Artifact 2 Enhancements: Data Structures and Algorithms CS 360 Inventory App - Manager's Dashboard Fragment

The artifact is a comprehensive dashboard for managers to oversee inventory in a management system. It has multiple sections:

1. Top Stock Items - Shows the most popular inventory items.

2. Recent Entries - Displays the latest inventory additions or updates.

3. Forecast Entries - Predicts future inventory needs based on past data trends.

The dashboard has a user-friendly swipe-able interface with different sections organized like tabs.

I included this dashboard in my portfolio because it demonstrates my skills in:

1. Designing complex user interfaces that are easy to navigate.

2. Working with data structures to efficiently store and access information.

3. Applying algorithms to analyze data and make predictions.

What Improvements Were Made?

Bar Chart for Top Stock Items:

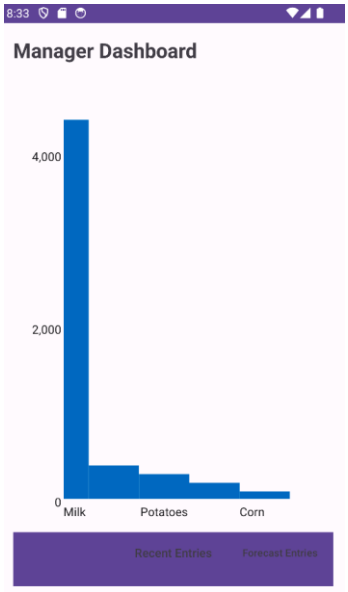
Data Visualization: I integrated a bar chart into the Top Stock Items Fragment to visually represent the inventory items with the highest stock levels. This visualization makes it easier for managers to quickly identify the top items in their inventory.

Advantages:

Quick Insights: The bar chart provides an at-a-glance view of the top items, enabling quick decision-making.

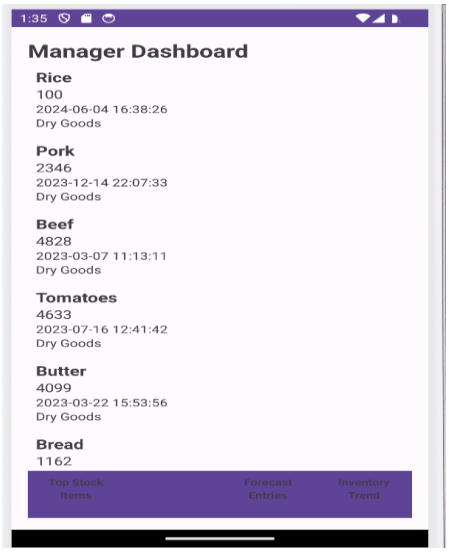
Interactive Elements: Users can interact with the chart to get detailed information about each item, such as stock levels and categories.

Clear Communication: Visual representation of data is often more effective than raw numbers, making it easier for managers to understand and act upon the information.



Recent Entries Clustering Algorithm:

K Means Clustering: To create a more affective reporting list. I implemented the K Means clustering algorithm to categorize inventory entries by the category that is most in demand. This algorithm groups items into clusters based on their similarities, helping in identifying patterns in inventory data.



The clustering process involves the following steps:

Data Preparation: Converting inventory attributes into numerical data points suitable for clustering.

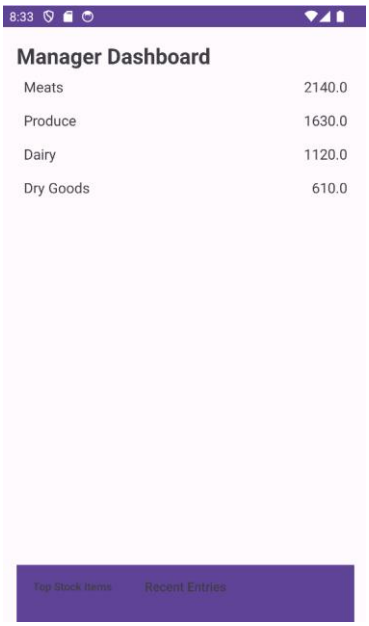
Cluster Initialization: Defining the number of clusters (k) and initializing centroids.

Iteration: Assigning data points to the nearest centroids and recalculating centroids based on the assigned points.

This clustering helps in categorizing inventory into distinct groups, making it easier to analyze stock patterns and manage inventory efficiently.

```java
public class CategoryClustering {
    static class CategoryWrapper implements Clusterable {
        private final int index;

        CategoryWrapper(int index) {
            this.index = index;
        }

        @Override
        public double[] getPoint() {
            return new double[]{index};
        }
    }

    // Method to perform K-means clustering
    public static List<CentroidCluster<CategoryWrapper>> performClustering(List<String> categories) {
        // Ensure there are exactly four categories
        if (categories.size() != 4) {
            throw new IllegalArgumentException("Exactly four categories are required for clustering.");
        }

        List<CategoryWrapper> points = new ArrayList<>();
        for (int i = 0; i < categories.size(); i++) {
            points.add(new CategoryWrapper(i));
        }

        // Specify the number of clusters as four
        int k = 4;
        KMeansPlusPlusClusterer<CategoryWrapper> clusterer = new KMeansPlusPlusClusterer<>(k);
        return clusterer.cluster(points);
    }

    // Helper method to get the category corresponding to the cluster center
    public static String getCategory(List<CategoryWrapper> points) {
        List<String> categories = new ArrayList<>();
        return points.get(0).index >= 0 && points.get(0).index < categories.size() ?
                categories.get(points.get(0).index) : "Unknown";
    }
```

Forecasting Model:



- A predictive model was implemented to forecast future inventory needs based on historical data.

- This involved collecting past inventory data, identifying relevant factors (like logged inventory trends), and training a model to make predictions.

- The forecasting helps managers plan for stock replenishments proactively, ensuring optimal inventory levels.

The enhancements to the Manager's Dashboard have improved its functionality and user experience, making it a strong addition to my portfolio. The use of advanced data structures and algorithms demonstrates my ability to solve complex real-world problems efficiently.

The enhancements to the dashboard align with the course objectives of implementing design solutions, programming algorithms and data structures, and addressing potential flaws in the design or structure of an application.

Enhancing this dashboard was a valuable learning experience that required a deep understanding of user interface design, lifecycle management, DSA and implementing machine learning algorithms for forecasting. Managing the different sections and ensuring smooth transitions was challenging. Ensuring the data analysis operations were performed correctly was also crucial.

Successfully implementing features like clustering algorithms and forecasting models demonstrates my ability to solve complex problems, implement robust solutions, and create innovative applications.