Product Requirements & Specification Document

Project Name

ClusterVision - Customer Segmentation Visualizer

Description

ClusterVision is an open-source, Streamlit-based tool for unsupervised customer segmentation. It enables users to perform K-Means clustering on retail customer data, visualize clusters interactively, and evaluate clustering quality using Elbow and Silhouette methods. Results are exportable for further analysis.

---

1. Goals & Objectives

Goal	Description
Customer Segmentation	Enable non-technical users to segment customers using K-Means
Interactive Visualization	Provide intuitive, interactive cluster visualizations
Clustering Evaluation	Support Elbow and Silhouette methods for cluster evaluation
Exportable Results	Allow users to export clustering results and visualizations
Open-Source & Extensible	Ensure code is open, well-documented, and easy to extend

---

2. Core Features

Feature	Description
Data Upload	Upload CSV files with customer data
Data Preprocessing	Basic cleaning: missing value handling, normalization
K-Means Clustering	Select number of clusters (k), run clustering, display results
Elbow Method	Plot inertia vs. k to suggest optimal cluster count
Silhouette Analysis	Plot silhouette scores for cluster quality assessment
Interactive Visualization	2D/3D scatter plots with cluster coloring, feature selection
Results Export	Download clustered data (CSV) and visualizations (PNG)
Session State	Retain user selections and results during session

---

3. User Stories

As a…	I want to…	So that…

Retail Analyst	Upload customer data and segment customers	I can identify customer groups
Data Scientist	Evaluate clustering quality with Elbow/Silhouette	I can select the optimal number of clusters
Entrepreneur	Export results and visuals	I can share insights with stakeholders

---

4. Functional Requirements

4.1 Data Handling

- Accept CSV uploads (max 10MB)
- Display data preview (first 10 rows)
- Handle missing values (drop or impute)
- Normalize numeric features (MinMax or StandardScaler)

4.2 Clustering

- User selects features and number of clusters (k: 2-10)
- Run K-Means using scikit-learn
- Display cluster assignments in data table

4.3 Evaluation

- Elbow Method: Plot inertia vs. k (range: 2-10)
- Silhouette Analysis: Plot average silhouette score vs. k

4.4 Visualization

- 2D/3D scatter plot (select features for axes)
- Color points by cluster
- Interactive plot (zoom, hover, select)

4.5 Export

- Download clustered data as CSV
- Download plots as PNG

---

5. Non-Functional Requirements

Requirement	Specification
Performance	<2s response for datasets up to 10,000 rows
Usability	Intuitive UI, minimal configuration
Security	No data stored server-side; session-based only
Extensibility	Modular codebase, clear documentation
Compatibility	Python 3.8+, Streamlit 1.0+, scikit-learn 0.24+

---

6. Technical Stack

Component	Technology
Frontend	Streamlit
Backend	Python
ML/Clustering	scikit-learn
Data Handling	pandas, numpy
Visualization	matplotlib, plotly

---

7. UI/UX Specifications

- **Sidebar:** Data upload, feature selection, clustering parameters
- **Main Area:** Data preview, evaluation plots, cluster visualizations, export buttons
- **Responsive Layout:** Works on desktop and tablet

---

8. Acceptance Criteria

ID	Criteria
AC1	User can upload CSV and preview data
AC2	User can select features and run K-Means
AC3	Elbow and Silhouette plots are generated and displayed
AC4	Interactive cluster visualization is available
AC5	Clustered data and plots can be exported
AC6	No user data persists after session ends

---

9. Out of Scope

- Supervised learning methods
- Real-time data ingestion
- User authentication or persistent storage

---

10. Example Workflow (Pseudocode)

```
# Pseudocode for main workflow
upload_data()
preprocess_data()
select_features_and_k()
run_kmeans()
plot_elbow_method()
plot_silhouette_analysis()
```

```
visualize_clusters()
export_results()
```

---

11. Milestones & Timeline

Milestone	Target Date
Requirements Finalized	Week 1
Core Functionality	Week 2-3
Visualization & Export	Week 4
Testing & Documentation	Week 5
Release	Week 6

---

12. License

- MIT License (open-source)

End of Document