Egg Price Portal - Project Report

Date: May 23, 2025

1. Introduction

1.1. Project Objective

The primary objective of the Egg Price Portal is to provide a comprehensive, user-friendly web platform for users in India to access, analyze, and visualize historical egg price data, view current market prices, and obtain future price predictions for various geographical locations, including NECC (National Egg Co-ordination Committee) recognized cities and specific districts.

1.2. Mission

Our mission is to empower consumers, traders, farmers, and analysts with timely and accurate egg price information and forecasts, enabling informed decision-making, market transparency, and better planning related to egg procurement, sales, and agricultural strategies.

1.3. Scope

This report details the architecture, functionalities, data sources, and technological stack of the enhanced Egg Price Portal, reflecting the improvements and new features implemented over its initial design.

2. System Architecture

The Egg Price Portal employs a standard client-server architecture:

- Frontend (Client-Side): A responsive web-based user interface developed using HTML5, CSS3, and JavaScript. It leverages several popular libraries to enhance interactivity, data visualization, and user experience.
- Backend (Server-Side): A Python application built with the Flask micro-framework. It serves as an API to handle data requests, perform data processing, generate price predictions, manage database interactions, and serve the frontend application.
- Databases: Two SQLite databases are used:
  - `necc_prices.db` : Stores historical daily egg price data.
  - `nearest_necc.db` : Contains mappings between districts and their nearest NECC cities, geographical coordinates for NECC cities and districts, and pre-calculated top nearby districts for NECC cities.

- Caching: FileSystemCache is implemented on the backend to improve performance by caching results of frequently accessed data and computationally intensive operations.

3. Technologies Used

3.1. Frontend

- HTML5: For the structural markup of the web pages.
- CSS3: For styling, layout, and responsive design.
  - Bootstrap 5: Utilized for its responsive grid system, pre-styled components (cards, modals, navigation, forms), and utility classes.
- JavaScript (ES6+): For client-side logic, interactivity, DOM manipulation, and API communication.
  - jQuery: Used for simplified DOM manipulation and event handling, and as a dependency for some libraries.
  - Select2: For enhanced, searchable, and multi-select dropdown menus.
  - Chart.js (v4.x): For rendering interactive and responsive charts (line, bar).
    - chartjs-adapter-moment: To integrate Chart.js time scales with Moment.js.
    - chartjs-plugin-annotation: For highlighting specific ranges or points on charts.
  - Moment.js: For parsing, validating, manipulating, and displaying dates and times.
  - Leaflet.js: For creating interactive maps.
    - Leaflet.markercluster: For clustering map markers at high zoom levels to improve performance and readability.
  - Daterangepicker.js: For a user-friendly date range selection UI, particularly for mobile filters.
  - Toastr.js: For displaying unobtrusive, non-blocking notifications (toast messages) for user feedback.

3.2. Backend

- Python (3.x): The core programming language for the backend logic.
- Flask: A lightweight WSGI web application framework used to build the API and serve the application.
- Pandas: For data manipulation and analysis, particularly for handling time-series price data.
- NumPy: For numerical operations, often used in conjunction with Pandas.
- Statsmodels: Utilized for its time-series analysis capabilities, specifically the Holt-Winters Exponential Smoothing model for price forecasting and confidence interval generation.

- SQLite3: Python's built-in library for interacting with SQLite databases.
- Flask-Caching: For implementing server-side caching mechanisms.
- Pytz: For robust timezone handling, ensuring date and time operations are consistent.

4. Key Features and Functionality

4.1. Global User Interface Elements

- Responsive Design: The portal is designed to be accessible and usable across various devices (desktops, tablets, mobile phones).
- Navigation Bar:
  - Portal branding and title.
  - "Compare Cities" button: Opens a modal for side-by-side NECC city price comparison.
  - "All Predictions" button: Opens a modal displaying consolidated prediction metrics for all NECC cities.
  - "Reset View" button: Clears all selections, filters, and resets the dashboard to its initial state.
  - Mobile Filter Toggle: On smaller screens, a "Filters" button appears, opening an off-canvas drawer for location and date selection.
- Loading Indicators:
  - Global loader overlay during initial page load or major data fetching operations.
  - Spinners within individual chart and map containers during their respective data loading/rendering phases.
- Toast Notifications: Used for providing feedback to users (e.g., success messages, warnings, errors) in a non-intrusive manner.
- Error Handling: User-friendly error messages are displayed via modals or toasts for API errors or unexpected issues.

4.2. Location and Date Selection

- Desktop Filters:
  - Select District: A searchable dropdown allowing users to select a specific district. Data displayed will be for the nearest associated NECC city.
  - Select NECC City: A searchable dropdown for users to directly select an NECC-recognized city.
  - Date Range for Price Trends: Two date input fields (`Start Date` , `End Date`) allowing users to specify a custom period for the historical price trends chart. Includes "Apply" and "Reset" buttons. Default view shows the last 5 years of monthly averages.
- Mobile Filters (Off-Canvas Drawer):

    - Provides the same "Select District" and "Select NECC City" dropdowns.
    - A single "Date Range (Price Trends)" input field utilizing `daterangepicker.js` for an optimized mobile experience.
    - An "Apply Filters" button to update the dashboard based on mobile selections.
- Selection Info Bar: Displays the currently selected location (district or NECC city) and, if a district is chosen, shows the name of the associated NECC city for which data is being displayed, along with the approximate distance.

4.3. Key Price Metrics Display

Four prominent cards display key price information for the selected location (or its associated NECC city):

- Latest Price: The most recent recorded egg price and the date of that price.
- Predicted Avg (Next Month): The forecasted average price for the immediate next calendar month. Includes the name of the predicted month.
- Predicted Avg (Next 12 Months): The forecasted average price over the next 12 months, starting from the month immediately following the current system month.
- Predicted Avg (Current Calendar Year): The forecasted average price for the entirety of the current calendar year (e.g., "Predicted Avg (2025)").

4.4. Main Data Display Tabs

4.4.1. Price Trends Tab

- Historical Price Trends Chart:
    - A line chart visualizing historical daily egg prices for the selected location/associated NECC city.
    - Responds to the date range selected in the desktop or mobile filters.
    - Defaults to showing monthly average prices for the last 5 years for better clarity on initial load or after a reset.
    - Improved line clarity for dense data by potentially using data thinning or adjusting line/point appearance.
    - Displays a "Loading..." spinner while data is being fetched and the chart is rendering.
    - Shows a "No data available..." message if no price data exists for the selection.
- Price Trend Stats: A small section below the chart displaying key statistics (e.g., Min, Max, Average price) for the data shown in the current chart view.

4.4.2. Predictions Tab

This tab consolidates all future-looking price information for the selected location.

- Historical vs. Predicted Monthly Prices Chart:
  - A line chart comparing historical monthly average prices with the predicted monthly prices for the next 24 months.
  - Optionally displays 95% confidence interval bands around the predicted prices if provided by the backend.
  - Includes a "Loading..." spinner and "No data..." message handling.
- Prediction Summary Averages Table:
  - A table summarizing key future average price predictions for the selected location:
    - Avg (Next 1 Month)
    - Avg (Next 3 Months)
    - Avg (Next 6 Months)
    - Avg (Next 9 Months)
    - Avg (Next 12 Months)
    - Avg (Current Calendar Year)
- Predicted Monthly Prices (Next 24 Months) Table:
  - A scrollable table listing the predicted price for each of the next 24 months.

4.4.3. Average Prices Tab

This tab presents aggregated historical price data.

- Historical Yearly Average Prices Chart & Table:
  - A chart (user can toggle between Bar and Line type) showing the average egg price for each year for which historical data is available.
  - A collapsible data table is displayed below the chart, displaying the raw yearly average data.
  - Includes a "Loading..." spinner and "No data..." message handling.
- Historical Monthly Average Prices (Last 5 Years) Chart & Table:
  - A chart (user can toggle between Line and Bar type) showing the monthly average egg price for the last five full years of available historical data.
  - A collapsible data table below the chart displays the raw monthly average data for this period.
  - Includes a "Loading..." spinner and "No data..." message handling.

4.4.4. Price Map Tab

This tab provides a geographical visualization of egg prices.

- Interactive Map (Leaflet.js):
  - Displays NECC cities and Districts as markers on an interactive map of India.
  - NECC City Markers: Larger, circular markers, color-coded based on their latest price level (e.g., Low, Medium-Low, Medium-High, High, No Data). Clicking an NECC city marker selects that city in the main dashboard and updates all data views.
  - District Markers: Smaller, circular markers (default color). Clicking a district marker selects that district in the main dashboard.
  - Marker Clustering: Used `Leaflet.markercluster` to group nearby markers at higher zoom levels, improving map readability and performance.
  - Map Legend: A dynamic legend clearly indicating the price bands associated with NECC city marker colors.
  - Popups & Tooltips:
    - Markers show popups on click with details (e.g., City/District Name, Latest Price for NECC cities).
    - Tooltips may provide quick info on hover.
- Line Drawing Functionality:
  - NECC City Selection: When an NECC city is selected (via dropdown or map click), lines are drawn on the map connecting it to its top 5 nearest districts (if coordinates are available).
  - District Selection: When a district is selected, a distinct line is drawn from the district's marker/location to its nearest associated NECC city's marker.
- Map Selection Info Panel:
  - Displays information about the currently selected map entity.
  - If an NECC city is selected, it lists its top 5 nearby districts with distances.
  - If a district is selected, it shows its name and the name of its nearest NECC city.
- Map Interaction:
  - The map centers on the associated NECC city (or the district itself if its coordinates are primary) when a location is selected from the dropdowns.
  - Selected NECC city/district markers are highlighted.

4.5. Modal Dialogs

4.5.1. "Compare Cities" Modal

- Allows users to select up to 4 NECC cities using a multi-select dropdown.
- The placeholder text "Select up to 4 cities" in the dropdown does not count as a selection.
- Displays side-by-side comparisons for the selected cities:
  - Key Metric Comparison Table: Shows predicted average, min, and max prices for the upcoming calendar year for each selected city.
  - Historical Price Trends Comparison Chart: A line chart comparing daily price trends (typically for the last 1 year) for the selected cities.
  - Predicted Monthly Prices Comparison Chart: A line chart comparing predicted monthly prices for the upcoming calendar year for the selected cities.
- Charts and tables update dynamically as cities are selected/deselected.
- Includes loading indicators for chart data.

4.5.2. "All Predictions" Modal (Consolidated Predictions)

- Accessed via a button in the navigation bar.
- Presents a scrollable table showing key prediction averages for *all* available NECC cities:
  - Avg (Next 1 Month)
  - Avg (Next 3 Months)
  - Avg (Next 6 Months)
  - Avg (Next 9 Months)
  - Avg (Next 12 Months)
  - Avg (Current Calendar Year - e.g., "Avg (2025)")
- Provides a quick comparative overview of predicted price levels across different NECC markets.
- Includes a loading state while data is fetched.

5. Data Sources and Backend Logic

5.1. Databases

- `necc_prices.db` :

- `DailyPrices` table: Stores historical daily egg prices.
  - Columns: `Date` (TEXT, YYYY-MM-DD), `City` (TEXT, NECC City Name), `Price` (REAL).
- `nearest_necc.db` :
  - `district_necc_map` table: Maps districts to their nearest NECC city.
    - Columns: `district` (TEXT), `necc_city` (TEXT), `distance` (REAL, in km).
  - `necc_city_coordinates` table: Stores latitude and longitude for NECC cities.
    - Columns: `city` (TEXT), `latitude` (REAL), `longitude` (REAL).
  - `district_coordinates` table: Stores latitude and longitude for districts.
    - Columns: `district_name` (TEXT, as per user confirmation), `latitude` (REAL), `longitude` (REAL).
  - `necc_top_districts` table: Stores pre-calculated top 5 nearest districts for each NECC city.
    - Columns: `necc_city` (TEXT), `nearby_district` (TEXT), `distance` (REAL), `rank` (INTEGER).

5.2. Backend API Endpoints (Flask `app.py`)

The backend exposes several API endpoints to serve data to the frontend:

- `/` : Serves the main `index.html` page.
- `/healthcheck` : Provides a basic health status of the application.
- `/api/necc_cities` : Returns a list of all NECC city names.
- `/api/districts` : Returns a list of all district names.
- `/api/district_locations` : Returns a list of all districts with their coordinates.
- `/api/prices/<city_type>/<location_name>` : Fetches historical daily prices for a given location (NECC city or district, which is then mapped to its NECC city) within an optional date range.
- `/api/averages/<city_type>/<location_name>` : Calculates and returns historical yearly and monthly average prices.
- `/api/predict/<city_type>/<location_name>` : (Preserved for potential direct calls or compatibility) Generates price predictions (24-month forecast. calendar year average, dynamic averages) for a location. Does not include confidence intervals in this specific endpoint's main payload.
- `/api/necc_cities_locations_prices` : Returns a list of all NECC cities with their coordinates and latest prices, primarily for map marker population.

- `/api/nearby_districts/<necc_city_name>` : Returns the top 5 nearest districts for a given NECC city.
- `/api/consolidated_predictions` : Fetches and returns key prediction averages for all NECC cities for the comparison table.
- `/api/city_info/<city_type>/<location_name>` : A comprehensive endpoint that returns all relevant information for a selected location in a single call, including latest price, coordinates, full predictions with confidence intervals, dynamic and calendar year averages, historical averages, and nearby districts. This is the primary endpoint used by the frontend for populating the dashboard upon location selection.
- All data-providing API endpoints include a `last_updated` (UTC ISO timestamp) field in their JSON responses.
- Error responses are standardized in JSON format (e.g., `{"error": "...", "message": "..."}`).

5.3. Prediction Model

- Model Type: Holt-Winters Exponential Smoothing.
- Training Data: Historical monthly average prices for the respective NECC city. The model is trained on data up to the end of the calendar year *preceding* the current system date (e.g., if today is May 2025, training data goes up to Dec 31, 2024).
- Minimum Data Requirement: A minimum of 36 months (3 years) of historical monthly average data is required to train the model for a city. If insufficient data is available, no forecast is generated for that city.
- Forecast Period: The model generates a 24-month forecast, starting from January of the current system year.
- Model Parameters:
  - Trend: Additive (`add`).
  - Seasonality: Multiplicative (`mul`) for monthly data.
  - Seasonal Periods: 12 (for yearly seasonality in monthly data).
- Confidence Intervals: The prediction model also generates 95% confidence intervals (lower and upper bounds) for the 24-month forecast, which are available via the `/api/city_info` endpoint and visualized on the prediction chart.
- Derived Averages:
  - Dynamic Averages: Calculated from the 24-month forecast, representing averages for the next 1, 3, 6, 9, and 12 months, starting from the month *after* the current system month.
  - Calendar Year Average: Calculated by averaging the predicted prices from the first 12 months of the 24-month forecast (which correspond to the months of the current calendar year).

6. Recent Enhancements Summary

This version of the Egg Price Portal includes significant enhancements over its initial design:

- Improved UI/UX: Off-canvas mobile filters, refined loading states (global, chart-specific, map-specific), toast notifications, more intuitive date filtering, and clearer tab organization.
- Enhanced Charting: Ability to toggle chart types for averages, clearer presentation of prediction data with confidence intervals, and improved handling of "no data" states.
- Advanced Map Features: Inclusion of district markers, marker clustering, dynamic map legend, and more explicit line drawing between selected locations and their associated/nearby counterparts.
- Comprehensive Backend API: Introduction of the `/api/city_info` endpoint to streamline data fetching, addition of confidence intervals to predictions, `last_updated` timestamps, and standardized error responses.
- Robustness and Maintainability: Switch to FileSystemCache, improved error handling and logging on the backend, and more resilient frontend data processing.
- Database Schema Awareness: Code updated to reflect confirmed or assumed database column names more accurately, with fallbacks for potential discrepancies.

7. Future Considerations / Potential Next Steps

- User Accounts & Personalization: Allow users to save favorite locations or comparison sets.
- Advanced Prediction Analytics: Display model accuracy metrics (e.g., MAPE, RMSE) if logged during training. Offer alternative forecasting models.
- Data Export: Allow users to export chart data or tables (e.g., to CSV, Excel).
- API Pagination: For endpoints returning large lists (e.g., `/api/consolidated_predictions`). If the number of NECC cities grows significantly.
- More Granular Caching Strategies: Implement more sophisticated cache invalidation or finer-grained caching for specific data elements.
- Unit & Integration Testing: Develop a comprehensive test suite for both frontend and backend components.
- Internationalization (i18n): Support for multiple languages if the portal's scope expands.

8. Conclusion

The enhanced Egg Price Portal provides a powerful and user-friendly tool for accessing and analyzing egg price data in India. The recent improvements have significantly uplifted its functionality, user experience, and backend robustness, making it a more valuable resource for its target audience. Continued development and incorporation of user feedback will further refine its capabilities.