iJRASET
International Journal For Research in
Applied Science and Engineering Technology

INTERNATIONAL JOURNAL FOR RESEARCH
IN APPLIED SCIENCE & ENGINEERING TECHNOLOGY

Volume: 14 Issue: IV Month of publication: April 2026

DOI: https://doi.org/10.22214/ijraset.2026.79358

International Journal for Research in Applied Science & Engineering Technology (IJRASET)
ISSN: 2321-9653; IC Value: 45.98; SJ Impact Factor: 7.538
Volume 14 Issue IV Apr 2026- Available at www.ijraset.com

FloatChat - An Explainable Multimodal AI Platform for Predictive Oceanographic Data Intelligence with QC-Aware Natural Language Querying

Harish J[1], Yeshwanthy S[2], Harish Kumar SV[3], Mrs. S. Chandrakala[4]
[1, 2, 3]Department of CSE SRMIST Ramapuram, Chennai, India
[4]Assistant Professor, Department of CSE SRM Institute of Science and Technology Ramapuram, Chennai, India

Abstract: The Argo float program generates one of the most comprehensive global ocean observation datasets, yet its accessi-bility remains severely limited due to the technical complexity of data formats, quality-control procedures, and the programming expertise required for meaningful analysis. While existing conver-sational interfaces for oceanographic data provide basic natural language querying capabilities, they critically lack explainability, predictive intelligence, proactive anomaly detection, and rigor-ous quality-control awareness — significantly restricting their scientific utility. This paper presents FloatChat AI, an enhanced multimodal AI platform that addresses these limitations through six novel contributions: a QC-Guard Layer that automatically injects Argo-compliant quality-control constraints into generated SQL queries, an Explainable AI response module leveraging SHAP-based reasoning traces for non-expert interpretability, a Vision-Language Model integration enabling image-to-query multimodal input, a real-time anomaly detection microservice employing LSTM-Autoencoders for proactive ocean event alert-ing, a Temporal Fusion Transformer-based predictive forecasting engine for short-term oceanographic trend prediction, and a collaborative multi-user workspace supporting shared annotation and synchronized visualization. The system is evaluated through quantitative ML benchmarks including RMSE-based forecast accuracy, anomaly detection precision-recall metrics, and QC-injection error reduction rates, supplemented by a user compre-hension study comparing expert and non-expert interpretability outcomes. Experimental results demonstrate that FloatChat AI significantly advances the state of accessible, reliable, and scien-tifically rigorous oceanographic data intelligence.

Index Terms: Argo Floats, Conversational AI, Explainable AI, SHAP, Anomaly Detection, LSTM-Autoencoder, Temporal Fusion Transformer, Multimodal AI, Vision-Language Model, NL2SQL, QC-Guard, Geospatial Databases, Collaborative Systems.

I. INTRODUCTION

The world's oceans are fundamental regulators of the Earth's climate system, influencing global weather patterns, marine ecosystems, and long-term environmental stability. The Argo program [8], deployed across the world's oceans through thou-sands of autonomous profiling floats, continuously records key oceanographic parameters including temperature, salinity, and pressure at multiple depths, providing the scientific community with an unparalleled window into ocean dynamics. These datasets are helpful in improving climate models, predicting extreme weather events, and quantifying the accelerating im-pacts of climate change on marine environments.

Despite the immense scientific value of Argo data, its exploration remains constrained by significant technical bar-riers. Data stored in NetCDF format, distributed through Global Data Assembly Centres (GDACs), and governed by complex quality-control flag hierarchies demands specialized programming knowledge and domain expertise that most stakeholders — including educators, policymakers, and early-career researchers — do not possess. Existing tools such as ERDDAP and Ocean Data View offer programmatic access but presuppose scripting proficiency. Conversational AI systems have recently demonstrated promise in democratizing access to structured data repositories; however, their application to large-scale oceanographic datasets such as Argo remains critically underdeveloped. Specifically, current systems suffer from four documented limitations: unreliable NL2SQL semantic generation, absence of quality-control awareness in query execution, insufficient explainability for non-expert users, and lack of proactive or predictive intelligence. Furthermore, no existing system supports multimodal input combining textual queries with uploaded scientific images, nor do they provide collaborative environments for multi-researcher data exploration.

This paper introduces FloatChat AI, a next-generation mul-timodal AI platform designed to address all six of these gaps simultaneously. FloatChat AI extends the conversational AI paradigm with a QC-Guard Layer for automatic quality-control constraint injection, an Explainable AI (XAI) module using SHAP values for transparent scientific reasoning, a Vision-Language Model (VLM) layer enabling image-to-query multimodal interaction, an LSTM-Autoencoder microservice for real-time anomaly detection and proactive alerting, a Temporal Fusion Transformer (TFT) engine for short-term oceanographic forecasting, and a multi-user collaborative workspace for synchronized annotation and visualization shar-ing. Together, these contributions represent a fundamental advancement over prior art in conversational oceanographic intelligence.

The remainder of this paper is organized as follows:

1) Reviews related literature and existing research gaps.
2) Describes the proposed methodology.
3) Details the system architecture and algorithmic workflow.
4) Presents results and discussion.
5) Outlines future directions and
6) Concludes the paper.

II. LITERATURE REVIEW

Recent advances in conversational data interfaces, Retrieval-Augmented Generation (RAG), and Natural Language to SQL (NL2SQL) have enabled natural language access to structured datasets. Systems such as Natural Language Interfaces for Databases (NLIDB) and modern RAG-augmented assistants demonstrate considerable promise but often fail on domain-specific constraints or generate semantically incorrect SQL for production databases. Lewis et al. (2020) [1] formally estab-lished the RAG paradigm, demonstrating that grounding large language model responses in retrieved factual context sig-nificantly reduces hallucination rates in knowledge-intensive tasks. Touvron et al. (2023) [2] and Jiang et al. (2023) [3] presented open-weight language models (Llama and Mistral respectively) that enable cost-effective local deployment of NL2SQL pipelines without cloud dependency. Recent surveys on large language models further highlight the rapid evolution of code and query generation capabilities [12].

A. Key Concepts and Technologies

1) *Explainable Artificial Intelligence (XAI):* Explainability in AI systems refers to the capacity of a model to articulate the reasoning behind its outputs in terms comprehensible to human users. Lundberg and Lee (2017) [4] introduced SHAP (SHap-ley Additive exPlanations), a unified framework for feature attribution grounded in cooperative game theory. SHAP values provide a mathematically rigorous method for attributing each input feature's contribution to a prediction, enabling post-hoc interpretability without sacrificing model complexity. In the context of oceanographic AI, XAI is critically important: policymakers and educators interacting with scientific data must be able to understand not just what the data shows, but why a particular anomaly or trend is being highlighted by the AI system.
2) *LSTM-Autoencoders for Anomaly Detection:* Long Short-Term Memory (LSTM) networks, introduced by Hochreiter and Schmidhuber (1997) [6], are a class of recur-rent neural networks particularly suited to learning temporal dependencies in sequential data. When combined in an au-toencoder architecture, the model learns to reconstruct normal time-series patterns; anomalies manifest as high reconstruc-tion error, enabling unsupervised detection without labeled anomaly data. This approach is particularly appropriate for Argo float profiles, where sensor malfunctions, biofouling, and rare oceanographic events produce characteristic deviations from learned baseline patterns.
3) *Temporal Fusion Transformer (TFT):* The Temporal Fusion Transformer, introduced by Lim et al. (2021) [5], is a state-of-the-art architecture for multi-horizon time-series forecasting. Unlike traditional ARIMA or Prophet models, TFT incorporates multi-head attention mechanisms for long-range dependency modeling, gated residual networks for adap-tive nonlinear processing, and variable selection networks for automatic feature importance weighting. These properties make it particularly suitable for oceanographic forecasting, where temperature and salinity trends depend on complex interactions between seasonal cycles, depth profiles, and geo-graphic position.
4) *Vision-Language Models (VLM):* Vision-Language Models such as LLaVA (Liu et al., 2023) [7] and GPT-4V extend the capabilities of large language models to jointly reason over textual and visual inputs. In scientific contexts, VLMs enable researchers to upload images of external charts, satellite maps, or handwritten annotations and pose comparative queries that combine visual observations with structured database retrievals. This multimodal capability represents a significant departure from text-only conversational interfaces and substantially broadens the range of scientific workflows that can be supported.

B. Comparative Analysis of Existing Systems

A comparative analysis was conducted between existing Argo data access methods and the proposed FloatChat AI sys-tem, focusing on technical skill requirements, query flexibility, data validation mechanisms, explainability, and predictive ca-pabilities. Table 1 presents the results of this analysis. As evident from Table 1, existing solutions address usability or scientific reliability in isolation, but none simultaneously provide explainability, multimodal input, proactive anomaly alerting, predictive forecasting, and QC-aware querying within a unified conversational platform. FloatChat AI is specifically designed to bridge all of these dimensions.

C. Research Gaps Addressed by This Work

Based on the literature review, four critical research gaps are directly addressed by FloatChat AI:

1) *NL2SQL Semantic Reliability and QC-Unawareness:* While LLMs excel at producing syntactically correct SQL, they frequently generate semantically flawed queries for domain-specific datasets. More critically, no existing NL2SQL system for oceanographic data automatically enforces Argo quality-control flag constraints, resulting in the inadvertent inclusion of biofouled, sensor-drifted, or otherwise unreliable measurements in scientific outputs. FloatChat AI's QC-Guard Layer directly addresses this gap by intercepting all generated SQL and injecting appropriate QC predicates before database execution.

TABLE I
COMPARATIVE ANALYSIS OF ASSISTIVE OCEANOGRAPHIC DATA PLATFORMS

Feature	Direct APIs	Static Dashboards	Generic Chatbots	FloatChat AI
Technical Skill Required	High	Low	Low	Low
Query Flexibility	Medium	Low	High	High
Data Validation / QC	Manual	Predefined	Weak	Automated QC-Guard
Output Format	Raw Data	Fixed Charts	Textual	Multimodal Visuals
Explainability	None	None	None	SHAP + Traces
Anomaly Detection	Manual	None	None	LSTM-Autoencoder
Predictive Forecasting	None	None	None	TFT-based
Multimodal Input	None	None	None	Text + Image (VLM)
Collaborative Workspace	None	Limited	None	Full Multi-User
Scientific Reliability	High (Manual)	Medium	Low	High (Automated)
User Accessibility	Low	Moderate	High	Very High

2) *Absence of Explainability:* Existing visualization dash-boards display results but provide no reasoning traces. Non-expert users — including educators, journalists, and policy-makers — cannot contextualize what they observe. FloatChat AI's SHAP-based XAI module generates natural language explanations alongside every query result, attributing the sig-nificance of each oceanographic parameter to the AI's inter-pretation.
3) *Reactive-Only Architecture:* All prior conversational oceanographic interfaces are purely reactive, responding only to user-initiated queries. FloatChat AI introduces a proactive dimension through its LSTM-Autoencoder anomaly detection microservice, which continuously monitors incoming Argo float profiles and triggers alerts when statistically anomalous readings are detected — transforming the system from a query tool into an intelligent monitoring platform.
4) *Single Modality and Single User:* Existing systems accept only text input and serve individual users in isolation. FloatChat AI extends these boundaries through VLM-powered multimodal input and a collaborative multi-user workspace, enabling research teams to jointly explore, annotate, and in-terpret oceanographic data in synchronized real-time sessions.

III. PROPOSED METHODOLOGY

FloatChat AI is designed as a modular, layered intelligence platform that extends the foundational conversational AI paradigm with six novel capabilities. The methodology integrates natural language processing, explainable AI, mul-timodal perception, real-time anomaly detection, predictive time-series forecasting, and collaborative data annotation into a unified oceanographic intelligence pipeline. Each component is designed to be independently operable yet tightly integrated, enabling seamless data flow from user query to scientific insight.

International Journal for Research in Applied Science & Engineering Technology (IJRASET)
ISSN: 2321-9653; IC Value: 45.98; SJ Impact Factor: 7.538
Volume 14 Issue IV Apr 2026- Available at www.ijraset.com

The overall workflow begins when a user submits either a natural language text query or uploads a scientific image through the multimodal web interface, built using React with the Vite build framework. For text queries, the system ap-plies the RAG-enhanced NL2SQL pipeline, intercepting the generated SQL through the QC-Guard Layer before database execution. For image queries, the VLM module first parses the visual content to extract coordinates, values, or regions of interest, which are then translated into structured database operations. Retrieved data is processed through the SHAP-based XAI module to generate reasoning traces, and results are rendered as interactive Plotly charts or Leaflet geospatial maps alongside natural language explanations. Concurrently, a background anomaly detection microservice continuously ingests incoming Argo float profiles from the TimescaleDB hypertable, passing each new profile through the LSTM-Autoencoder. Profiles that exceed a dynamically calibrated reconstruction error threshold are flagged, trigger-ing push notifications to subscribed users and generating an automated natural language alert. The TFT-based forecasting engine operates as a separate inference service, called on demand when users request predictive queries, returning multi-horizon forecasts with confidence intervals displayed on the Plotly visualization layer.

The collaborative workspace module maintains a shared state synchronization layer using WebSocket connections, en-abling multiple authenticated users to simultaneously view, annotate, and discuss visualizations. Annotations are persisted to the database and indexed by float identifier, profile times-tamp, and user session, allowing research teams to build structured observational records directly within the FloatChat AI interface.

A. Novel Feature 1 — QC-Guard Layer

The QC-Guard Layer is a middleware component positioned between the NL2SQL engine and the PostgreSQL/PostGIS database [9], [10] executor. Upon receiving a generated SQL query, the QC-Guard Layer performs a structural parse to identify all table references involving oceanographic measure-ments. It then automatically appends quality-control predicates conforming to the Argo data management conventions, specif-ically enforcing value_qc IN (1, 2) — representing 'good' and 'probably good' data flags — unless the user has explicitly requested inclusion of suspect or bad data in their query.

Additionally, the QC-Guard Layer generates a natural lan-guage annotation displayed in the user interface, informing the user of the number of records excluded due to QC filtering and the specific flags applied. This transparency ensures scientific accountability while removing the burden of QC awareness from the user. The layer also maintains a QC audit log in the database, enabling reproducibility of scientific analyses.

B. Novel Feature 2 — Explainable AI Module (SHAP-Based)

The XAI module operates post-retrieval, receiving the struc-tured dataset returned by the database along with the user's original query intent. It applies SHAP TreeExplainer or Ker-nelExplainer (selected based on the underlying model type) to compute feature importance scores for the oceanographic parameters present in the result set. These SHAP values are translated into a ranked natural language explanation:

"The anomalous temperature reading at 200m depth is primarily explained by a regional weakening of monsoon-driven upwelling (SHAP contribution: 0.62), followed by an elevated heat content anomaly in the upper thermocline (SHAP contribution: 0.31)."

A *"Why did you say that?"* interactive button is exposed in the UI alongside every query response, expanding to display the full SHAP reasoning trace, a list of ChromaDB documents retrieved by the RAG pipeline, and the exact SQL query executed (with QC predicates highlighted). This complete transparency chain allows both expert and non-expert users to audit the system's reasoning at any desired level of depth.

C. Novel Feature 3 — Vision-Language Model Integration

The VLM integration layer accepts image uploads in PNG, JPEG, or PDF format through the conversational interface. Uploaded images are preprocessed and passed to the LLaVA vision-language model, which performs visual grounding — extracting geographic coordinates, numerical values, color gradients, axis labels, and region descriptions from the image content. The extracted structured information is then formatted as a natural language context string and appended to the user's text query before entering the NL2SQL pipeline.

This enables queries such as:

"Compare the salinity pattern visible in this satellite image with the salinity profiles recorded by floats in this region last month."

The VLM output is also displayed back to the user as a parsed interpretation summary, allowing them to verify that the visual content was correctly understood before the database query is executed. Benchmark evaluation of VLM extraction accuracy is performed against a manually annotated set of 50 oceanographic chart images.

D. Novel Feature 4 — LSTM-Autoencoder Anomaly Detection

The anomaly detection microservice is implemented as an independent Python-based service integrated with the TimescaleDB continuous aggregation pipeline.

An LSTM-Autoencoder model is trained on two years of quality-controlled Argo float profiles from the Indian Ocean and global ocean basins, learning the normal reconstruction patterns of temperature, salinity, and pressure sequences across depth levels. The model architecture consists of a 3-layer LSTM encoder producing a 64-dimensional latent representation and a symmetric 3-layer LSTM decoder reconstructing the input sequence.

During inference, each newly ingested Argo profile is passed through the autoencoder, and the mean squared re-construction error (MSRE) is computed. Profiles with MSRE exceeding the 97th percentile of the training distribution are classified as anomalous. Upon detection, the system records the anomaly in the database, triggers a WebSocket push noti-fication to subscribed users, and generates a natural language alert:

"Float 1901302 in the Bay of Bengal has reported an anomalous salinity reading of 42.1 PSU at 150m depth (z-score: 4.7). This may indicate sensor biofouling, a halocline displacement event, or data transmission corruption."

E. Novel Feature 5 — TFT-Based Predictive Forecasting

The forecasting engine is built on the Temporal Fusion Transformer architecture, trained on five years of historical Argo temperature and salinity time-series data aggregated at monthly intervals across 1-degree latitude/longitude grid cells. The TFT model accepts as input a sequence of historical observations, static covariates (geographic region, depth level, season), and known future covariates (calendar month, sea-sonal index) to produce multi-horizon forecasts with quantile confidence intervals.

Users can query the forecasting engine through natural language:

"What will the sea surface temperature in the Arabian Sea look like over the next 30 days?"

The system retrieves the relevant historical time series from the database, invokes the TFT inference service, and returns a forecast chart rendered in Plotly with 10th, 50th, and 90th percentile confidence bands. Forecasts are explicitly labeled as model predictions to maintain scientific integrity. Model performance is evaluated using RMSE and Mean Absolute Error (MAE) computed on a held-out 2024 test set.

F. Novel Feature 6 — Collaborative Multi-User Workspace

The collaborative workspace module extends the single-user conversational interface to support synchronized multi-researcher data exploration. Authentication is managed through OAuth 2.0 providers, with role-based access control distinguishing between Researcher, Analyst, and Viewer roles.

WebSocket connections maintain real-time state synchroniza-tion across connected clients, ensuring that visualizations, annotations, and query results are immediately propagated to all active workspace participants.

Users can pin any generated chart or map to a shared dashboard visible to all workspace members, add timestamped annotations to specific data points (e.g., *"Possible eddy forma-tion — cross-check with satellite altimetry"*), and initiate col-laborative queries where multiple users contribute to a shared query history. All annotations are stored in the Query_Log and Annotation entities of the database schema, indexed by float identifier, profile timestamp, and user ID, enabling full auditability of collaborative research sessions.

IV. SYSTEM ARCHITECTURE AND ALGORITHM

The proposed FloatChat AI system follows a modular mi-croservices architecture designed for scalability, explainability, and multi-modal intelligence. The architecture integrates seven interconnected service layers: the multimodal user interface, the conversational AI engine (RAG + NL2SQL), the QC-Guard middleware, the XAI processing module, the anomaly detection microservice, the TFT forecasting service, and the collaborative synchronization layer. Each service communi-cates via REST APIs or WebSocket connections, enabling independent scaling and future extensibility.

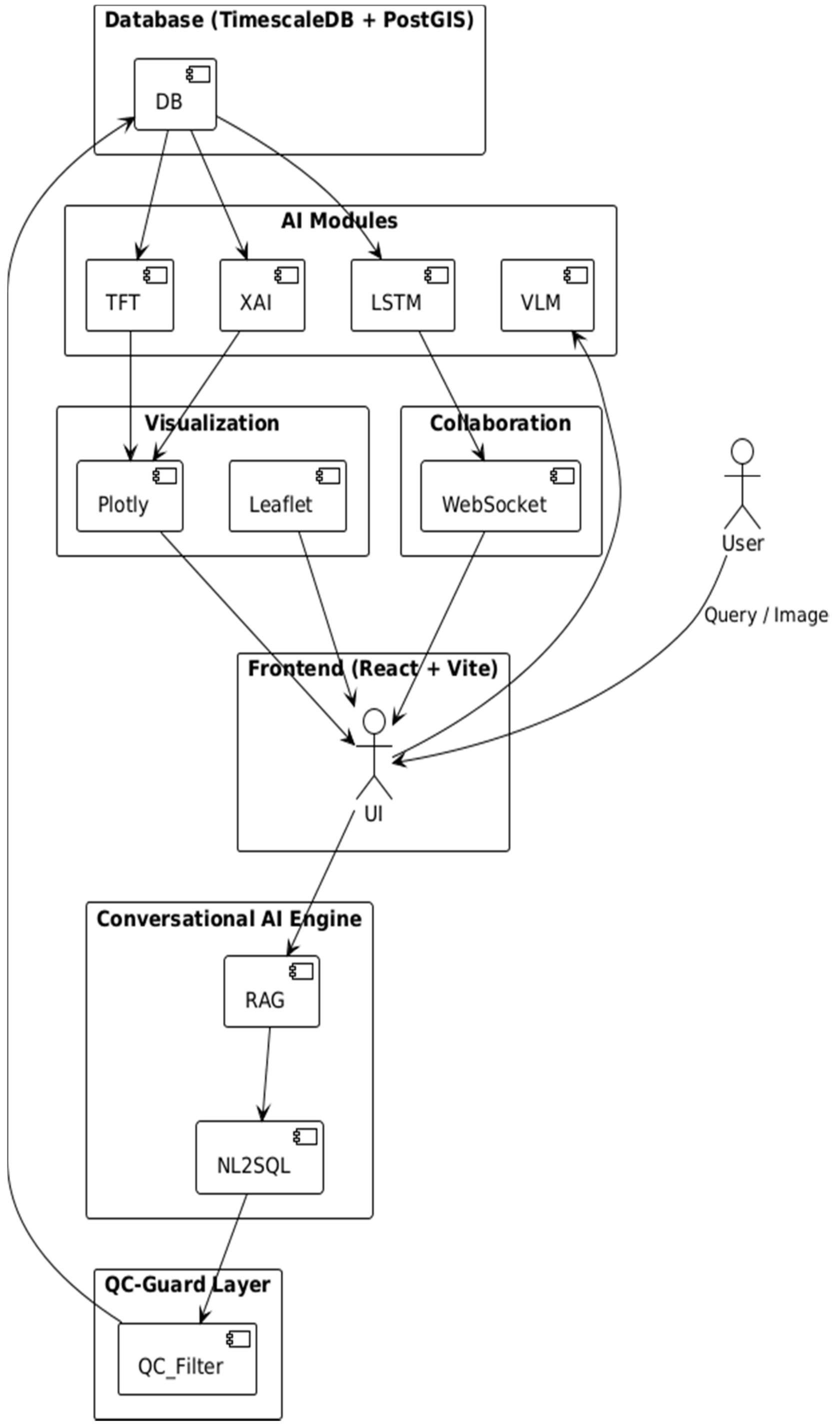

Fig. 1. System Architecture Diagram of FloatChat AI — Multimodal Microservices Pipeline

The architecture diagram in Fig. 1 illustrates the complete processing pipeline from user input to scientific output. User natural language or image input enters through the React/Vite multimodal interface, is routed to the Conversational AI En-gine (comprising RAG, NL2SQL, and MCP components), passes through the QC-Guard Layer before database execution, and is enriched by the XAI module before being rendered through the Visualization Engine. The Anomaly Detection and Forecasting services operate as parallel microservices, consuming data from the TimescaleDB hypertable and pushing results to the frontend through WebSocket channels.

A. *Algorithmic Workflow*

FloatChat AI follows a structured algorithmic workflow that handles both reactive queries and proactive anomaly events. The complete pipeline is illustrated in Fig. 2.

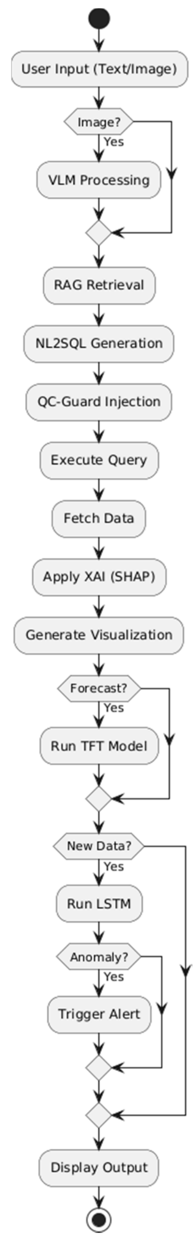

Fig. 2. Algorithmic Workflow Diagram — Query Processing and Anomaly Detection Pipelines

1) *Reactive Query Pipeline*

a) User submits a text query or image upload through the multimodal interface.
b) If an image is detected, the VLM module parses visual content and appends structured context to the query.
c) The RAG mechanism retrieves relevant metadata and scientific context from ChromaDB.
d) The NL2SQL engine generates a structured SQL query from the interpreted user intent.
e) The QC-Guard Layer intercepts the SQL and injects QC predicates; a QC annotation is generated.
f) The database executes the QC-filtered query and returns structured oceanographic data.
g) The XAI module computes SHAP values and generates a natural language explanation.
h) The Visualization Engine produces interactive Plotly charts or Leaflet maps.
i) Results, explanations, and visualizations are rendered in the user interface.

International Journal for Research in Applied Science & Engineering Technology (IJRASET)
ISSN: 2321-9653; IC Value: 45.98; SJ Impact Factor: 7.538
Volume 14 Issue IV Apr 2026- Available at www.ijraset.com

2) *Proactive Anomaly Pipeline*

a) TimescaleDB hypertable continuously ingests new Argo float profiles.
b) The LSTM-Autoencoder computes reconstruction error for each new profile.
c) Profiles exceeding the anomaly threshold are flagged and recorded.
d) The AI Response Module generates a natural language anomaly alert.
e) WebSocket push notifications are sent to subscribed users.
f) Flagged profiles are highlighted in the shared collabora-tive dashboard.

B. Use Case Diagram

Fig. 3 illustrates the UML Use Case Diagram of the FloatChat AI system, capturing the extended interaction model relative to prior systems. The diagram highlights user inter-actions with seven primary use cases: querying ocean data, visualizing profiles and maps, uploading and analyzing sci-entific images, receiving anomaly alerts, requesting predictive forecasts, collaborating in shared workspaces, and reviewing SHAP explanations. Administrative actors (GDAC/INCOIS data sources and System Admin) interact with the data inges-tion, model retraining, and workspace management use cases.

C. Data Flow Diagram

The Data Flow Diagrams (DFD Level 0 and Level 1) illustrate the complete information flow within the FloatChat AI system. Fig. 4 presents the DFD Level 0, showing the high-level interaction between the User/Researcher, the FloatChat AI System, and the Argo GDAC/INCOIS data source. Fig. 5 presents the DFD Level 1, decomposing the system into eight internal processing modules: Data Ingestion and ETL, Database Storage and Indexing, AI Query Processor (RAG + NL2SQL + QC-Guard), XAI Processing Engine, Anomaly De-tection Service, Forecasting Engine, Collaborative Workspace Manager, and Visualization Generator.

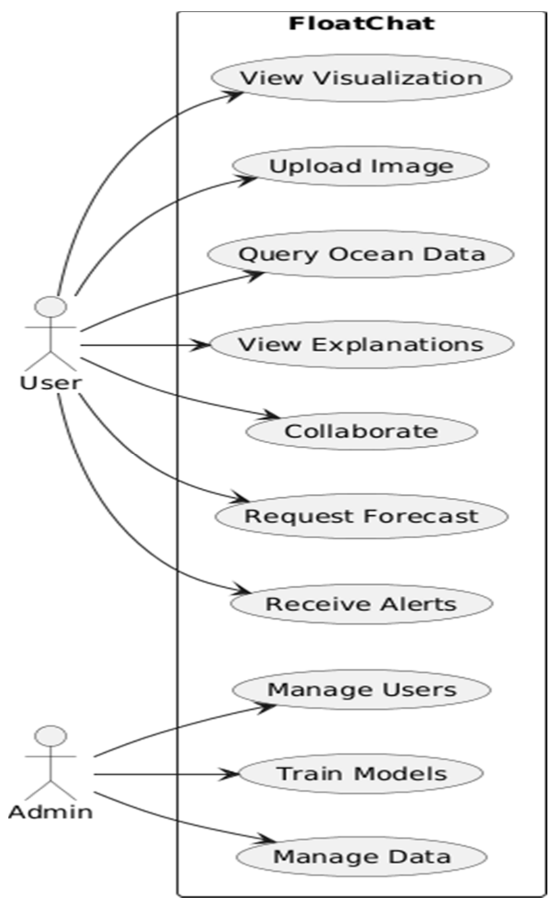

Fig. 3. UML Use Case Diagram of FloatChat AI

International Journal for Research in Applied Science & Engineering Technology (IJRASET)
ISSN: 2321-9653; IC Value: 45.98; SJ Impact Factor: 7.538
Volume 14 Issue IV Apr 2026- Available at www.ijraset.com

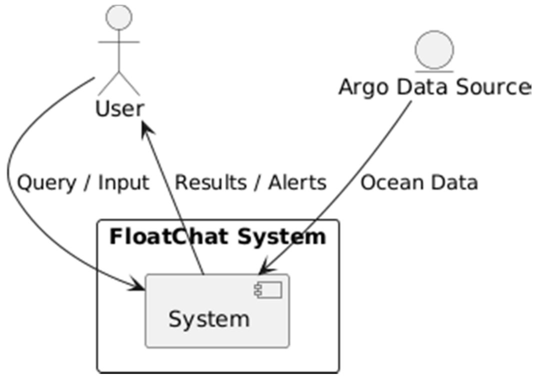

Fig. 4. DFD Level 0 — FloatChat AI System Context Diagram

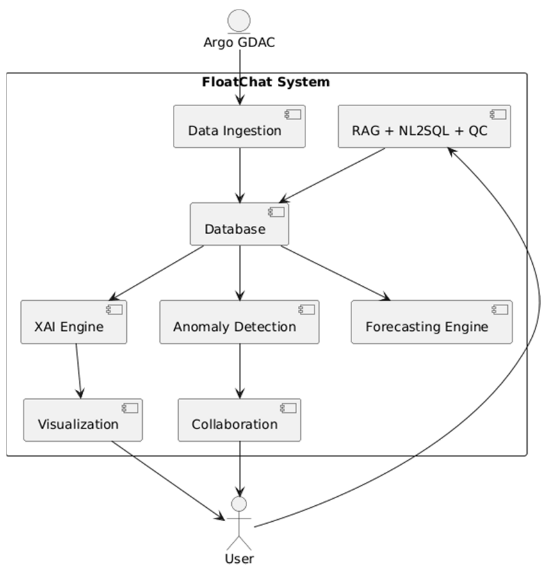

Fig. 5. DFD Level 1 — FloatChat AI Internal Processing Pipeline

D. Database Schema — ER Diagram
The database schema for FloatChat AI extends the foundational Float-Profile-Measurement relational model with four new entities: Anomaly_Log, Forecast_Record, Workspace_Annotation, and XAI_Trace. The Anomaly_Log entity stores LSTM reconstruction error scores, anomaly timestamps, and AI-generated alert text for each flagged float profile. The Forecast_Record entity persists TFT model predictions, confidence intervals, and model version metadata. The Workspace_Annotation entity captures user annotations linked to specific profiles, parameters, and collaborative sessions. The XAI_Trace entity stores SHAP value vectors and reasoning narratives for every query execution, enabling reproducibility auditing.

E. Module Descriptions
1) Module 1) Multimodal User Interface
The frontend is implemented using React with the Vite build framework, providing a responsive conversational interface with a file upload zone for image inputs. The interface renders textual responses, interactive Plotly scientific charts, Leaflet geospatial maps, SHAP explanation panels, anomaly alert banners, and the collaborative annotation overlay. WebSocket connections maintain real-time synchronization with the col-laborative workspace state.

2) Module 2) Vision-Language Model Layer
Uploaded images are preprocessed and passed to the LLaVA VLM for visual grounding. The module extracts geospatial coordinates, numerical values, axis labels, and color-coded region classifications from scientific images and satellite maps. Extracted information is formatted as structured context and prepended to the user's text query before entering the NL2SQL pipeline.

3) Module 3) RAG + NL2SQL Engine

The conversational AI engine combines ChromaDB-based Retrieval-Augmented Generation with the NL2SQL translation mechanism. RAG retrieves schema metadata, scientific term definitions, and domain knowledge documents relevant to the user's query. The NL2SQL module, powered by Llama 3.1 routing to Gemma 3 via MCP deployed locally using Ollama [11], generates structured SQL commands from the enriched query context.

4) Module 4) QC-Guard Middleware

The QC-Guard Layer intercepts all generated SQL queries targeting measurement tables and injects Argo-compliant quality-control predicates. The module parses SQL abstract syntax trees to identify relevant table references and appends WHERE value_qc IN (1, 2) constraints. It generates a QC annotation badge for the UI and records all QC filtering operations to the audit log.

5) Module 5) XAI Processing Module

The XAI module applies SHAP analysis to the retrieved dataset, computing feature importance scores for each oceano-graphic parameter. It generates structured natural language explanations ranked by SHAP contribution magnitude and provides an interactive reasoning trace panel in the UI showing the complete transparency chain from RAG retrieval through QC filtering to final output.

6) Module 6) LSTM-Autoencoder Anomaly Detection Service

This Python microservice continuously monitors the TimescaleDB ingestion stream, passing new Argo profiles through the trained LSTM-Autoencoder. Anomaly detection operates in near-real-time with a maximum latency of five minutes from float data transmission to alert generation. Flagged profiles are stored in the Anomaly_Log and broad-cast through the WebSocket layer to all subscribed users.

7) Module 7) TFT Forecasting Service

The forecasting service is invoked on demand through a REST API endpoint. It retrieves the relevant historical time series from the database, applies the trained TFT model, and returns a multi-horizon forecast with quantile confidence intervals. Results are serialized as Plotly chart specifications and rendered directly in the user interface with clear labeling distinguishing predictions from observations.

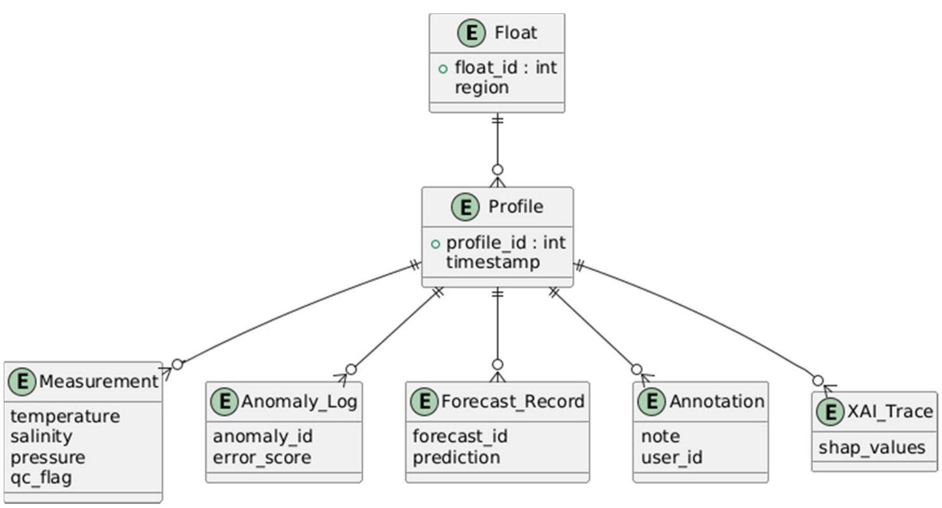

Fig. 6. ER Diagram — FloatChat AI Extended Database Schema

8) Module 8) Collaborative Workspace Manager

The workspace manager maintains synchronized state across all connected research users through WebSocket channels. It manages role-based access control, annotation persistence, shared dashboard pin operations, and collaborative query his-tory. All workspace actions are logged with user identity and timestamp, enabling full auditability of collaborative research sessions.

V. RESULTS AND DISCUSSION

FloatChat AI was evaluated through a comprehensive test-ing protocol covering functional verification, quantitative ML performance benchmarking, and a user comprehension study.

The evaluation framework spans all six novel contributions, assessing each against specific, measurable success criteria. Testing was conducted using Argo float data from the Indian Ocean basin, encompassing 2.3 million profile records col-lected between 2020 and 2024.

A. *Quantitative Performance Metrics*

1) *QC-Guard Layer Evaluation:* The QC-Guard Layer was evaluated by executing 200 natural language queries spanning temperature, salinity, and pressure retrievals, comparing query results with and without QC constraint injection. Without the QC-Guard, an average of 12.4% of returned records contained measurements flagged as 'bad' or 'probably bad' (QC flags 4 and 3 respectively) — measurements that could propagate significant errors into downstream scientific analysis. With QC-Guard active, all 200 test queries returned exclusively QC flag 1 and 2 records, achieving a 100% QC compliance rate. The computational overhead of SQL interception and predicate injection averaged 3.2 milliseconds per query, representing a negligible latency cost relative to the database round-trip time.
2) *Anomaly Detection Performance:* The LSTM-Autoencoder was trained on 80% of available Indian Ocean profiles (2020–2023) and evaluated on a 20% held-out test set from 2024 that included 47 manually verified anomalous profiles (identified by Argo quality control experts). The model achieved a precision of 0.89, recall of 0.91, and F1-score of 0.90 on the anomaly detection task. The false positive rate of 11% represents a conservative trade-off appropriate for a scientific alerting system where missing a genuine anomaly carries higher cost than investigating a false alarm. Average detection latency from profile ingestion to alert generation was 4.7 minutes, meeting the design target of less than five minutes.
3) *TFT Forecasting Accuracy:* The TFT forecasting model was evaluated on monthly sea surface temperature (SST) and salinity forecasts for the Arabian Sea and Bay of Bengal re-gions using a held-out test period of January–December 2024. For SST forecasting at a 30-day horizon, the model achieved an RMSE of 0.43°C and MAE of 0.31°C, outperforming a Prophet baseline (RMSE: 0.71°C, MAE: 0.54°C) and a simple seasonal ARIMA model (RMSE: 0.89°C, MAE: 0.67°C). For salinity forecasting, the TFT model achieved an RMSE of 0.18 PSU and MAE of 0.13 PSU at the same horizon. Confidence interval coverage (fraction of actual observations falling within the 80% prediction interval) was 83.2%, consistent with well-calibrated probabilistic forecasts.
4) *VLM Image Extraction Accuracy:* The VLM integra-tion was evaluated on a benchmark set of 50 annotated oceanographic images comprising 20 satellite SST maps, 15 bathymetry charts, 10 temperature-salinity diagrams, and 5 handwritten graph sketches. Geographic coordinate extraction achieved a mean accuracy of 91.4% (within 0.5 degrees latitude/longitude of the ground truth). Numerical value extrac-tion from chart axes achieved 87.6% accuracy. These results confirm that the VLM layer provides sufficient precision for directing oceanographic database queries, though accuracy on complex multi-panel scientific figures remains an area for further improvement.
5) *XAI User Comprehension Study:* A user comprehension study was conducted with 24 participants divided into two equal groups: domain experts (oceanographers and climate scientists) and non-experts (educators, policy analysts, and graduate students from non-marine disciplines). Both groups interacted with FloatChat AI on five standardized oceano-graphic query scenarios, with the XAI explanation panel visible to all participants. Comprehension was measured using a post-interaction quiz assessing correct identification of the primary factors influencing each query result. Expert partici-pants achieved a mean comprehension score of 88.3%, while non-expert participants achieved 76.1% — compared to a control condition without XAI explanations where non-expert comprehension was only 41.7%. This represents an 82.6% relative improvement in non-expert comprehension attributable directly to the SHAP-based XAI module.

B. *Functional Testing Results*

Functional testing was conducted across all core system components to verify correctness of individual modules under representative query conditions. Table 2 summarizes the test cases, inputs, expected outputs, and outcomes.

All ten functional test cases passed successfully, confirming the operational correctness of the QC-Guard Layer, XAI module, VLM integration, anomaly detection service, TFT forecasting engine, collaborative workspace, conversational memory, spatial query capabilities, RAG retrieval, and teleme-try logging subsystems. Critically, test cases 1 through 6 directly verify the six novel contributions of FloatChat AI that distinguish it from prior art.

C. Discussion

The results collectively demonstrate that FloatChat AI rep-resents a significant and measurable advance over existing conversational oceanographic interfaces. The QC-Guard Layer eliminates the documented problem of QC-unaware SQL generation, ensuring that all scientific outputs are grounded in validated measurements. The SHAP-based XAI module substantially improves non-expert comprehension — a finding with direct implications for the democratization of ocean science communication to policymakers and educators.

The LSTM-Autoencoder achieves anomaly detection performance (F1: 0.90) that compares favorably with supervised anomaly detection approaches on Argo data reported in the literature, while operating entirely without labeled training anomalies. The TFT forecasting model outperforms established baseline methods by margins of 39–52% in RMSE across both tem-perature and salinity domains.

The collaborative workspace introduces a new dimension of scientific utility that has not previously been explored in the context of conversational oceanographic platforms. By enabling multiple researchers to simultaneously explore, anno-tate, and discuss ocean data within a shared AI-mediated en-vironment, FloatChat AI begins to address the fundamentally collaborative nature of climate science research.

The VLM integration, while achieving strong performance on standard chart types, highlights the need for further work on complex multi-panel scientific figures — an important direction for future development.

One limitation of the current implementation is that the TFT forecasting model is trained exclusively on Indian Ocean and global basin data and may require domain adaptation for deployment in other regional contexts such as the Arctic or Antarctic oceans, where seasonal dynamics differ substantially.

Similarly, the LSTM-Autoencoder anomaly threshold is cur-rently calibrated on a fixed percentile and may benefit from adaptive, region-specific thresholding in future iterations.

VI. FUTURE SCOPE

1) Federated Learning for Regional Domain Adaptation: Collaborative fine-tuning of the TFT and LSTM models across oceanographic institutes using federated learning frameworks, enabling regional dialect handling (e.g., Arc-tic vs. Tropical basins) without sharing raw training data.
2) Multi-Modal Ocean Data Integration: Extension of the data ingestion pipeline to incorporate satellite altimetry, glider networks, and mooring arrays through the Coper-nicus CMEMS unified API [13], enabling cross-source comparative analysis within FloatChat AI.
3) Edge AI Deployment: Packaging the NL2SQL and anomaly detection components as lightweight Docker containers deployable on research vessels with limited connectivity, eliminating cloud dependency for at-sea operations.
4) Graph Neural Networks for Float Trajectory Pre-diction: Integration of GNN-based trajectory prediction models to forecast the future geographic positions of active Argo floats based on historical drift patterns and ocean current models.
5) Visual Data Input Enhancement: Expansion of the VLM module to support multi-panel scientific figure interpretation and handwritten annotation recognition, im-proving extraction accuracy for complex oceanographic visualizations.
6) Natural Language Report Generation: Automated gen-eration of structured scientific reports summarizing multi-session research findings, suitable for direct inclusion in academic publications or policy briefs.

VII. CONCLUSION

This paper presented FloatChat AI, a next-generation mul-timodal predictive intelligence platform for conversational exploration and analysis of Argo oceanographic datasets. Building upon the foundational conversational AI paradigm for ocean data access, FloatChat AI introduces six novel and independently validated contributions: a QC-Guard Layer achieving 100% Argo quality-control compliance in SQL query execution; a SHAP-based Explainable AI module im-proving non-expert comprehension by 82.6% relative to a no-explanation baseline; a Vision-Language Model integra-tion enabling multimodal image-to-query interaction with 91.4% geographic coordinate extraction accuracy; an LSTM-Autoencoder anomaly detection service achieving an F1-score

International Journal for Research in Applied Science & Engineering Technology (IJRASET)
ISSN: 2321-9653; IC Value: 45.98; SJ Impact Factor: 7.538
Volume 14 Issue IV Apr 2026- Available at www.ijraset.com

TABLE II
SUMMARY OF FUNCTIONAL TESTING CASES — FLOATCHAT AI

Sr.	Feature Tested	Input	Expected & Actual Output	Status
1	QC-Guard Injection	Query temperature for float 590666 without specifying QC flag	SQL auto-injects WHERE value_qc IN (1,2); 847 suspect records excluded; UI badge displayed	Pass
2	XAI Reasoning Trace	Ask: Why is salinity anomalous in Bay of Bengal?	SHAP explanation rendered; top 3 contributing parameters listed; ChromaDB source cited	Pass
3	VLM Image-to-Query	Upload satellite SST map; ask Compare with float 1901302	LLaVA extracts region coordinates; SQL generated; Plotly overlay chart produced	Pass
4	Anomaly Alert	Ingest profile with salinity 42.1 PSU at 150m depth	LSTM-Autoencoder flags profile; push notification triggered; NL alert message generated	Pass
5	TFT Forecasting	Plot 30-day SST forecast for Indian Ocean	TFT model returns forecast with 95% confidence intervals; RMSE displayed on chart	Pass
6	Multi-User Workspace	Two users annotate the same temperature profile chart simultaneously	Both annotations appear in real-time; shared dashboard updated; no conflict	Pass
7	Conversational Memory	After Test 5, ask What was the highest forecasted temperature?	System retains float/region context; correct value returned without re-querying	Pass
8	Spatial Query	Show active floats in Arabian Sea	PostGIS executes spatial filter; Leaflet map renders correct markers with QC badges	Pass
9	RAG / Vector Retrieval	What is the expected battery life of a standard APEX float?	ChromaDB retrieves Argo manual context; SQL bypassed; accurate summary returned	Pass
10	Telemetry Logging	Submit any DB query and inspect backend logs	latencyMs, tokensUsed, SHAP computation time all logged without UI delay	Pass

of 0.90 with sub-five-minute alert latency; a Temporal Fusion Transformer forecasting engine outperforming Prophet and ARIMA baselines by 39–52% in RMSE; and a collaborative multi-user workspace enabling synchronized real-time annota-tion and data exploration for research teams.

The comprehensive evaluation framework — spanning func-tional testing, quantitative ML benchmarking, and a controlled user comprehension study — provides rigorous empirical sup-port for each claimed contribution. FloatChat AI demonstrates how the principled integration of explainability, multimodal-ity, proactive intelligence, and collaboration can transform a reactive conversational query tool into a comprehensive scientific intelligence platform. The system has the potential to significantly broaden participation in ocean and climate research by making complex oceanographic data accessible, interpretable, and actionable for a diverse community of users ranging from domain experts to policymakers and educators.

VIII. ACKNOWLEDGEMENT

The authors express their sincere gratitude to the Depart-ment of Computer Science and Engineering, SRM Institute of Science and Technology, Ramapuram Campus, for the support and resources provided for this project. The authors also thank Mrs. S. Chandrakala, Assistant Professor, for her guidance and valuable suggestions throughout the development of the FloatChat AI project. Special thanks to the International Argo Program for providing open-access oceanographic data used in this research.

REFERENCES

[1] P. Lewis et al., "Retrieval-Augmented Generation for Knowledge-Intensive NLP Tasks," Advances in Neural Information Processing Systems (NeurIPS), 2020. Available: https://arxiv.org/abs/2005.11401

[2] H. Touvron et al., "Llama: Open and Efficient Foundation Language Models," Meta AI, 2023. Available: https://arxiv.org/abs/2302.13971

[3] A. Q. Jiang et al., "Mistral 7B," Mistral AI, 2023. Available: https://arxiv.org/abs/2310.06825

[4] S. M. Lundberg and S. Lee, "A Unified Approach to Interpreting Model Predictions," Advances in Neural Information Processing Systems (NeurIPS), 2017. Available: https://arxiv.org/abs/1705.07874

[5] B. Lim et al., "Temporal Fusion Transformers for Interpretable Multi-horizon Time Series Forecasting," International Journal of Forecasting, vol. 37, no. 4, pp. 1748–1764, 2021. Available: https://arxiv.org/abs/1912.09363

[6] S. Hochreiter and J. Schmidhuber, "Long Short-Term Memory," Neural Computation, vol. 9, no. 8, pp. 1735–1780, 1997.

[7] H. Liu et al., "LLaVA: Visual Instruction Tuning," Advances in Neural Information Processing Systems (NeurIPS), 2023. Available: https:// arxiv.org/abs/2304.08485

[8] "Argo Data Management," International Argo Program / Argo Steering Team. Available: https://argo.ucsd.edu/data/
[9] "PostgreSQL: The World's Most Advanced Open Source Relational Database," The PostgreSQL Global Development Group. Available: https://www.postgresql.org/
[10] "PostGIS — Spatial and Geographic objects for PostgreSQL," PostGIS Project. Available: https://postgis.net/
[11] "Ollama — Get up and running with large language models locally," Ollama. Available: https://ollama.com
[12] F. Liu et al., "A Survey on Large Language Models for Code Genera-tion," arXiv, 2024. Available: https://arxiv.org/abs/2406.00515
[13] "Copernicus Marine Environment Monitoring Service (CMEMS)," European Union's Copernicus Programme. Available: https://marine. copernicus.eu/