Test Plan Document

For

Chore Tracking Android Application

Team: Jonathan Samuelsen, Troy Laskiewicz, Sean Bermingham, Ryan Ford
Client: Dr. Karen Anewalt
Date: March 23, 2018

Table of Contents

1. Introduction

1.1. Purpose

The purpose of the test plan document is to provide a description of the client's requested software, a list the requirements that must be satisfied in order to implement the system, and lay out a plan and schedule to test the system and ensure that all requirements are met satisfactorily.

1.2. Scope

The system is an Android application that allows parents to easily communicate various household chores to their children in an incentivized manner. The application allows parents to create and post various chores for their children. The children sign off on each chore after completing them and receive some amount of reward points that were allocated during the chore's creation. Parents also have the option of creating and posting rewards that cost a set number of rewards points and are redeemable by the children. The client is Dr. Karen Anewalt, a professor of computer science at the University of Mary Washington. Dr. Anewalt and her children are the end users of this software.

1.3. Overview of the Remainder of Document

The remainder of the document details the system requirements, test plans, and test procedures into the following sections:

1.3.1. Project Description

Contains in-depth information about the expected system functionality, specific system requirements, client characteristics, and user characteristics.

1.3.2. Test Plan

Describes the overall testing strategy that is required to properly execute effective testing, including the necessary resources, products, record keeping strategies, and schedule to carry out the testing phase as efficiently as possible.

1.3.3. Test Procedure

Provides a detailed test procedure including test tactics and test cases for each type of test that is required to ensure flawless functionality within the system.

1.3.4. Appendix

Contains a glossary of any technical terms that the reader may not understand as well as a list of authors with their respective contributions.

2. Project Description

2.1. System Overview

The system will be an Android phone application that allows a parent to have several children accounts linked to theirs. Once linked, the parents will be able to create chores, delete chores, and create rewards for their children to view and complete. Children will have access to viewing chores from their parent linked account and completing them for stars. Once enough stars have been gained, they will be able to redeem a prize from a list their parent has generated. A user without an account will be shown the main menu and will have the ability to create an account and will be prompted for a username, email, password, and role. Once the account is created, they will be signed in. Signed in users will be able to navigate to their appropriate view page. On the view pages, they will be able to do role-specific tasks. To log out, the user must return to the main menu where there will be a button that will give them the ability to log out.

2.2. Client Characteristics

Our client is Dr. Karen Anewalt. She is a mother and is pursuing this app in order to have an easy way to motivate and keep track of her children’s chores. By using this app it will make chores more fun to do because there are listed rewards her children can strive for. This will encourage her children to do chores more frequently.

2.3. User Characteristics

The end users will be Dr. Karen Anewalt and her children. Dr. Anewalt will be in a parent role while her children will be in a child role. Although created for her family, it will also be available to anyone else's family.

2.4. Functional Requirements

2.4.1. The ability for children to mark chores as done and claim rewards

2.4.2. The ability for parents to create and delete chores

2.4.3. The ability for parents to create and delete rewards

2.4.4. The ability to create an account

2.4.5. The ability to select a role when creating an account

2.4.6. The ability to login to the system

2.5. General Constraints

2.5.1. The system will be hosted on Amazon Web Services

2.5.2. The application will be coded using Java
2.5.3. The database will be hosted in the cloud via Firebase
2.5.4. The users and chores will be stored in the database

3. Test Plan

3.1. Testing Strategy

The core functionality of the system can be broken down into three categories: accounts, chores, and rewards. The functionality and requirements pertaining to these categories all operate independently of each other, allowing for in-depth testing of each one regardless of whether or not the other categories function properly. To test accounts, users must be able to simply create an account and login to the system. To test chores, parent accounts must be able to create and delete chores, while child accounts must be able to mark them as completed. To test rewards, parent accounts must be able to create and delete rewards, while child accounts must be able to use their points to claim them.

3.2. Testing Resources and Staffing

In order to test the system, the testing team will need an IDE that allows for Android OS simulation. The recommended software is Android Studio, which can be downloaded and installed through the following link:

https://developer.android.com/studio/index.html

After following the installation and setup instructions provided by the software, the testing team can access the system through the following repository link:

https://github.com/tlaskiew/choreTracker

To clone the repository in Android Studio, follow these steps:

1. In the toolbar, select File > New > Project From Version Control > GitHub
2. Copy the repository link into the Git Repository URL field
3. Click “Clone”

After you have cloned the repository, the application can be run by selecting Run > Run ‘app’ from the toolbar or clicking on the green triangle at the top of your screen.

The integration team will be responsible for setting up a functional work environment for the testing team. The testing team will be responsible for starting the application and evaluating its functionality.

3.3. Test Work Products

Testing of the three independent functionalities: accounts, chores, and rewards will produce these results:

- Account:
    - All users can:
        - Create an account with an email and password
        - Log out of the account
        - Log in to the account
- Chores:
    - Parent users can:
        - Create chores
        - Delete chores
    - Child users can:
        - View chores created by Parent account
        - Mark chores as completed
- Rewards:
    - Parent users can:
        - Create rewards
        - Delete rewards
    - Child users can:
        - Claim rewards using earned points

3.4. Test Record Keeping

Below is a form to be completed by the testing team to check the features listed above.

Test Category	**User type**	**Tested action**	**Bugs:**	**Notes:**
Account	*All*	Create Account		
Account	*All*	Log out		
Account	*All*	Log in after logging out		

Chores	*Parent*	Create chore		
Chores	*Parent*	Delete chore		
Chores	*Child*	View chores		
Chores	*Child*	Mark as complete		
Rewards	*Parent*	Create rewards		
Rewards	*Parent*	Delete rewards		
Rewards	*Child*	Claim rewards		

3.5. Test Schedule

Subsystem	Start Test Date	End Test Date
Account	3/30/2018	4/13/2018
Chores	4/13/2018	4/27/2018
Rewards	4/20/2018	4/27/2018

4. Test Procedure

4.1. Create an Account

The tester will attempt to create an account. This test is connected to functional requirement 2.4.4. The tester will attempt to successfully create a separate parent and child account with all of the required information. The system will also be tested by attempting to create an account without a username entered and then without a password entered.

Test #	**Related to Requirement**	**Subsystem**	**Purpose**	**Test Case Data**	**Expected Results**
1	2.4.4	Accounts	Test a successful parent account creation	Email: t@me.com Username: testParent Password:	The account will be created and the user will be

				password Role: Parent	logged into the application
2	2.4.4	Accounts	Test a successful child account creation	Email: t@me.com Username: testChild Password: password Role: Child	The account will be created and the user will be logged into the application
3	2.4.4	Accounts	Test an unsuccessful account creation	Email: leave blank Username: testFail Password: password Role: Parent	An error message will be displayed asking for all required information
4	2.4.4	Accounts	Test an unsuccessful account creation	Email: t@me.com Username: leave blank Password: password Role: Parent	An error message will be displayed asking for all required information
5	2.4.4	Accounts	Test an unsuccessful account creation	Email: t@me.com Username: testFail Password: leave blank Role: Parent	An error message will be displayed asking for all required information

4.2. Login

The tester will attempt to login to an account after first opening the application. A valid username and password will be provided for the tester. If successful, the user should be redirected to the welcome screen of the application, where they can see the list of posted chores. The tester will also enter various invalid login credentials, which will result in an error message from the application. This test is connected to functional requirement 2.4.6.

Test #	**Related to Requirement**	**Subsystem**	**Purpose**	**Test Case Data**	**Expected Results**
1	2.4.6	Login	Test a successful login attempt	Username: test Password: abc123	Successful redirection to the chore list
2	2.4.6	Login	Test an unsuccessful login attempt	Username: test Password: wrongpass	An error message for invalid login
3	2.4.6	Login	Test an unsuccessful login attempt	Username: wrongname Password: abc123	An error message for invalid login

4.3. Create and Delete a Chore

The tester will attempt to create and delete chores in the application. This test is directly related to functional requirement 2.4.2. The tester will create a chore, view it in the database to ensure successful creation, and then delete the chore they just created. The tester will also attempt to create a chore without filling out all necessary information and receive an error message from the application.

Test #	**Related to Requirement**	**Subsystem**	**Purpose**	**Test Case Data**	**Expected Results**
1	2.4.2	Chores	Test a successful chore creation	Name: Mop Kitchen Star Value: 5	The chore will appear in the current chores list.
2	2.4.2	Chores	Test a	Chore	The chore

			successful chore deletion	Name: Delete Me	will be removed from the current chores list.
3	2.4.2	Chores	Test an unsuccessful chore completion	Name: Leave Blank Star Value: 1	The app will show an error message.

4.4. Mark a Chore as Completed

The tester will attempt to mark a chore as completed. This test is related to functional requirement 2.4.1. Since there is no data validation involved with marking chores as completed, the tester will only have one unit test to complete for this feature. If it is successful, the chore will disappear from the list and the user's rewards points will be incremented.

Test #	**Related to Requirement**	**Subsystem**	**Purpose**	**Test Case Data**	**Expected Results**
1	2.4.1	Chores	Test a successful chore completion.	Chore Name: Redeem Me	The chore will be removed from the list and the user's rewards points will be incremented

4.5. Create and Delete a Reward

The tester will attempt to create a reward as a parent role and then delete it. This test is related to functional requirement 2.4.1. The tester will create a reward and make sure that it was added to the database. Then they will delete the reward and make sure that it's no longer in the database.

Test #	Related to Requirement	Subsystem	Purpose	Test Case Data	Expected Results
1	2.4.1	Reward	To test creating a reward	Reward: $10 Star Value: 25 stars	The reward will be added to the current rewards list
2	2.4.1	Reward	To testing deleting a reward	Reward: $10 (25)	The reward will be removed from the current rewards list
3	2.4.1	Reward	To test invalid input for creating a reward	Reward: Leave blank Star Value: 10	The app will prompt the user to input a reward name

4.6. Claim a Reward

The tester will attempt to redeem a reward using their rewards points. They will be provided with login information for an account that already has rewards points accumulated. They will first attempt to redeem a reward that they have sufficient points for, and then attempt to redeem a reward that they cannot afford with their current number of points.

Test #	Related to Requirement	Subsyste m	Purpose	Test Case Data	Expected Results
1	2.4.1	Rewards	To test turning in a reward	Reward: Extra hour of screen time Star Cost: 50 Stars available: 50	The system will inform the user that their reward has been redeemed and notify the connected parent user

2	2.4.1	Rewards	To test turning in a reward with insufficient points	Reward: Extra hour of screen time Star Cost: 50 Stars available: 25	The system should inform the user that they don't have enough stars for that reward

5. Appendices

5.1. Glossary of Terms

5.1.1. IDE - Integrated development environment; a software application that provides comprehensive facilities to computer programmers for software development

5.1.2. Repository - a central place where data and/or files are stored allowing for distribution over a network

5.2. Author Information

5.2.1. Jonathan Samuelsen

Table of Contents
1.1 Purpose
1.2 Scope
1.3 Overview of the Remainder of the Document
2.5 General Constraints
3.1 Testing Strategy
3.2 Testing Resources and Staffing
4.1 Create an Account
4.2 Login
4.3 Create and Delete a Chore
4.4 Mark a Chore as Completed
5.1 Glossary of Terms
5.2 Author Information

5.2.2. Troy Laskiewicz

2.1 System Overview
2.2 Client Characteristics
2.3 User Characteristics
2.4 Functional Requirements
4.5 Create and Delete a Reward

5.2.3. Ryan Ford

3.3 Test Work Products
3.4 Test Record Keeping
3.5 Test Schedule
Proofreading, Grammar edits

5.2.4. Sean Bermingham

3.5 Test Schedule
4.6 Claim a Reward