CS683 Project Assignment   
Iteration 1: Project proposal  
CS683\_CulverGrant\_InventoryTracker  
Grant Culver

**Instructions**

* Please name your report as CS683\_<Last Name><First Name>\_<ProjectTitle>\_Iter0. It can be either a PDF or Word document.
* Please provide your feedback in the “Add comments” section when submitting your lab report. Thanks!

[**Overview**](#_g6igqliy7rm) **2**

[**Related Work**](#_bf21eadgjj29) **2**

[**Proposed High level Requirements**](#_rgyo4hi9stmq) **2**

[**Android Components and Features To Be Used**](#_ypm7oed533us)3

[**Timeline**](#_w7ryh1629f9n)4

[**References**](#_ts358bsdtbcv) **3**

# Overview

(Please give an overview of your project. It should include the motivation, the purpose and the potential users of the proposed application )

I work and own part of two automotive dealerships on 1 campus. At any given time, we have a couple hundred cars on the ground. At the end of the month, taking a physical inventory is difficult at times with cars in many different spots. Some are in service, some are sold that haven’t been finalized in accounting, some are on test drives, some are wholesaled, etc. Currently, we have out lot manager print off the accounting schedule for inventory and physically touch every car and mark it off the list with a pen. Creating an app that would allow him to do this by simply clicking a check box for each unit would be beneficial. Not only would our lot manager use this app but also our sales managers and accounting staff could use it as well.

# Related Work

(Please describe any similar applications that you have found through the online research, and the differences between your application and those applications.)

When I conducted my research, I did find several applications that were geared toward inventory management. However, none were standalone applications for an automotive dealership. The ones I did see, integrated with a CRM tool or were generic apps that would require considerable setup time.

# Proposed High level Requirements

* 1. Essential Features:   
     (Please describe all essential features that your application will have. For each feature, please give a feature title, a brief description, possible acceptance tests, as well as one or multiple mock-up screen(s) if apply.)
     1. Database driven inventory management
        1. Description: Inventory will be stored and accessed via a mongo DB.
        2. Acceptance test: If inventory is retrieved from mongo DB.
        3. Acceptance test: If inventory is updated based on user input when conducting inventory.
     2. Segregation of new and used inventory.
        1. Will have separate tabs for new inventory and used car inventory.
     3. Inventory status update
        1. Lot manager will have ability to mark the status of inventory as on Lot, sold, test-drive, status, IDK.
        2. Data will update in database to reflect change.
        3. Acceptance Test: Lot manager can appropriately select status of unit location.
        4. Acceptance Test: Inventory is updated in database.
     4. Finalize Report
        1. When inventory is done app will email status update to sales managers and office controller.
        2. Will list the units by status.
        3. Acceptance test: do the email send and deliver to recipients with correct format.
  2. Desirable Features

(Please describe all desirable features that your application would like to have. For each feature, please give a feature title, a brief description, possible acceptance tests, as well as one or multiple mock-up screen(s) if apply.)

* 1. Optional Features  
     (Please describe all optional features that your application would like to have if the time permits. For each feature, please give a feature title and a brief description.)
     1. Authentication
        1. Users can authenticate to ensure they have authorization.
     2. Integrated GPS tracking
        1. Each unit will have GPS module installed to track location.
     3. KeyTrack Integration
        1. API created to integrate with out key vault to inform lot manager if keys are accounted for. If not, will let manager know who last checked keys out.
     4. Camera VIN Scanner
        1. Ability to scan VIN number to facilitate quicker accounting of inventory.

# Android Components and Features To Be Used

(Please specify basic android components and features your application plans to have. Here is a list of components/features you can use: activities, databases and/or content providers, files, settings (through sharedpreferences), services, broadcast receivers, multi-threaded, multi-process, animation, graphics, remote server connection, GPS, microphone, camera, or other sensors, google APIs, etc. For each android component/feature, briefly describe how this can be fit into your application).

1. Database: Connection to database from application.
2. Services: To make calls to database and do heavy lifting of logic.
3. Activities: Application will make function/method calls that will need to be process by activities.
4. TextView
   1. To display menu text and inventory text.
5. Buttons
   1. To select Inventory or menu items.
6. Form Groups
   1. To stack the views in an organized and cohesize manner.
7. Radio Buttons
   1. To set inventory status
8. Intents
   1. Cross component communication

# Timeline

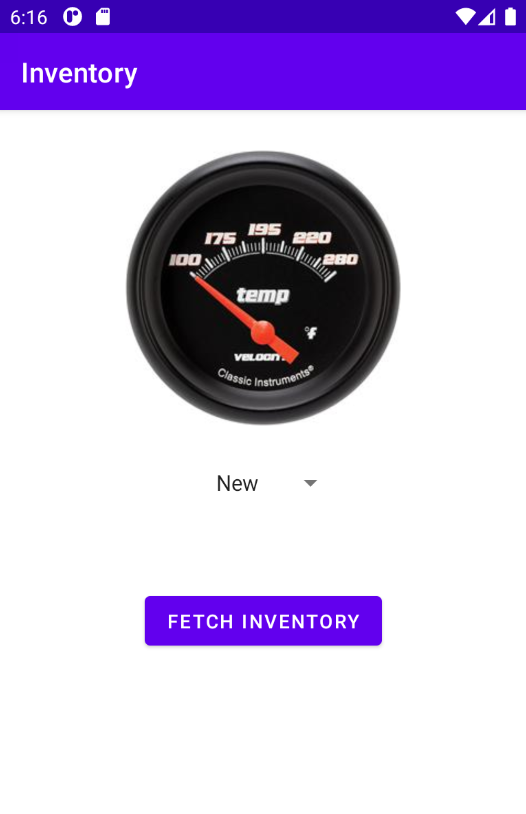
(Please provide a detailed plan to specify when the above requirements and android features will be implemented)

|  |  |  |
| --- | --- | --- |
| Iteration | Application Requirements(E/D/O) | Android Components and Features |
| 1 | Basic Layout | Screen layout for segregation of inventory: new or used. Then page showing all inventory in stock. First iteration will contain dummy data. |
| 2 | Extended Inventory Accounting Features | Being able to click on the inventory and set the stock status. |
| 3 | Retrieve inventory from database | Populate display pages will all in stock inventory and have the ability to click on each unit and set stock status. |
| 4 | Update inventory when marked complete | Send inventory status update to database when completed. |
| 5 | Email stakeholders | Send report to stakeholders detailing and listing status of inventory. |

# Design and Implementation

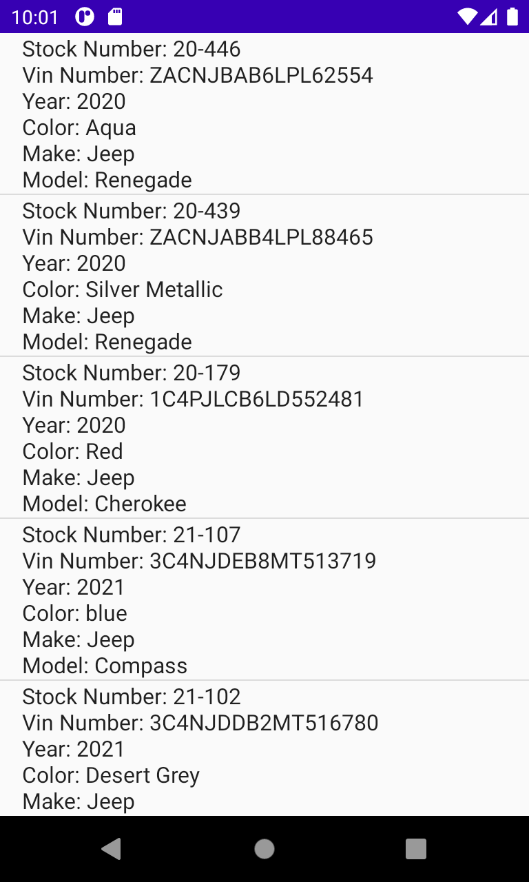
For iteration 1 I have worked on the basic layout of the application.

For the main activity, I added an application logo using an ImageView component.



The image view gives the user ability to click either new or used inventory. When the user is finished selected and clicks the “Fetch Inventory” button the inventory is displayed in another activity. This was tested on Nexus 4 device.

The activity that displays the data currently is a LinearLayout that is utilizing a ListView to iterate through the inventory that is in-stock.



Both of these are exisiting requirements that I specified in the first iteration.

# Design and Implementation

From the first iteration, I have added 2 activities and 2 corresponding Java files. In addition, I added a New\_Inventory class that has “dummy data” along with my constructor and getters and setters.

# 

# References