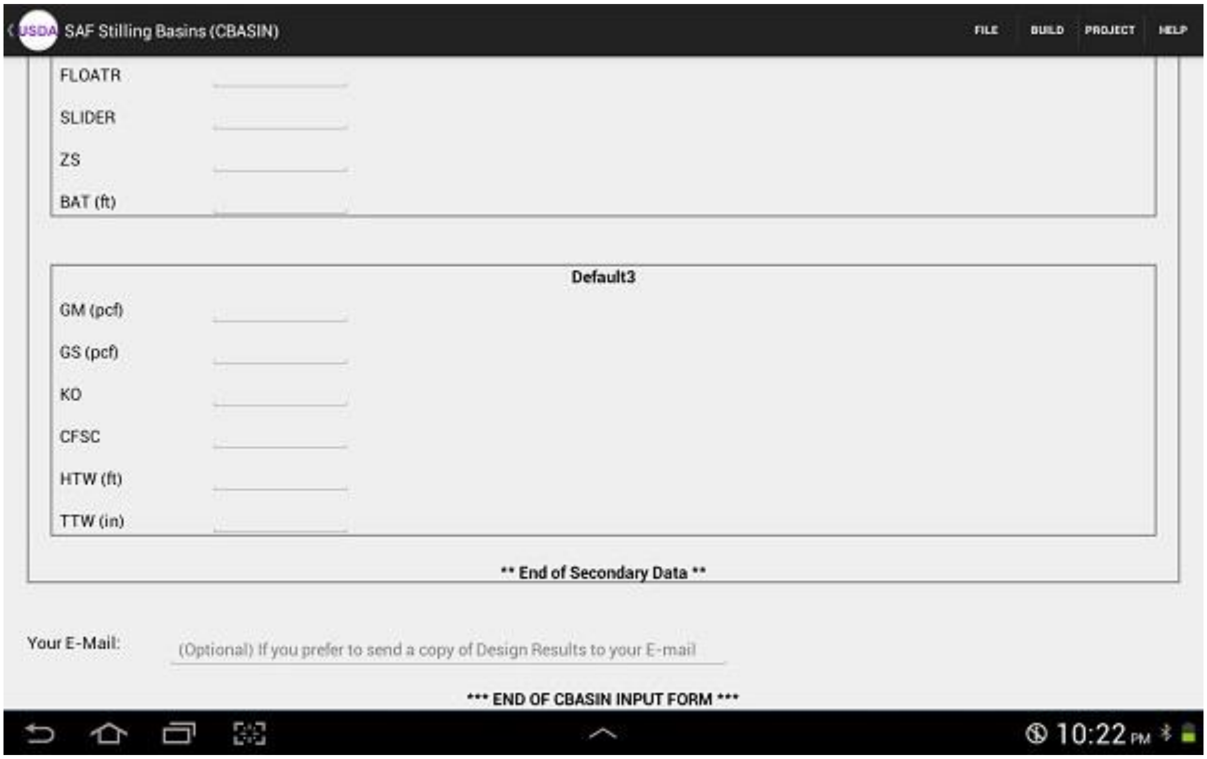
ZS

BAT (ft)

Default3

GM (pcf)

Figure 4.16 CBASIN Project Screen - II



FLOATR

SLIDER

ZS

BAT (ft)

Default3

GM (pcf)

GS (pcf)

KO

CFSC

HTW (ft)

TTW (in)

** End of Secondary Data **

Your E-Mail: (Optional) If you prefer to send a copy of Design Results to your E-mail

*** END OF CBASIN INPUT FORM ***

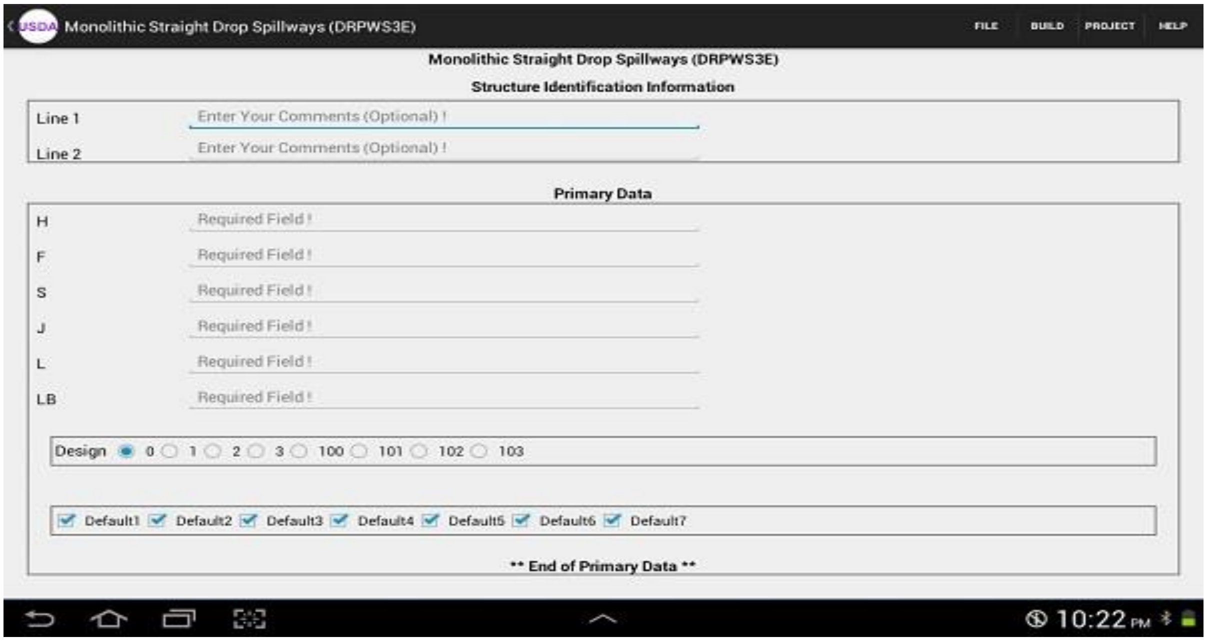
Figure 4.17 CBASIN Project Screen - III

4.1.2.6 Monolithic Straight Drop Spillways (DRPWS3E) Project Screen

The DRPWS3E Project Screen consists of the Input form for the DRPWS3E project [7] along with the title of the project and a dedicated space for Action Bar at the top of the screen. The input fields for this project screen include:

- Structure Identification Information (Line 1, Line 1) – Comment Lines to identify the structure. These two input fields are optional.
- Primary Data (H, F, S, J, L, LB, Design, Default1, Default2, Default3, Default4, Default5, Default6, Default 7)
- Secondary Data (Default1, Default2, Default3, Default4, Default5, Default6, Default 7, User Interactive Switches)
- Default1 (CREEPR, FLOATR, SLIDER, BAT, SWLDRN)
- Default2 (HB, ZPS, HTOE, TTOE, CFSS, CFSC)
- Default3 (KOH, GMH, GSH)
- Default4 (KOF, GMF, GSF, KPF)
- Default5 (KOW, GMW, GSW, KPW)
- Default6 (DW2, HEAD2, TAIL2, HEAD1)
- Default6 (DWM2, DWM3, DWM4, DWM5, HEADM2, HEADM3, HEADM4, HEADM5, TAILM2, TAILM3, TAILM4, TAILM5)
- User Interactive Switches (R-C Design Method, Material Properties, Other Options)
- R-C Design Method (Working Stress Design, Strength Design)
- Material Properties (Soil Conservation Default Values, Corps of Engineers Default Values, User Supplied Values)
- Other Options (Floatation Criteria, Moment/Thrust/Shear Report, Full or Summary Report)
- Floatation Criteria (Soil Conservation Service, Corps of Engineers)
- Moment/Thrust/Shear Report (Yes, No)
- Full or Summary Report (Full, Summary)
- Email to send Structural Design results (Optional)

If Default 1 checkbox is checked, the input entry fields corresponding to Default 1 will be shown, otherwise they will not be shown. This will be similar for Default2, Default3, Default4, Default5, Default6, and Default7.



Monolithic Straight Drop Spillways (DRPWS3E)

Monolithic Straight Drop Spillways (DRPWS3E)

Structure Identification Information

Line 1 Enter Your Comments (Optional) !

Line 2 Enter Your Comments (Optional) !

Primary Data

H Required Field !

F Required Field !

S Required Field !

J Required Field !

L Required Field !

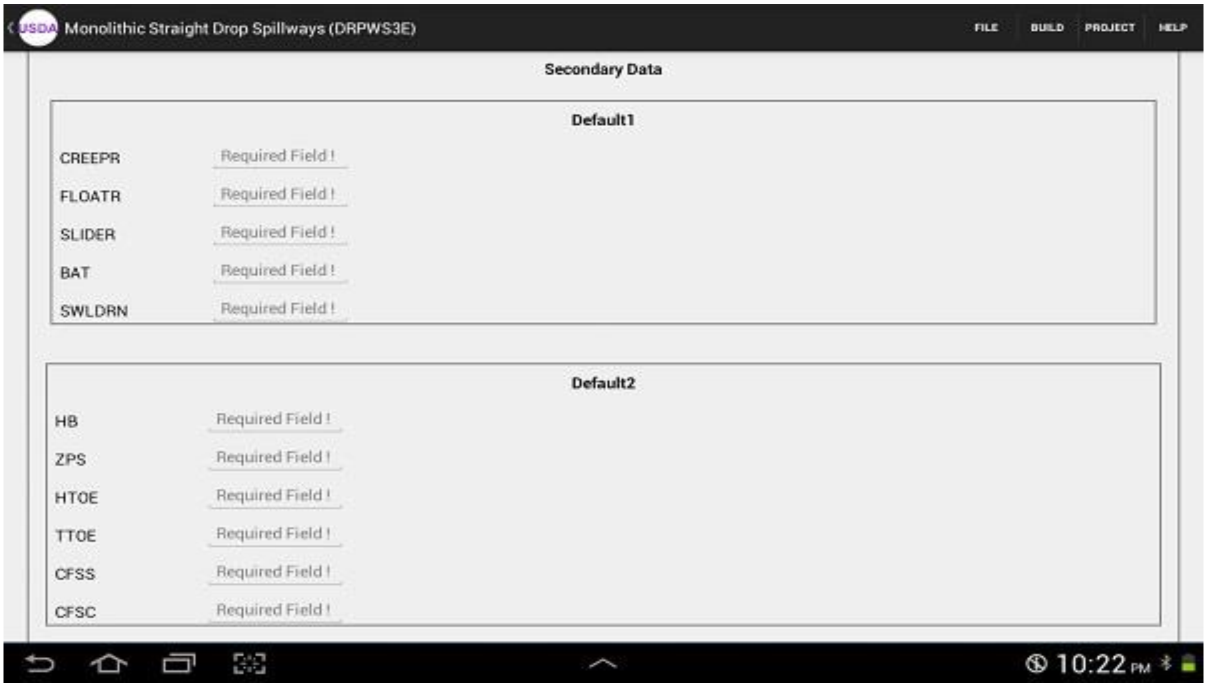
LB Required Field !

Design ● 0 ○ 1 ○ 2 ○ 3 ○ 100 ○ 101 ○ 102 ○ 103

☑ Default1 ☑ Default2 ☑ Default3 ☑ Default4 ☑ Default5 ☑ Default6 ☑ Default7

** End of Primary Data **

Figure 4.18 DRPWS3E Project Screen - I



Monolithic Straight Drop Spillways (DRPWS3E)

Secondary Data

Default1

CREEPR Required Field !

FLOATR Required Field !

SLIDER Required Field !

BAT Required Field !

SWLDRN Required Field !

Default2

HB Required Field !

ZPS Required Field !

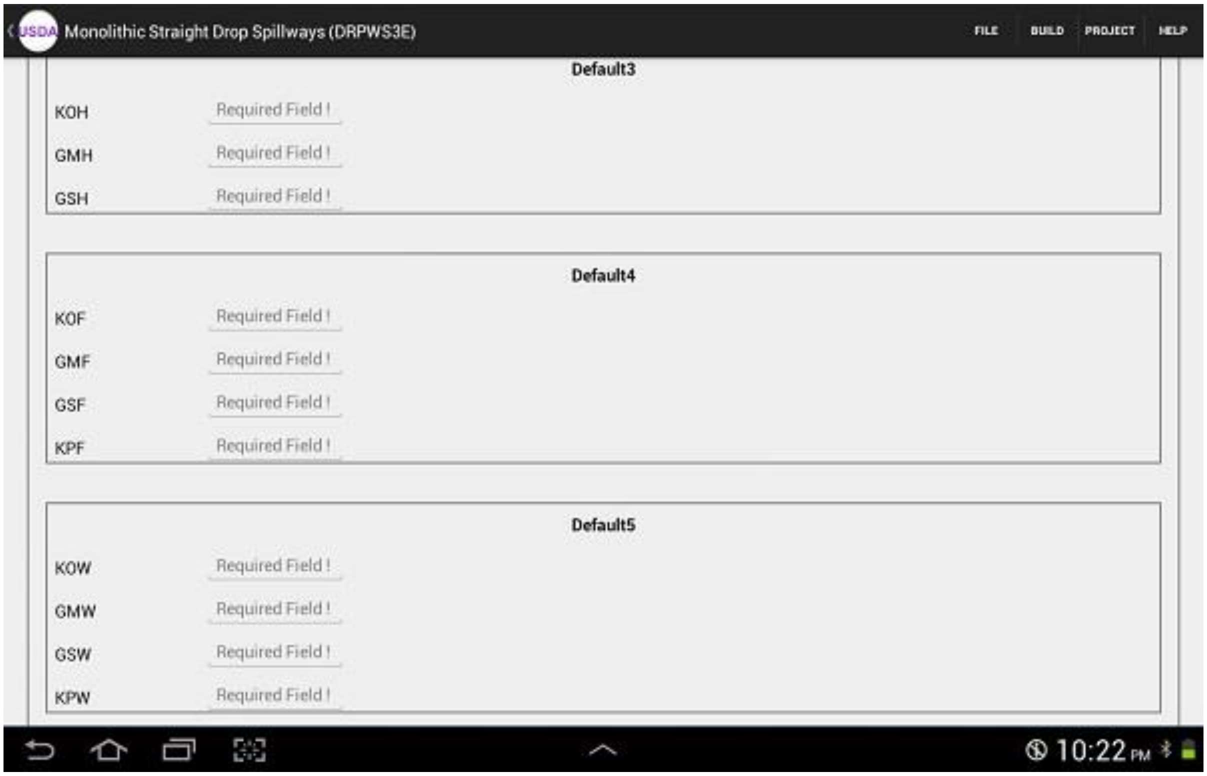
HTOE Required Field !

TTOE Required Field !

CFSS Required Field !

CFSC Required Field !

Figure 4.19 DRPWS3E Project Screen - II



Default3

KOH	Required Field !
GMH	Required Field !
GSH	Required Field !

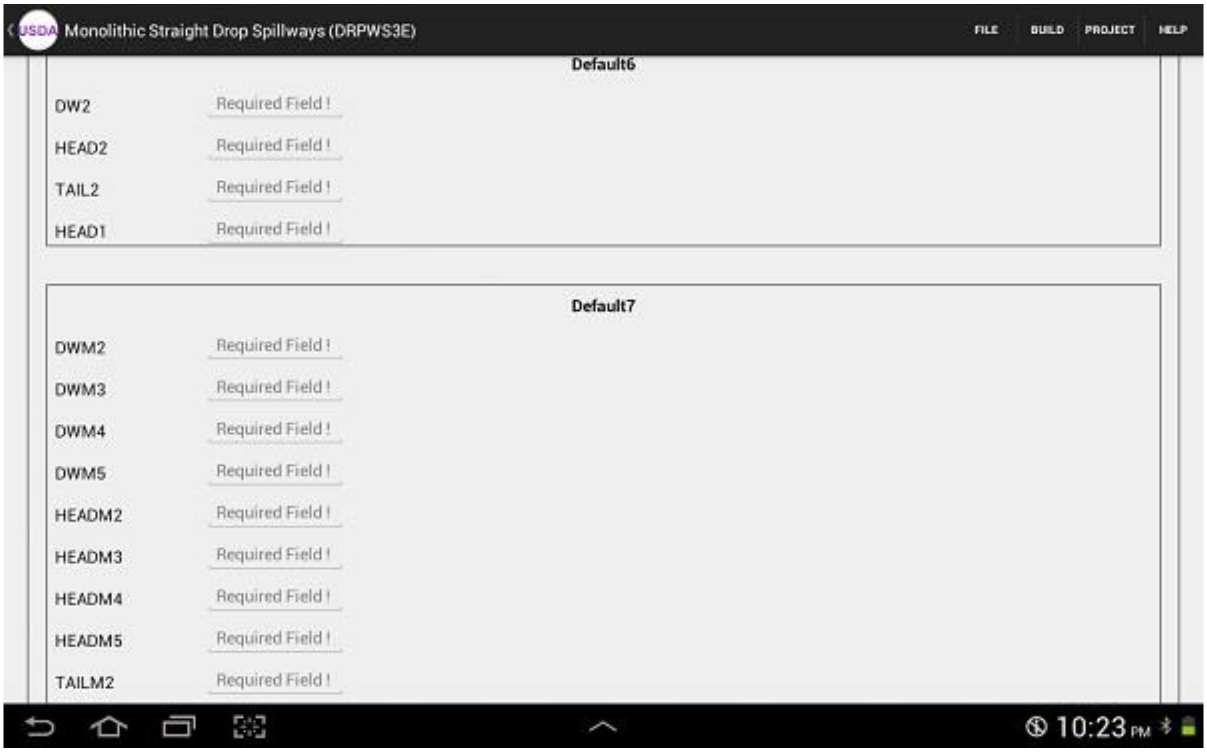
Default4

KOF	Required Field !
GMF	Required Field !
GSF	Required Field !
KPF	Required Field !

Default5

KOW	Required Field !
GMW	Required Field !
GSW	Required Field !
KPW	Required Field !

Figure 4.20 DRPWS3E Project Screen - III



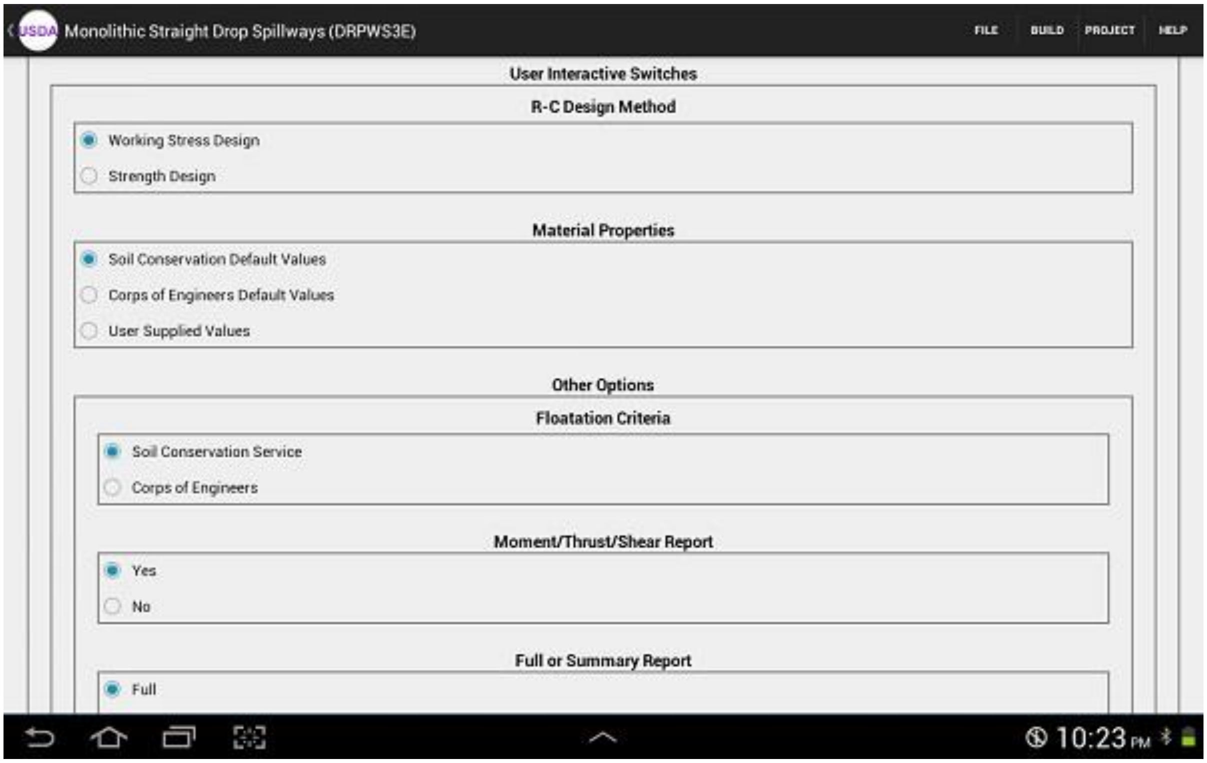
Default6

DW2	Required Field !
HEAD2	Required Field !
TAIL2	Required Field !
HEAD1	Required Field !

Default7

DWM2	Required Field !
DWM3	Required Field !
DWM4	Required Field !
DWM5	Required Field !
HEADM2	Required Field !
HEADM3	Required Field !
HEADM4	Required Field !
HEADM5	Required Field !
TAILM2	Required Field !

Figure 4.21 DRPWS3E Project Screen - IV



User Interactive Switches

R-C Design Method

● Working Stress Design
○ Strength Design

Material Properties

● Soil Conservation Default Values
○ Corps of Engineers Default Values
○ User Supplied Values

Other Options

Floatation Criteria

● Soil Conservation Service
○ Corps of Engineers

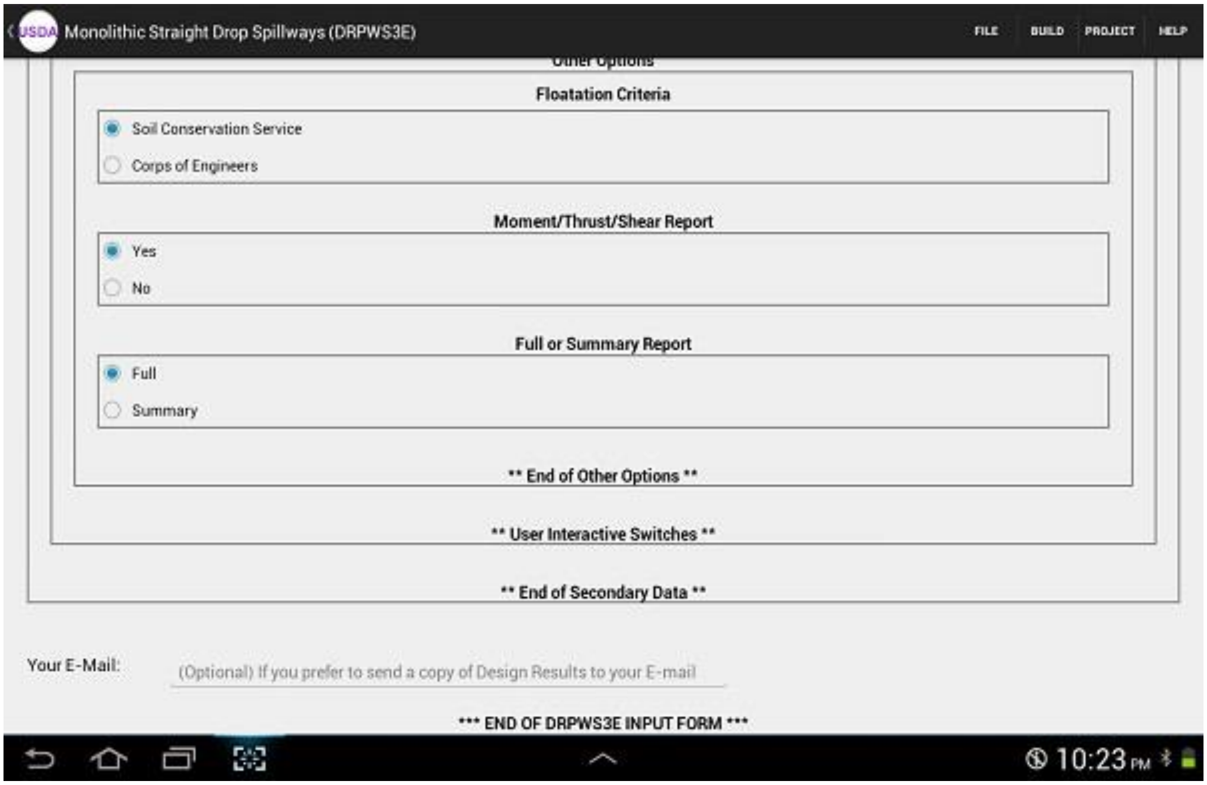
Moment/Thrust/Shear Report

● Yes
○ No

Full or Summary Report

● Full

Figure 4.22 DRPWS3E Project Screen - V



Other Options

Floatation Criteria

● Soil Conservation Service
○ Corps of Engineers

Moment/Thrust/Shear Report

● Yes
○ No

Full or Summary Report

● Full
○ Summary

** End of Other Options **

** User Interactive Switches **

** End of Secondary Data **

Your E-Mail: (Optional) If you prefer to send a copy of Design Results to your E-mail

*** END OF DRPWS3E INPUT FORM ***

Figure 4.23 DRPWS3E Project Screen - VI

4.2 Testing

4.2.1 FORTRAN to Java Translation

Manual Testing has been adopted to test and ensure the correctness of Java source files obtained from the FORTRAN to Java translation process. Manual Testing is the process of manually testing the software for defects.

Manual Testing for the Java class files obtained from FORTRAN to Java translation is performed by comparing the results obtained by executing the Java source files with the results obtained from the FORTRAN executable. This type of testing has been performed on different input sets

4.2.2 Android Application

4.2.2.1 Logger and Debugger

Android provides a default logging and debugging tools called Dalvik Debug Monitor Server (DDMS). DDMS works with both the Emulator and the Android device and it gives a complete view of all the system messages such as stack traces, print statements, etc., - using logcat.

4.2.2.2 Performance

This application is manually tested for performance based on the transition times from one activity to another. Transition time is the time span between the click of a button and the time at which the activity screen is displayed to the user. The following table displays the average transition times for the different screens.

Screen	Transition Time
Single Cell Project Screen	0.35
Twin Cell Project Screen	0.35
Cchan Project Screen	0.42
Cbasin Project Screen	0.46
Drpws3e Project Screen	0.55
File Open Screen	0.28
File Save Screen	0.28

Table 4.2 Transition Times of Screens

4.2.2.3 Unit Testing

Unit testing involves testing of each module of the application individually without the interference of the outside world. This application is unit tested manually with the help of Samsung Tab 10.1. The following test cases were used to perform the Unit Testing.

No.	Test Case	Expected Result	Result
1	Testing the buttons Single Cell, Twin Cell, CCHAN, CBASIN, DRPWS3E on the Home Screen	Single Cell button opens Single Cell Project Screen, Twin Cell button opens Twin Cell Project Screen, CCHAN button opens CCHAN Project Screen, CBASIN button opens CBASIN Project Screen and DRPWS3E button opens DRPWS3E Project Screen	Pass
2	Loading of Single Cell Project Screen	Display Single Cell Input form with Action Bar and Title of the screen at the top	Pass
3	Loading of Twin Cell Project Screen	Display Twin Cell Input form with Action Bar and Title of the screen at the top	Pass
4	Loading of CCHAN Project Screen	Display CCHAN Input form with Action Bar and Title of the screen at the top	Pass
5	Loading of CBASIN Project Screen	Display CBASIN Input form with Action Bar and Title of the screen at the top	Pass
6	Loading of DRPWS3E Project Screen	Display DRPWS3E Input form with Action Bar and Title of the screen at the top	Pass

7	Functionality of Sub Menu Item “New” of Menu Item “File” of the Action Bar	Clears all the Input fields, sets Radio buttons and Checkboxes to default values	Pass
8	Functionality of Sub Menu Item “Exit” of Menu Item “File” of the Action Bar	Exits the Current Project Screen	Pass
9	Required field validations when trying to build project on any of the Project Screens	Prompts user with Build Error alert dialog when all the required fields are not entered	Pass
10	Hiding Default1, Default2, Default3 input fields when checkbox value changed from checked to not checked state on CCHAN Project Screen	Hides Default1, Default2, Default3 input fields when respective check box is not checked	Pass
11	Showing Default1, Default2, Default3 input fields when checkbox value changed from not checked to checked state on CCHAN Project Screen	Shows Default1, Default2, Default3 input fields when respective check box is checked	Pass
12	Hiding Default1, Default2, Default3 input fields when checkbox value changed from checked to not checked state on CBASIN Project Screen	Hides Default1, Default2, Default3 input fields when respective check box is not checked	Pass
13	Showing Default1, Default2, Default3 input fields when checkbox value changed from not checked to checked state on CBASIN Project Screen	Shows Default1, Default2, Default3 input fields when respective check box is checked	Pass
14	Hiding Default1, Default2, Default3, Default4, Default5,	Hides Default1, Default2, Default3, Default4, Default5,	Pass

	Default6, Default7 input fields when checkbox value changed from checked to not checked state on DRPWS3E Project Screen	Default6, Default7 input fields when respective check box is not checked	
15	Showing Default1, Default2, Default3, Default4, Default5, Default6, Default7 input fields when checkbox value changed from not checked to checked state on DRPWS3E Project Screen	Shows Default1, Default2, Default3, Default4, Default5, Default6, Default7 input fields when respective check box is not checked	Pass

Table 4.3 Unit Test Cases

4.2.2.4 Integration Testing

In Integration Testing, individual modules are combined together and tested as a group to check if they are working successfully on integration or not. Integration testing is performed with the help of the following test cases to make sure the navigation among the modules is working correctly.

No.	Screen	Expected Result	Result
1	Loading of “File Open” Window when Sub Menu Item “Open” in “File” Menu is clicked on any of the Project Screens	Loads a file open window with all the folders and files on the external storage listed	Pass
2	Loading of “File Save” Window when Sub Menu Item “Save As” in “File” Menu is clicked on any of the Project Screens	Loads a file Save window with all the folders and files on the external storage listed	Pass

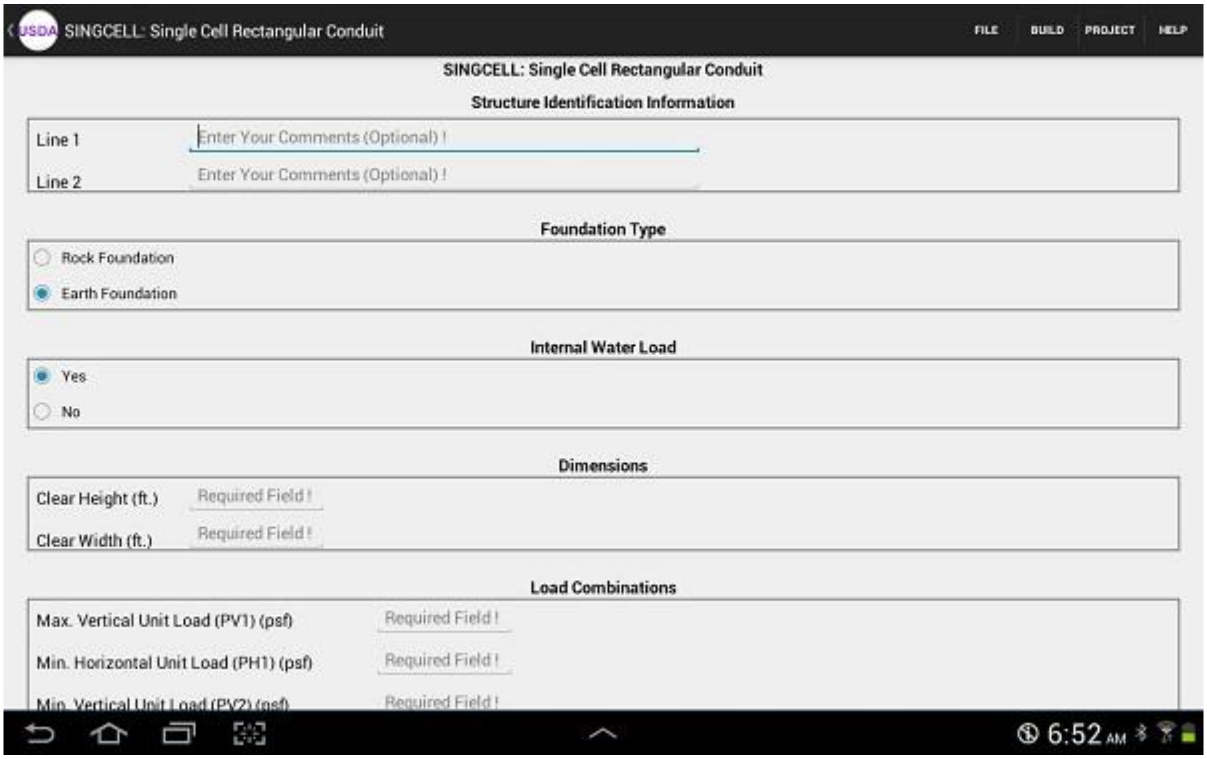
3	Functionality of “Open” Sub Menu Item of “File” Menu of the Action Bar on any of the Project Screens	Opens a “File Open” Window and retrieves the input from the text file selected from a specified location on the external storage	Pass
4	Functionality of “Save As” Sub Menu Item of “File” Menu of the Action Bar on any of the Project Screens	Opens a “File Save” Window and save the input filled in the form in a text file on a specified location on external storage	Pass
5	Build Structural Design on any of the Project Screens	Builds Structural Design, Prompts user for an external PDF viewer intent to show the results and prompts the user for an external email intent if email input field is entered	Pass
6	Displays correct Structural Design results for any of the Project Types	Displays correct Structural Design results when compared with the results obtained using Fortran executable’s	Pass
7	Navigation between different project screens	Navigates from any of the project screen to the other project screen selected in	Pass

		"Project" Menu Item of the Action Bar	
8	Displaying Help Content for the project selected in PDF format	Prompts user for an external PDF intent to show the Help content for any of the projects selected in the "Help" menu Item of the Action Bar	Pass

Table 4.4 Integration Testing Test Cases

4.2.2.5 Compatibility Testing

The application has been tested in both Portrait and Landscapes modes and will work best in both modes for any of the Android powered tablet devices. The Application has been tested for different screen sizes as a part of compatibility testing. This application has been basically designed targeting android tablets. But it may work well even with the Android mobile phones due to the uniform layout used during the development of screens. Because of the requirement of lot of input entry from the users, this application will work best with tablet devices compared to mobile phones.



SINGCELL: Single Cell Rectangular Conduit

Structure Identification Information

Line 1 Enter Your Comments (Optional) !

Line 2 Enter Your Comments (Optional) !

Foundation Type

○ Rock Foundation

● Earth Foundation

Internal Water Load

● Yes

○ No

Dimensions

Clear Height (ft.) Required Field !

Clear Width (ft.) Required Field !

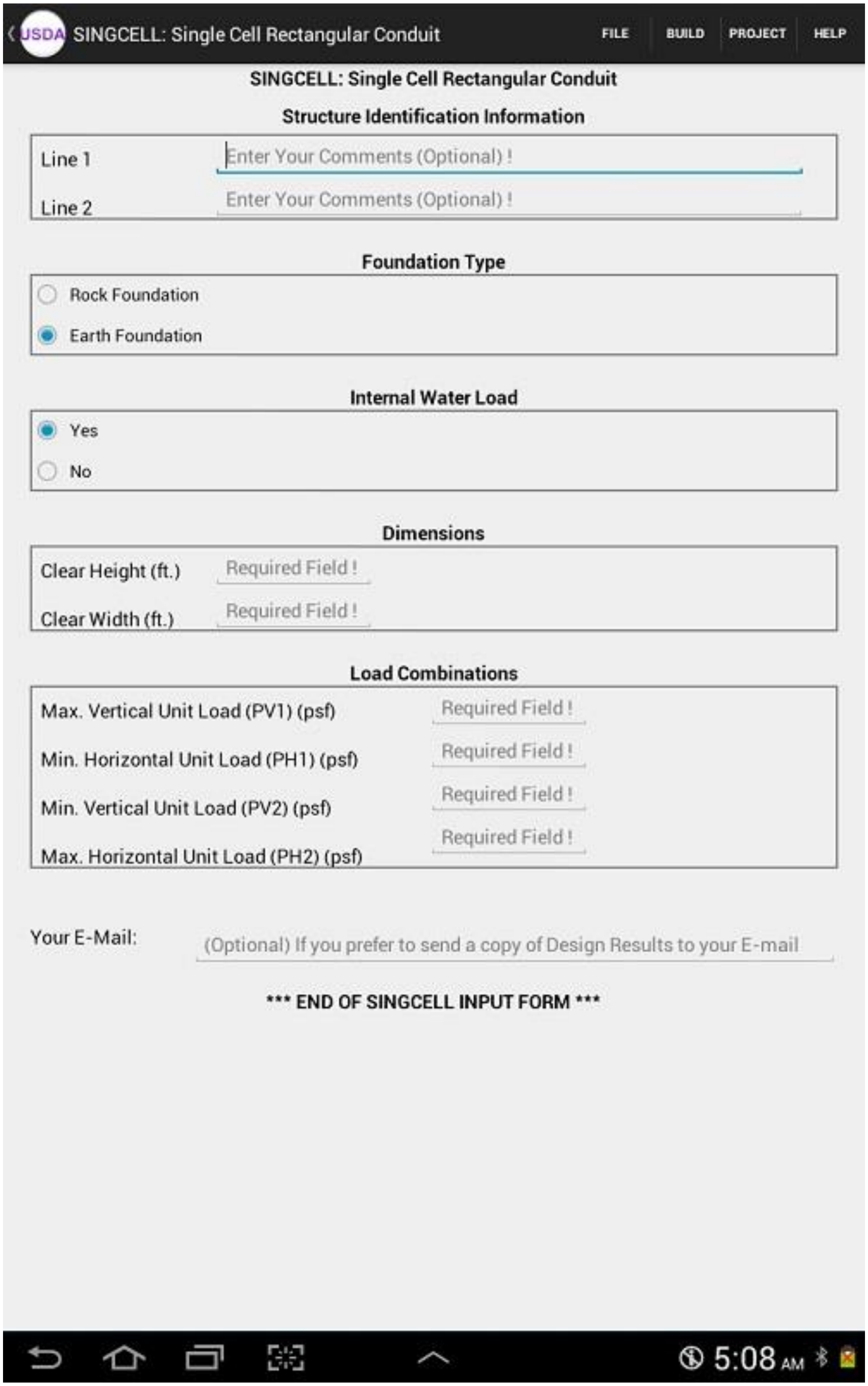
Load Combinations

Max. Vertical Unit Load (PV1) (psf) Required Field !

Min. Horizontal Unit Load (PH1) (psf) Required Field !

Min. Vertical Unit Load (PV2) (psf) Required Field !

Figure 4.24 Single Cell Project Screen in Landscape Mode



SINGCELL: Single Cell Rectangular Conduit

Structure Identification Information

Line 1 Enter Your Comments (Optional) !

Line 2 Enter Your Comments (Optional) !

Foundation Type

○ Rock Foundation

● Earth Foundation

Internal Water Load

● Yes

○ No

Dimensions

Clear Height (ft.) Required Field !

Clear Width (ft.) Required Field !

Load Combinations

Max. Vertical Unit Load (PV1) (psf) Required Field !

Min. Horizontal Unit Load (PH1) (psf) Required Field !

Min. Vertical Unit Load (PV2) (psf) Required Field !

Max. Horizontal Unit Load (PH2) (psf) Required Field !

Your E-Mail: (Optional) If you prefer to send a copy of Design Results to your E-mail

*** END OF SINGCELL INPUT FORM ***

Figure 4.25 Single Cell Project Screen in Portrait Mode

4.3 Results

The current application displays the simulation results in a PDF format for the projects Single Cell [3], Twin Cell [4], CCHAN [5], CBASIN [6], and DRPWS3E [7]. A sample input and structural design simulation result for CCHAN project is shown below.

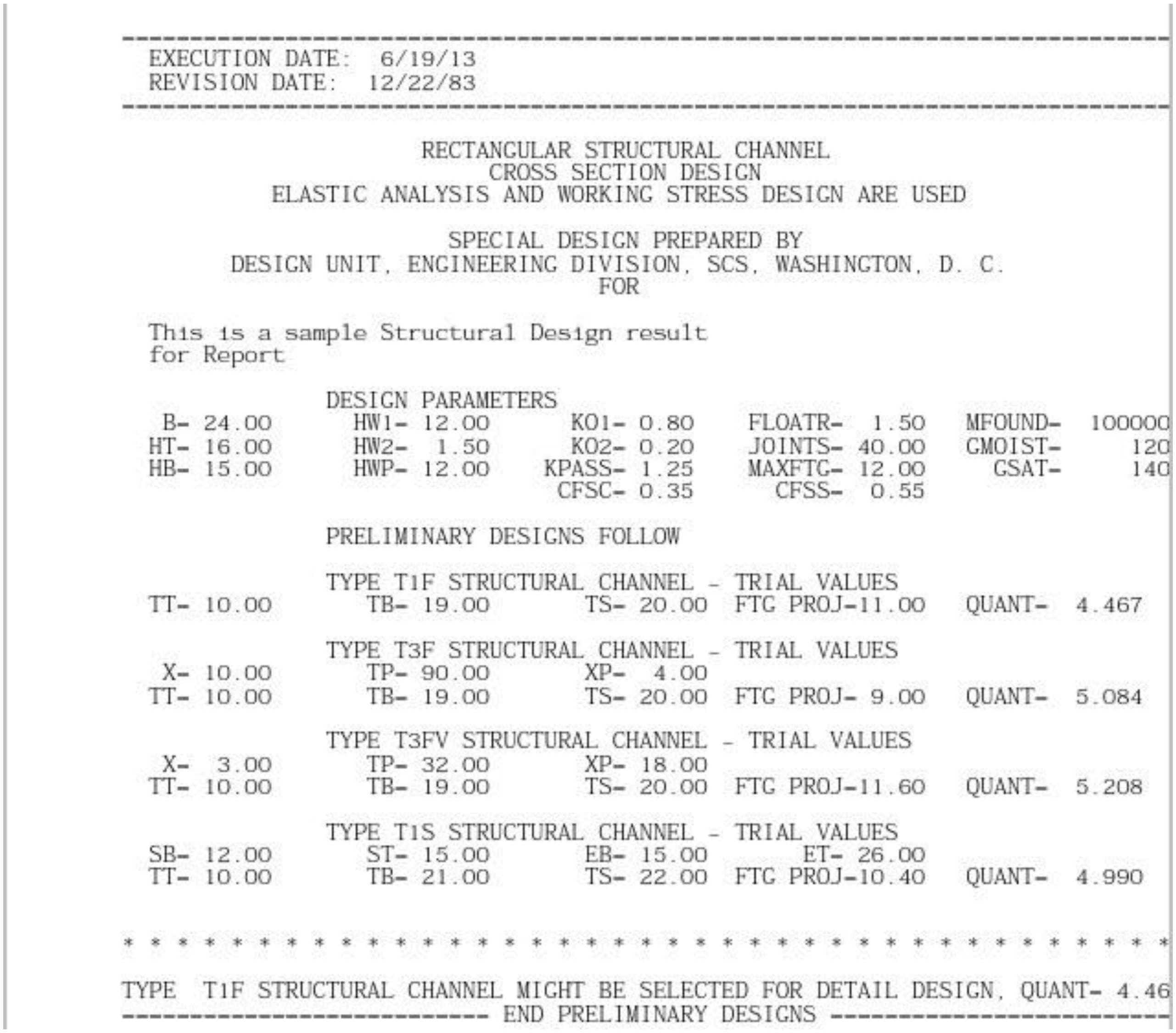
Input (for CCHAN)

This is a sample Structural Design result

For Report

24 16 15 0 0 0 0

Structural Design Simulation Result



```
------------------------------------------------------------------------------
  EXECUTION DATE:  6/19/13
  REVISION DATE:  12/22/83
------------------------------------------------------------------------------

                      RECTANGULAR STRUCTURAL CHANNEL
                           CROSS SECTION DESIGN
              ELASTIC ANALYSIS AND WORKING STRESS DESIGN ARE USED

                          SPECIAL DESIGN PREPARED BY
           DESIGN UNIT, ENGINEERING DIVISION, SCS, WASHINGTON, D. C.
                                     FOR

  Th1s 1s a sample Structural Des1gn result
  for Report

               DESIGN PARAMETERS
   B- 24.00      HW1- 12.00      KO1- 0.80    FLOATR-  1.50   MFOUND-  100000
  HT- 16.00      HW2-  1.50      KO2- 0.20    JOINTS- 40.00   GMOIST-     120
  HB- 15.00      HWP- 12.00    KPASS- 1.25    MAXFTC- 12.00     GSAT-     140
                                CFSC- 0.35      CFSS-  0.55

               PRELIMINARY DESIGNS FOLLOW

               TYPE T1F STRUCTURAL CHANNEL - TRIAL VALUES
  TT- 10.00       TB- 19.00       TS- 20.00  FTC PROJ-11.00   QUANT-  4.467

               TYPE T3F STRUCTURAL CHANNEL - TRIAL VALUES
   X- 10.00       TP- 90.00       XP-  4.00
  TT- 10.00       TB- 19.00       TS- 20.00  FTC PROJ- 9.00   QUANT-  5.084

               TYPE T3FV STRUCTURAL CHANNEL - TRIAL VALUES
   X-  3.00       TP- 32.00       XP- 18.00
  TT- 10.00       TB- 19.00       TS- 20.00  FTC PROJ-11.60   QUANT-  5.208

               TYPE T1S STRUCTURAL CHANNEL - TRIAL VALUES
  SB- 12.00       ST- 15.00       EB- 15.00       ET- 26.00
  TT- 10.00       TB- 21.00       TS- 22.00  FTC PROJ-10.40   QUANT-  4.990

* * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * *

TYPE  T1F STRUCTURAL CHANNEL MIGHT BE SELECTED FOR DETAIL DESIGN, QUANT- 4.46
-------------------------- END PRELIMINARY DESIGNS ---------------------------
```

Figure 4.26 Sample Structural Design Simulation Result for CCHAN

Chapter 5 - Conclusion and Future Work

5.1 Conclusion

The application is successfully implemented and tested on real Android devices. The Java class files for DRPWS3E project obtained from FORTRAN to Java translation are executed on different sets of inputs and are compared with the results obtained using the FORTRAN executable. This comparison verifies the accuracy of the java class files obtained.

Being a first attempt in developing an android application, it gave me good exposure to learn the basic understanding of Android application development and its challenges. This android application can be used to build Structural Designs for any of the five project types (SINGLE CELL [3], TWINCELL [4], CCHAN [5], CBASIN [6], and DRPWS3E [7]). The simulation results obtained from this application will be useful to design engineers or anyone interested in developing structural designs.

5.2 Future Work

The application can be extended in an extensive basis in the following ways

- Currently, there is no API for Android which can be capable of showing the PDF files directly from the app's asset folder. During the installation of the app, files from the assets folder will be copied to a folder on external storage and the app will show PDF files from that location. Lack of sufficient storage space on external storage will be a drawback with this scenario. A more sophisticated way of showing the PDF file can be developed by implementing a library.
- The same application can be extended to the ios platform by developing an application for the ios tablets.

Chapter 6 - References

[1] USDA Soil Conservation Service

http://www.nrcs.usda.gov/wps/portal/nrcs/site/national/home/

[2] Design Procedures for planning and designing Earth Dams and Reservoirs (TR – 60)

ftp://ftp.wcc.nrcs.usda.gov/wntsc/H&H/TRsTPs/TR60.pdf

[3] TR-210-42 - Single Cell Rectangular Conduits Criteria and Procedures for Structural Design (12/1969)

http://directives.sc.egov.usda.gov/viewerFS.aspx?id=3840

[4] TR-210-45 - Twin Cell Rectangular Conduits-Criteria and Procedures for Structural Design (9/1970)

http://directives.sc.egov.usda.gov/viewerFS.aspx?id=3840

[5] TR-210-50 - Design of Rectangular Structural Channels (Revision 1) (7/1977)

http://directives.sc.egov.usda.gov/viewerFS.aspx?id=3840

[6] TR-210-54 - Structural Design of SAF Stilling Basins (10/1974)

http://directives.sc.egov.usda.gov/viewerFS.aspx?id=3840

[7] TR-210-54-1 - Structural Design of SAF Stilling Basins, revised wingwall design, Amendment 1 (9/1981)

http://directives.sc.egov.usda.gov/viewerFS.aspx?id=3840

[8] TR-210-63 - Structural Design of Monolithic Straight Drop Spillways (2/1977)

http://directives.sc.egov.usda.gov/viewerFS.aspx?id=3840

[9] F2J Translator

http://icl.cs.utk.edu/f2j/

[10] Structural Design Software (Fortran Executable) and Formatters used for F2J translation

http://sites.cis.ksu.edu/tr42/

[11] Android Developer Guide

http://developer.android.com/index.html

[12] Android Coding

http://stackoverflow.com/

[13] FORTRAN 77 Tutorial

http://www.fortran.com/F77_std/rjcnf0001.html

[14] Android Development Tutorial

http://www.vogella.com/android.html

[15] Android PDF Writer

http://coderesearchlabs.com/androidpdfwriter/

[16] USDA Natural Resources Conservation Service Logo

http://www.nrcs.usda.gov/wps/portal/nrcs/site/national/home/