

Grazioso Salvare Animal Dashboard

**Overview**

This project delivers a dynamic, interactive dashboard for Grazioso Salvare, an international organization that trains rescue dogs. Developed using Python, Dash, and MongoDB, the dashboard enables users to:  
- Filter dogs based on rescue specialization: Water Rescue, Mountain/Wilderness Rescue, and Disaster/Tracking.  
- Visualize each selected animal’s geolocation on a map.  
- Interact with a responsive data table backed by live queries to a MongoDB database.

Developed by: Tony McCormack  
Role: Software Engineering Lead, Global Rain

**Technologies Used**

- Python 3.9  
- MongoDB (hosted on nv-desktop-services.apporto.com)  
- Dash / JupyterDash  
- Dash Leaflet (map support)  
- Pandas  
- Plotly  
- Jupyter Notebook

**Project Features**

- Live connection to MongoDB with full CRUD integration  
- Interactive rescue-type filtering (radio buttons)  
- Responsive DataTable (with pagination, search, and row selection)  
- Real-time updates to a Leaflet geolocation map  
- Custom branding with Grazioso Salvare logo and developer identifier

**Installation Instructions**

1. Clone the project repository or download the files manually.  
2. Install required Python packages:  
 pip install pandas pymongo dash dash-leaflet jupyter-dash matplotlib  
3. Load the Austin Animal Center Outcomes dataset into your MongoDB instance.  
4. Confirm that the logo image grazioso\_logo.png is located in an assets directory inside your project folder.

**How to Run the Dashboard**

1. Launch Jupyter Notebook.  
2. Open the ProjectTwoDashboard.ipynb file.  
3. Run the notebook. The dashboard will open inline using:  
 app.run\_server(mode='inline')

**Folder Structure**

- aac\_crud\_module.py: CRUD operations using MongoDB  
- ProjectTwoDashboard.ipynb: Dash dashboard application with UI, filtering, and maps  
- assets/grazioso\_logo.png: Organization logo for branding  
- README.docx: Project documentation (you are reading it)

**Filter Options and Queries**

• Water Rescue:  
 Labrador Retriever, Chesapeake Bay Retriever, Newfoundland  
 Query:  
 {"animal\_type": "Dog", "breed": {"$regex": "Labrador Retriever|Chesapeake Bay Retriever|Newfoundland", "$options": "i"}}  
  
• Mountain or Wilderness Rescue:  
 German Shepherd, Alaskan Malamute, Old English Sheepdog, Siberian Husky, Rottweiler  
 Query:  
 {"animal\_type": "Dog", "breed": {"$regex": "German Shepherd|Alaskan Malamute|Old English Sheepdog|Siberian Husky|Rottweiler", "$options": "i"}}  
  
• Disaster or Individual Tracking:  
 Doberman Pinscher, German Shepherd, Golden Retriever, Bloodhound, Rottweiler  
 Query:  
 {"animal\_type": "Dog", "breed": {"$regex": "Doberman Pinscher|German Shepherd|Golden Retriever|Bloodhound|Rottweiler", "$options": "i"}}  
  
• Reset:  
 All animals (no filter)  
 Query:  
 {}

**Expected Dashboard Screens**

1. Default View: Full dataset with no filters  
2. Water Rescue Filter Applied  
3. Mountain/Wilderness Rescue Filter Applied  
4. Disaster/Tracking Filter Applied  
5. Reset View

**License**

This application is open-source and intended for use by non-profit animal rescue and training organizations. Code reuse is permitted with attribute to the developer.

A screenshot of a computer

AI-generated content may be incorrect.A screenshot of a computer

AI-generated content may be incorrect.A screenshot of a computer

AI-generated content may be incorrect.A screenshot of a data

AI-generated content may be incorrect.