

# Design Document

## Inventory App

Student: Keith Mullins  
Supervisor: Nigel Whyte  
Student ID: C00118202

## Contents

Introduction .....	3
Scope .....	4
Overview .....	4
Use Case Diagram.....	5
Class Diagram .....	6
Use Cases .....	7
Register.....	7
Login .....	7
Add Item.....	8
Export.....	8
Search Item .....	9
List Inventory.....	9
Scan Barcode.....	10
Scan Barcode.....	10
System Sequence Diagrams.....	11
Add / Edit Item.....	11
Scan Barcode / Scan ISBN.....	12
Database Design .....	13
User Interface .....	14
Main Screen      Main Menu.....	14
Inventory List      Add / Edit Item .....	15
Barcode Scanner.....	16
.....	16
ISBN Scanner.....	16
Category List      Export .....	17
References .....	18

## Introduction

The purpose of the design document is to demonstrate how the Inventory App will work. This document is a continuation of the functional specification, which detailed what the app should do, so this document looks at the functionality provided in the functional specification and describes in detail, how it should be implemented. This document should allow any developer to develop and build the app based and the information provided.

The design document includes the following:

- Use case diagram
- Use cases
- Class diagram
- System sequence diagrams

## Scope

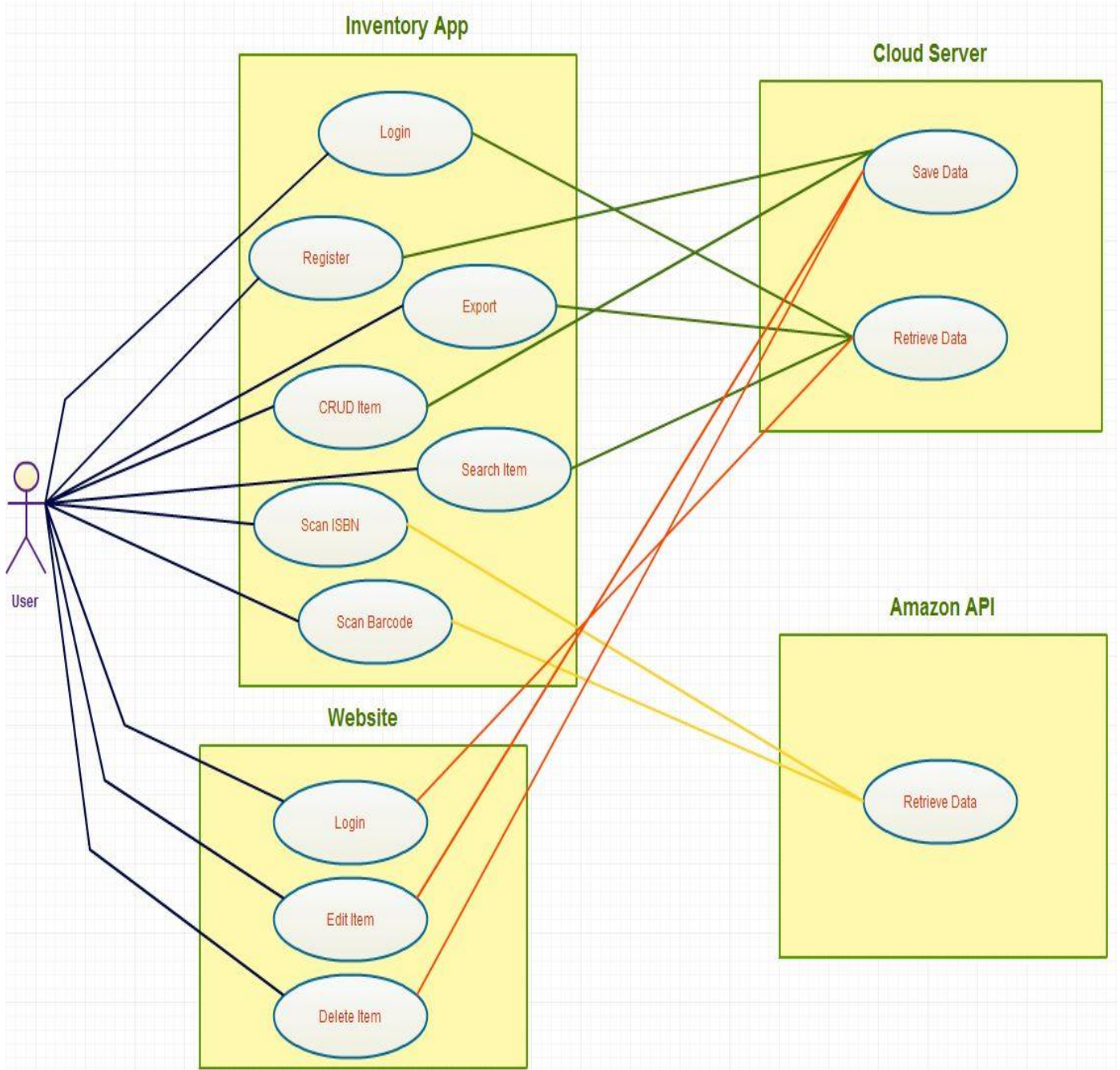
This document contains complete description of the design of Inventory Application. The document describes each component of the application. This document is intended for the developers of the application. It defines architecture, use cases, data design and user interface design. The application will be developed according to this document; however minor changes may occur during the development process.

## Overview

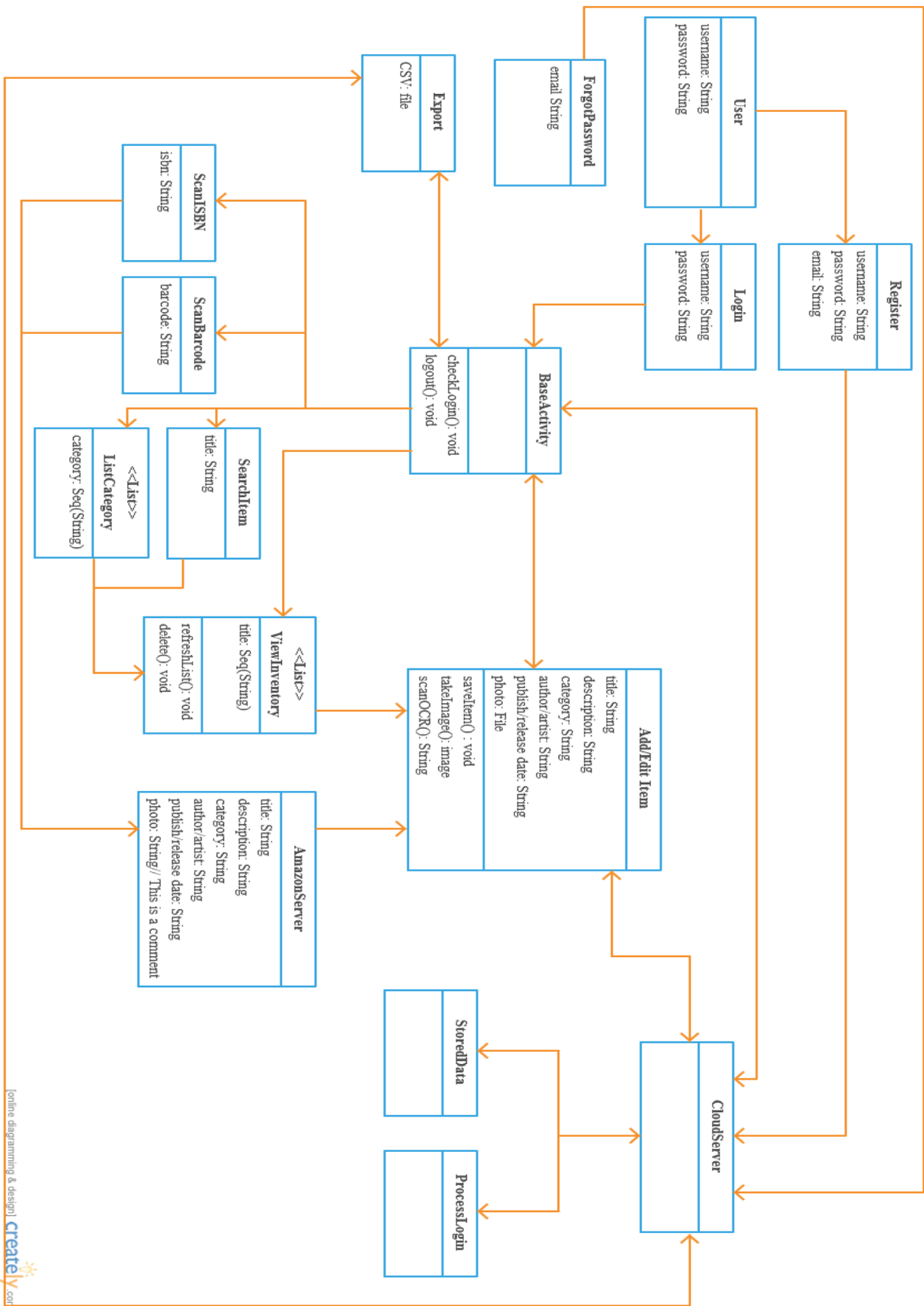
This project aims to develop a mobile application which will allow users to keep a record of items that they own. The application will allow users to add items to the system for different categories. The application will make it easier for people to keep track of their collections.

# Architecture Design

## Use Case Diagram



# Class Diagram



# Use Cases

## Register

Actors: User

Course of events:

1. This use case begins when the user wishes to create an account in order to gain access to the Inventory App. During the account creation process, the user enters the required details, i.e. their username, their contact e-mail address and password.
2. Once they have entered the details they then tap sign up to complete the account creation.
3. The account is now created and the user can login.

Alternatives:

- 1a. If the username or email has already been used as error message will be displayed to the user confirming that fact.

## Login

Actors: User

Course of events:

1. This use case begins when a user wishes to gain access to the app. They enter their username and password and tap login to proceed to the main app screen.
2. If the alternative method of logging in via social networks is developed, then the user can choose to login using the details of the chosen network.

Alternatives:

- 1a. The username and/or password are incorrect. An error is displayed to the user letting them know and they must re-enter one or both.
- 1b. The app could not retrieve account data due to bad connection. Display an error.

## **Add Item**

Actors: User

Course of events:

1. This use case begins when the user wishes to add a new item. The user can either scan an item to get the item details or create a new item by clicking the “+” icon on the actionbar on the inventory list activity.
2. The user can enter the details or edit the details if the item details were retrieved automatically.
3. An image of the item can be captured and saved.
4. When the details of the item are entered the user will press the save button to save the item to the cloud.

Alternatives:

- 4a. The app could not send the list data to the cloud. Display an error.
- 4a. The app could not retrieve list data from the cloud. Display an error.

## **Export**

Actors: User

Course of events:

1. This use case begins when the user wishes to export their collection to an external file to view on their computer.
2. When the user click on the export button a file is CSV generated which can then be attached to an email and sent.

Alternatives:

- 2a. The app could not get the list of data from the cloud. Display an error.



## **Search Item**

Actors: User

Course of events:

1. This use case begins when the user wishes to search for an item in their inventory.
2. The user can enter the title of an item to search for and the item will be displayed
3. A partial search can also be performed such as entering a substring of the title.

Alternatives:

- 2a. The app could not get the list of data from the cloud. Display an error.

## **List Inventory**

Actors: User

Course of events:

1. This use case begins when the user wishes to view their inventory
2. When the user clicks the inventory button the app will display the list of all the items already added
3. When an item is clicked on the add / edit item screen, it is displayed with the item details.

Alternatives:

- 2a. The app could not get the list of data from the cloud. Display an error.

## Scan Barcode

Actors: User

Course of events:

1. This use case begins when the user wishes to add an item to the app by scanning the barcode
2. When the user clicks the scan barcode button the app will display the barcode scanning activity.
3. The user will hold the scanner over the barcode of the item and the scanner will capture the barcode automatically
4. The item details will automatically be retrieved from the Amazon server

Alternatives:

- 4a. The app could not get the list of data from the server. Display an empty add / edit activity and inform the user.

## Scan Barcode

Actors: User

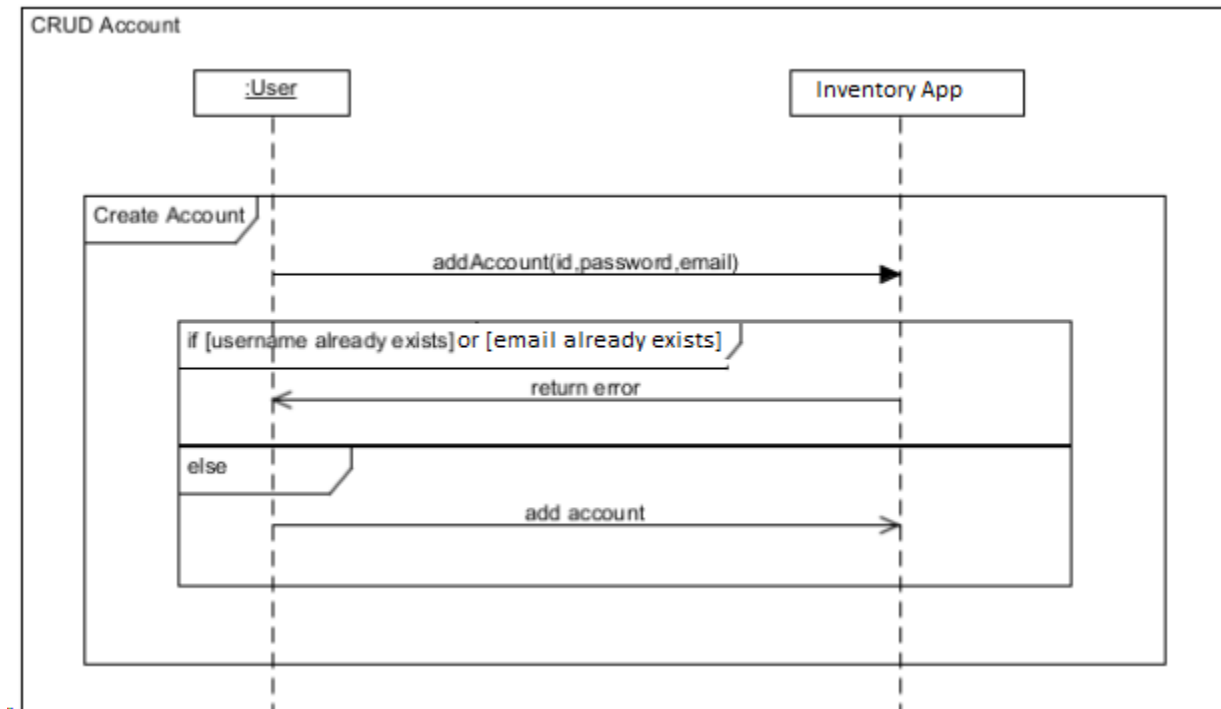
Course of events:

1. This use case begins when the user wishes to add a book to the app by scanning the ISBN
2. When the user clicks the scan isbn button the app will display the capture ocr scanning activity.
3. The user will hold the scanner over the isbn of the item and click the capture button
4. The item details will automatically be retrieved from the Amazon server

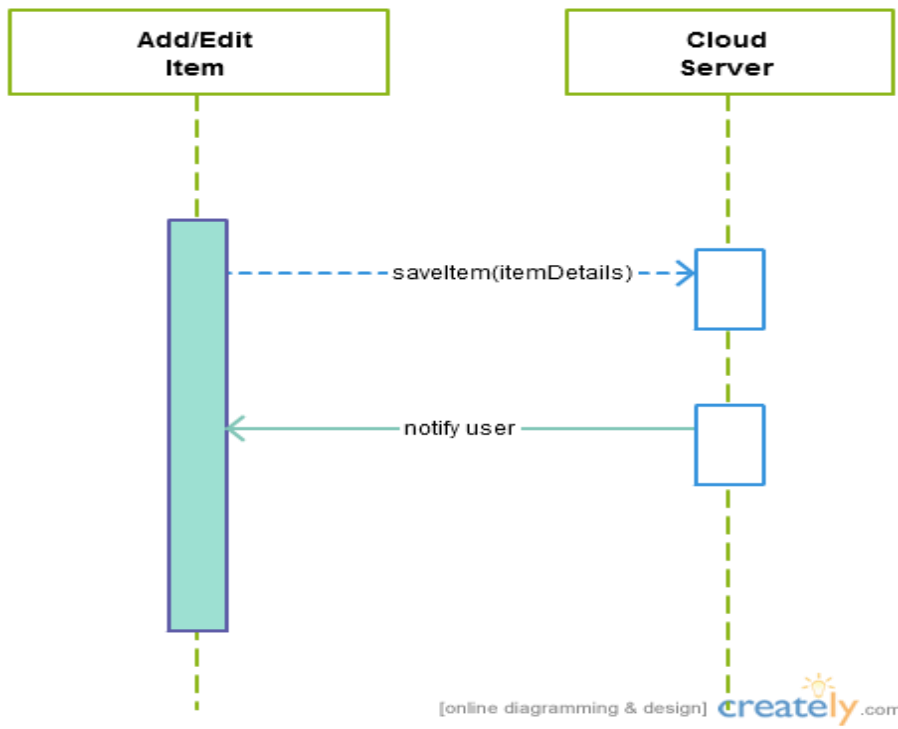
Alternatives:

- 3a. The OCR could not capture the text. Display error.
- 4a. The app could not get the list of data from the server. Display empty add / edit activity and inform the user.

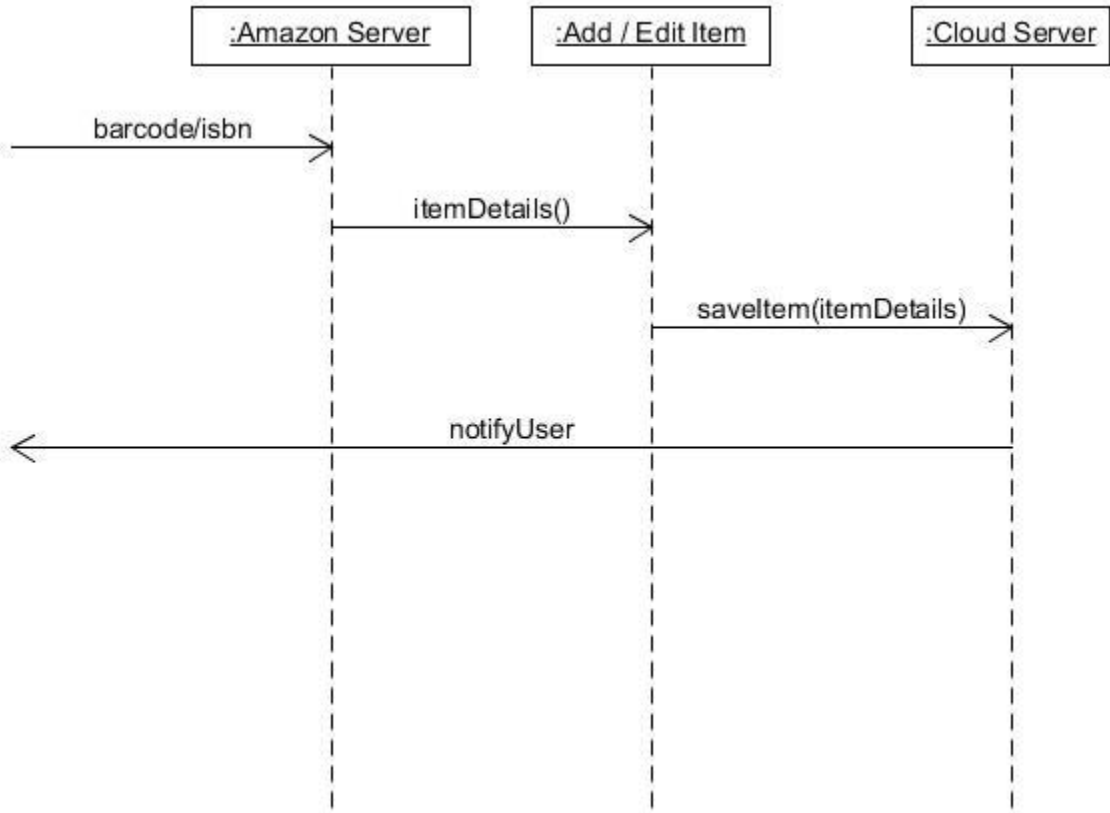
# System Sequence Diagrams



## Add / Edit Item



Scan Barcode / Scan ISBN



## Database Design

The Inventory Application will store all of the data it needs in the cloud using a backend called Parse. This will include everything from user account data to item details.

The Parse backend stores data using ParseObjects which contain “contains key-value pairs of JSON-compatible data”. [1]

This data is schemaless, which means that you don't need to specify ahead of time what keys exist on each ParseObject. You simply set a key-value pair and the backend will store it.

You can store basic data types, including locations and photos, and query across them without spinning up a single server. [2]

Parse works off of requests/sec with the free option offering 30req/s which should be sufficient for 10,000 concurrent users.

### User Account

Property	Data Type	Comment
username	string	Allows the user to login
email	string	Allows user to recover password
password	string	Allows user to login

### Inventory

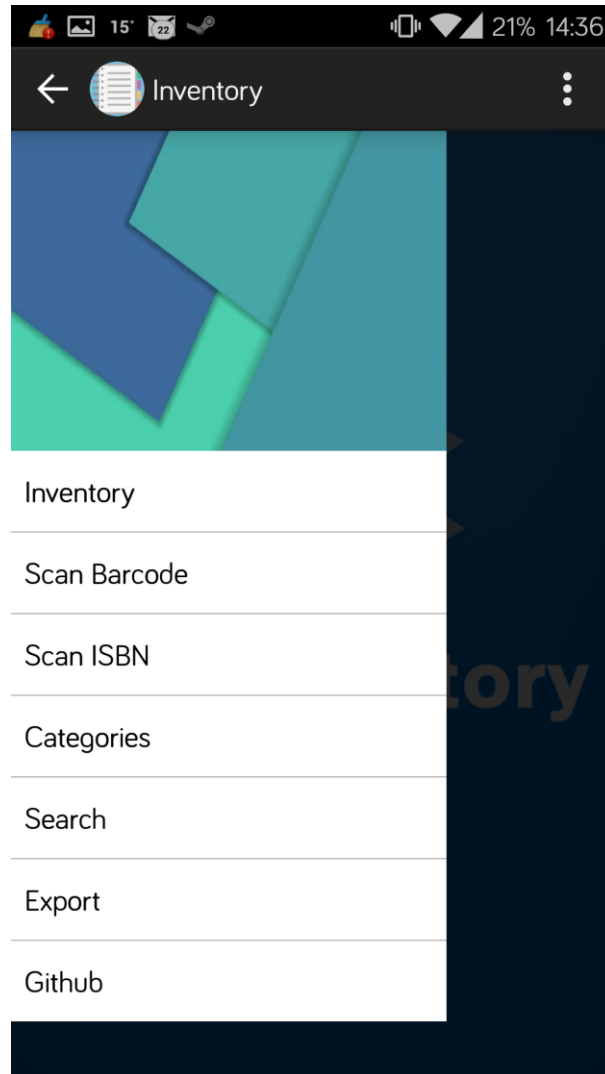
Property	Data Type	Comment
author	Pointer<User>	the owner of the item
barcode	string	the item barcode
title	string	the item title
description	string	the item description
category	string	the item category
author/artist	string	the item author/artist if applicable
release date	string	the item release date

# User Interface

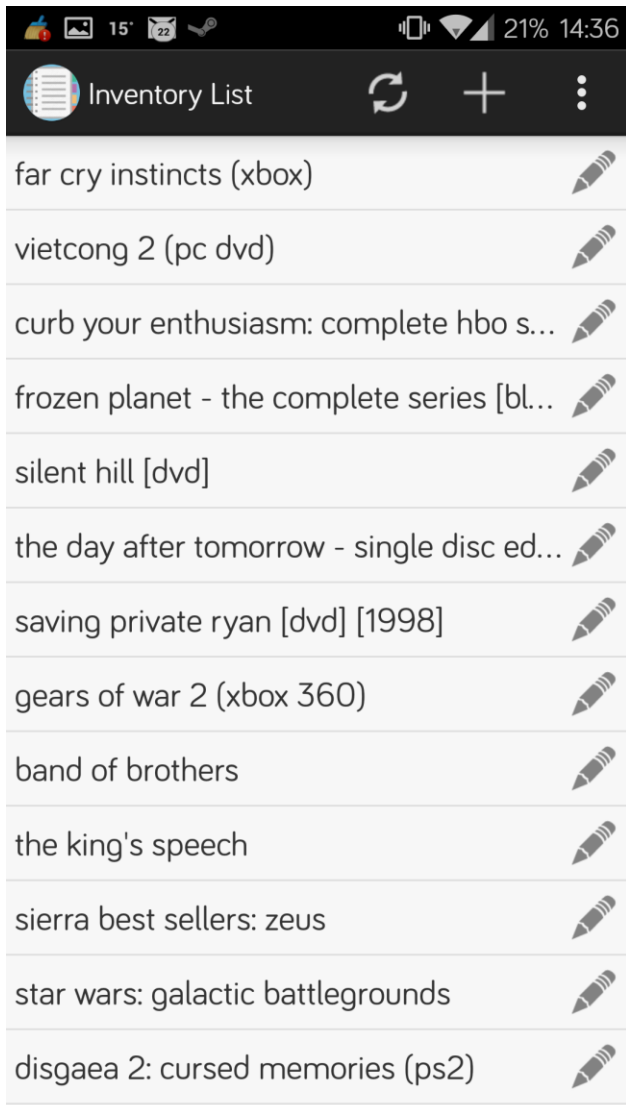
## Main Screen



## Main Menu



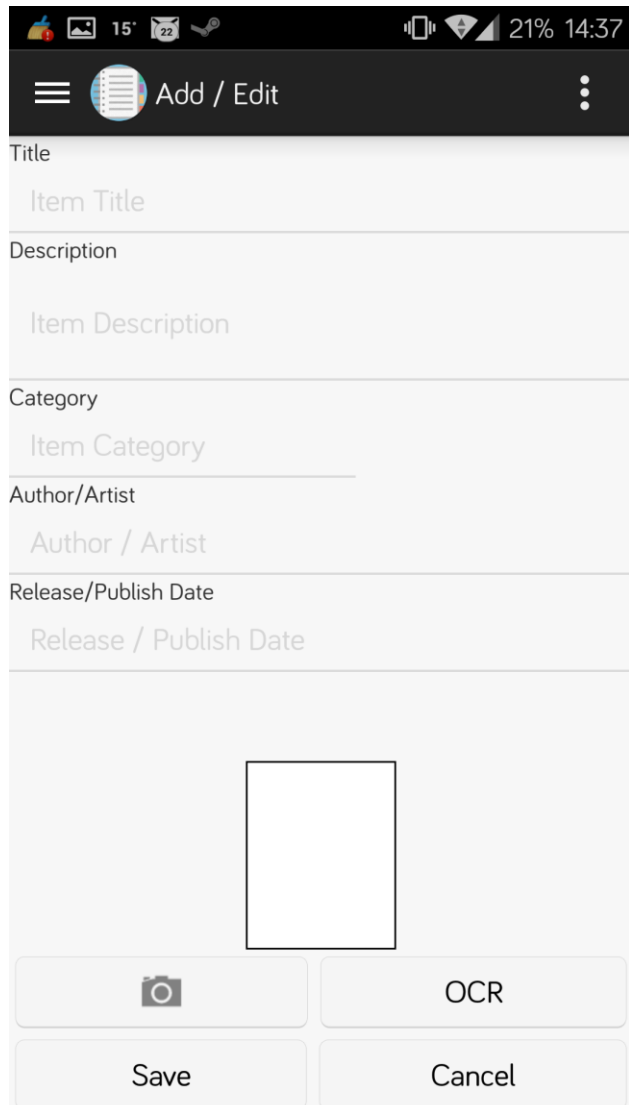
## Inventory List



The screenshot shows the 'Inventory List' app interface. At the top, there is a dark header with the app name 'Inventory List', a refresh icon, a plus sign for adding items, and a three-dot menu icon. Below the header is a list of items, each with a pencil icon for editing. The items listed are:

Item Name	Action
far cry instincts (xbox)	Edit
vietcong 2 (pc dvd)	Edit
curb your enthusiasm: complete hbo s...	Edit
frozen planet - the complete series [bl...	Edit
silent hill [dvd]	Edit
the day after tomorrow - single disc ed...	Edit
saving private ryan [dvd] [1998]	Edit
gears of war 2 (xbox 360)	Edit
band of brothers	Edit
the king's speech	Edit
sierra best sellers: zeus	Edit
star wars: galactic battlegrounds	Edit
disgaea 2: cursed memories (ps2)	Edit

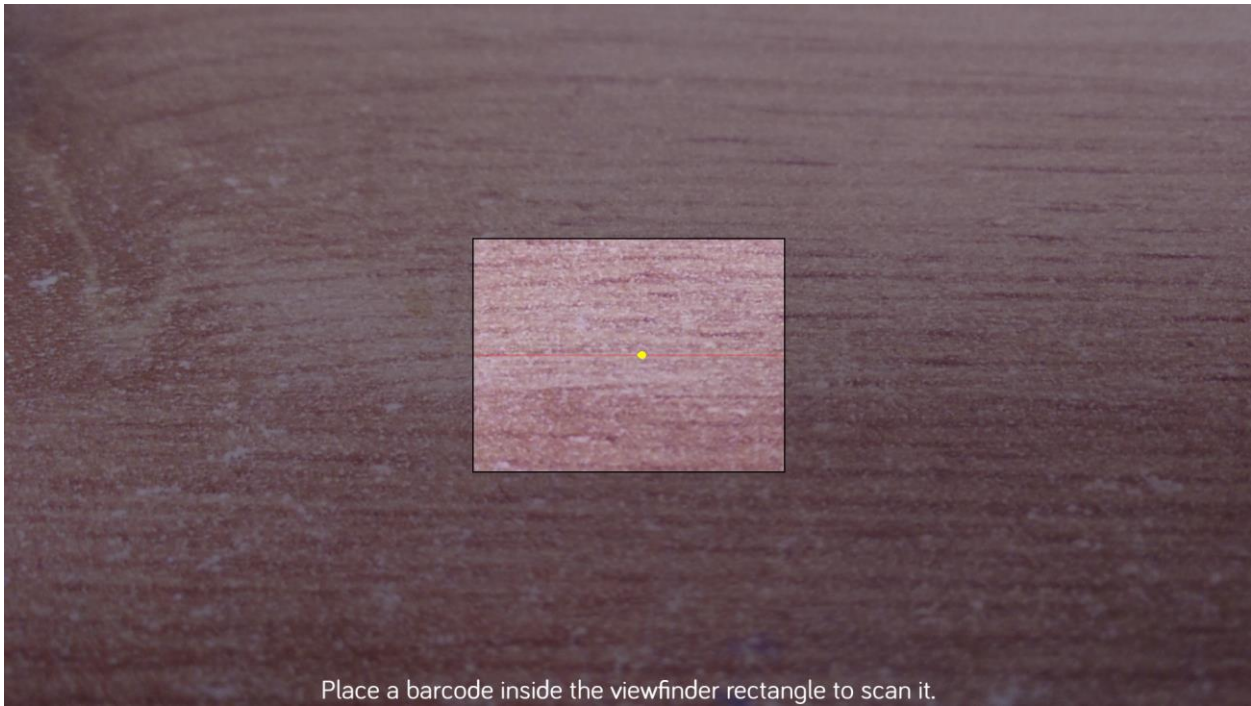
## Add / Edit Item



The screenshot shows the 'Add / Edit Item' app interface. At the top, there is a dark header with a hamburger menu icon, the app name 'Add / Edit', and a three-dot menu icon. Below the header is a form with the following fields:

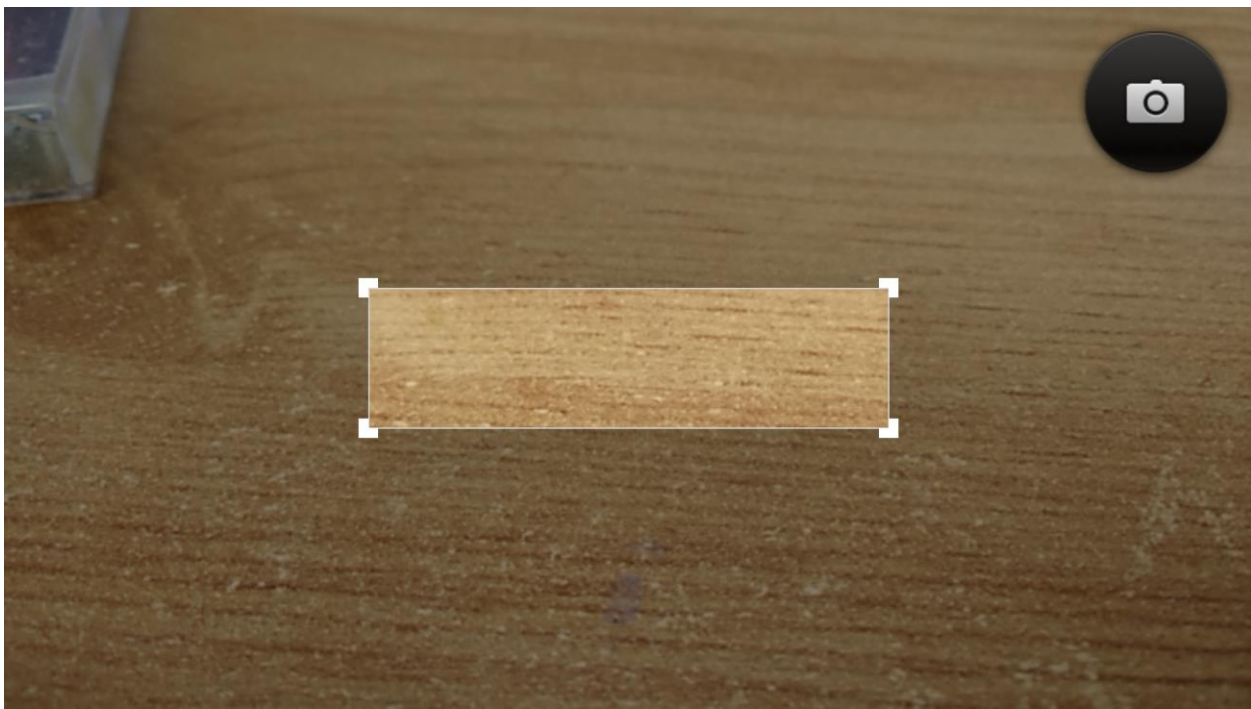
- Title: Item Title
- Description: Item Description
- Category: Item Category
- Author/Artist: Author / Artist
- Release/Publish Date: Release / Publish Date

Below the form is a large empty rectangular box for an image. At the bottom, there are four buttons: a camera icon for taking a photo, 'OCR', 'Save', and 'Cancel'.



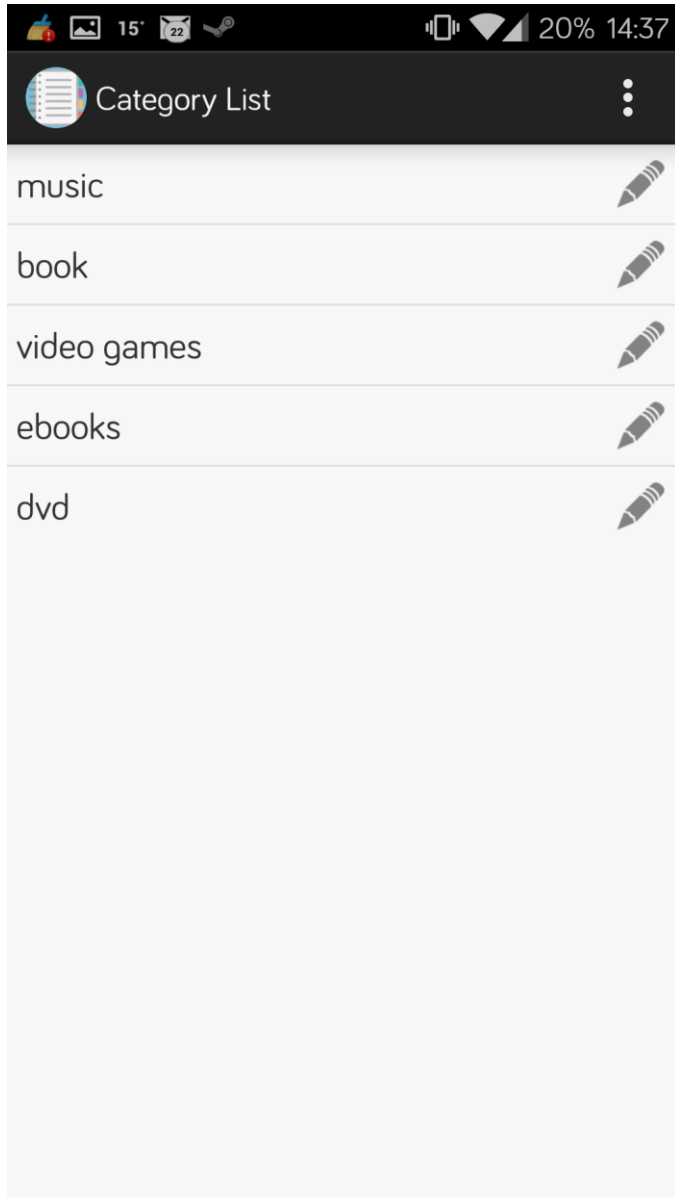
**Barcode Scanner**

**ISBN Scanner**

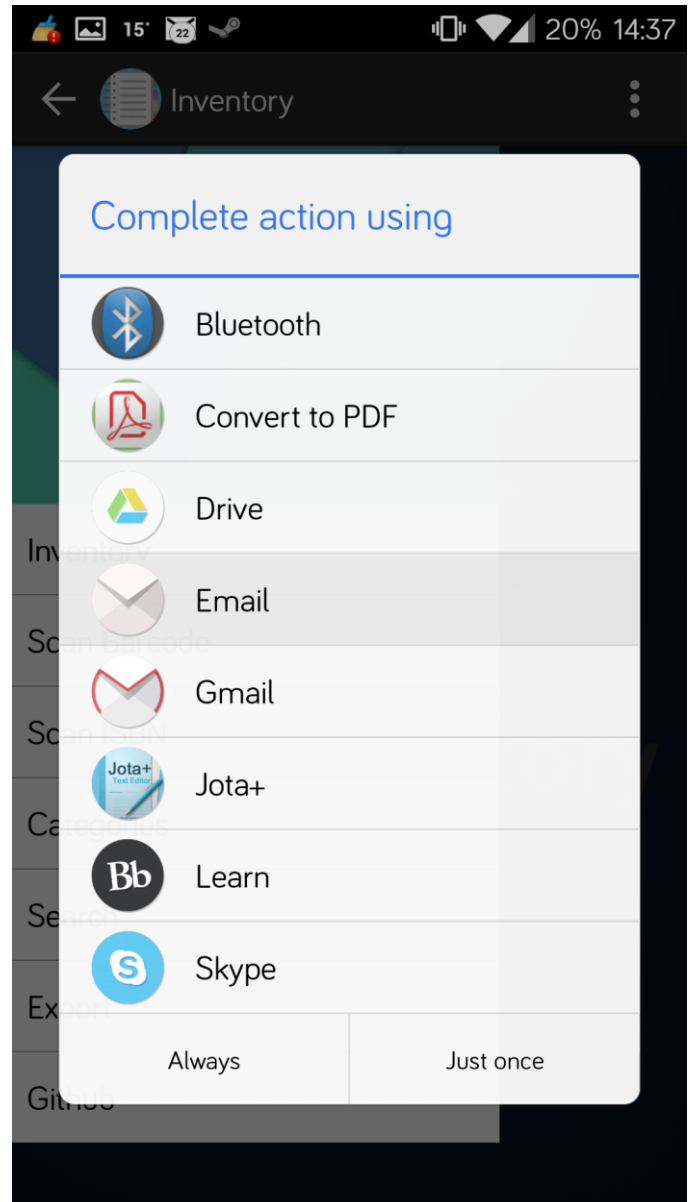




## Category List



## Export



## References

[1] Parse - <https://parse.com/products/core>

[2] Parse Documentation – [https://www.parse.com/docs/android\\_guide](https://www.parse.com/docs/android_guide)