

# Architecture Design Document

## Big Mart Store Sales Prediction

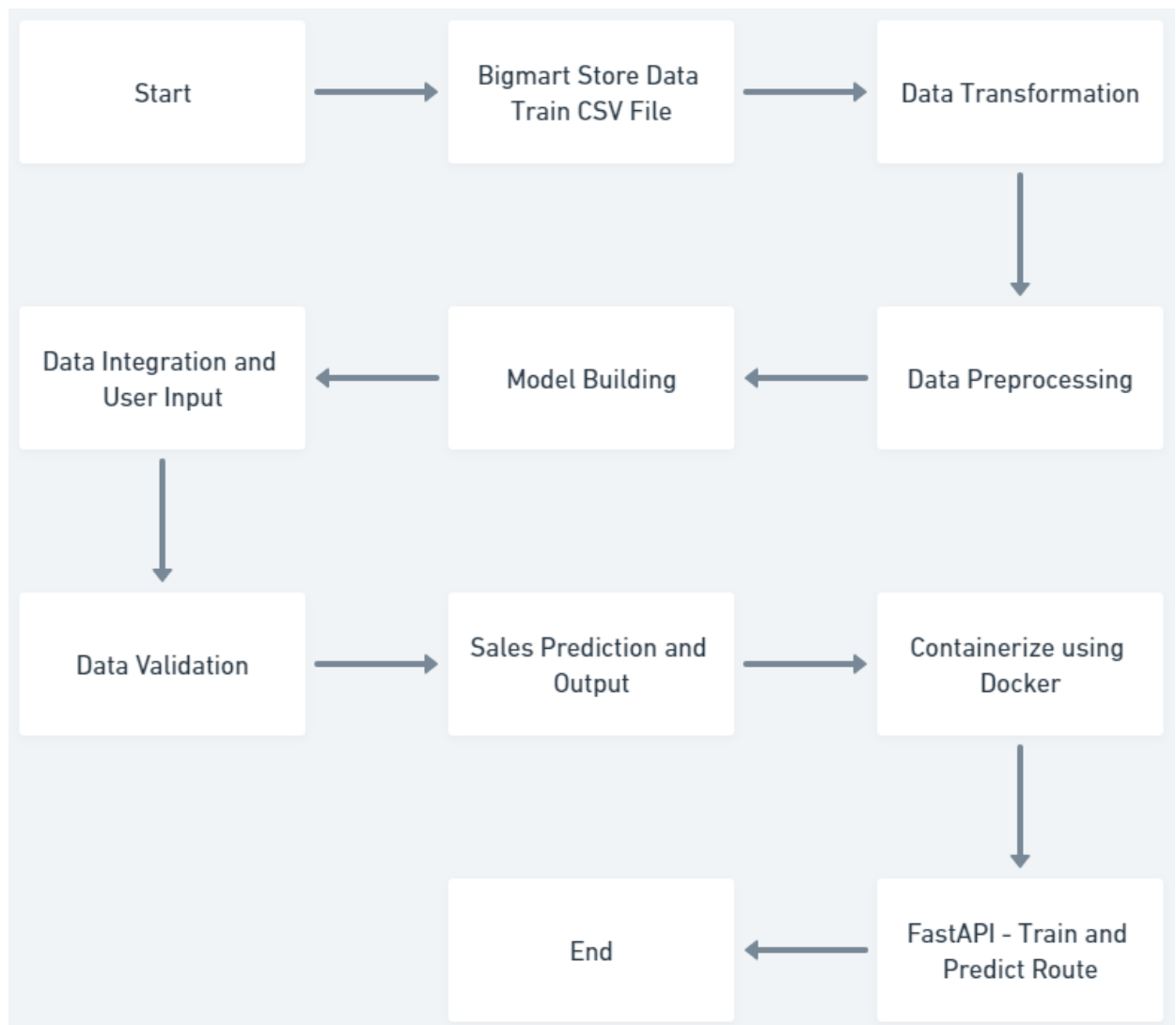
Written By	Vijaykumar Rewate
Document Version	0.1
Last Revised Date	25.5.2023

## 1. Introduction

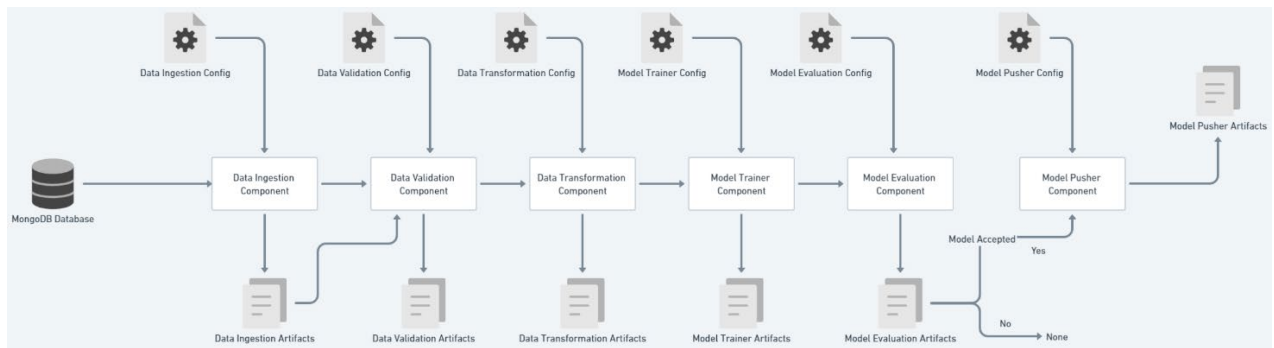
- Purpose: This document outlines the architecture design for the Big Mart Store Sales Prediction project, providing an overview of the system's structure, components, and interactions.
- Scope: The architecture design covers the high-level system architecture, key components, data flow, and integration points.

## 2. System Architecture

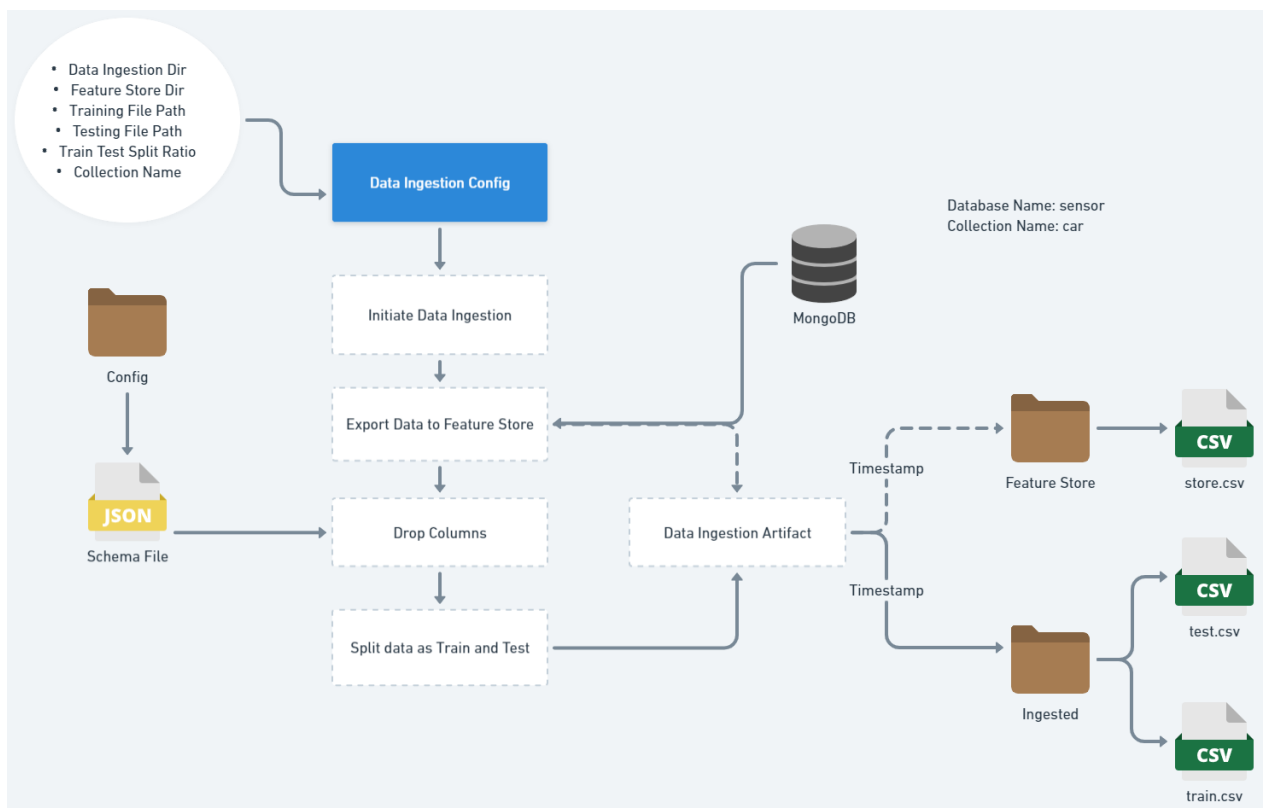
- Overview: The system architecture follows a client-server model, where the client interacts with the web-based user interface and the server handles the data processing, model training, and prediction.



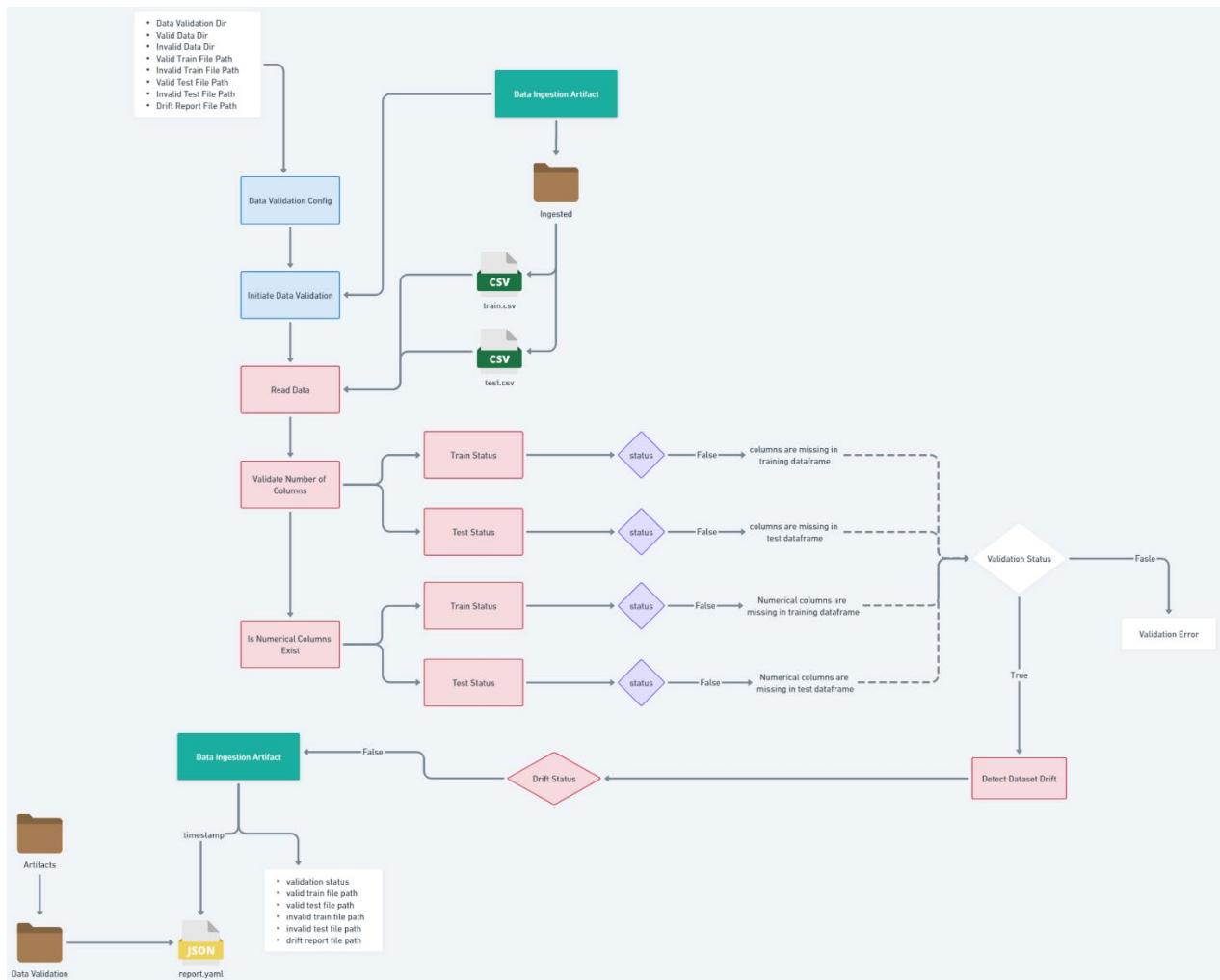
## High Level Code Flow - Pipeline



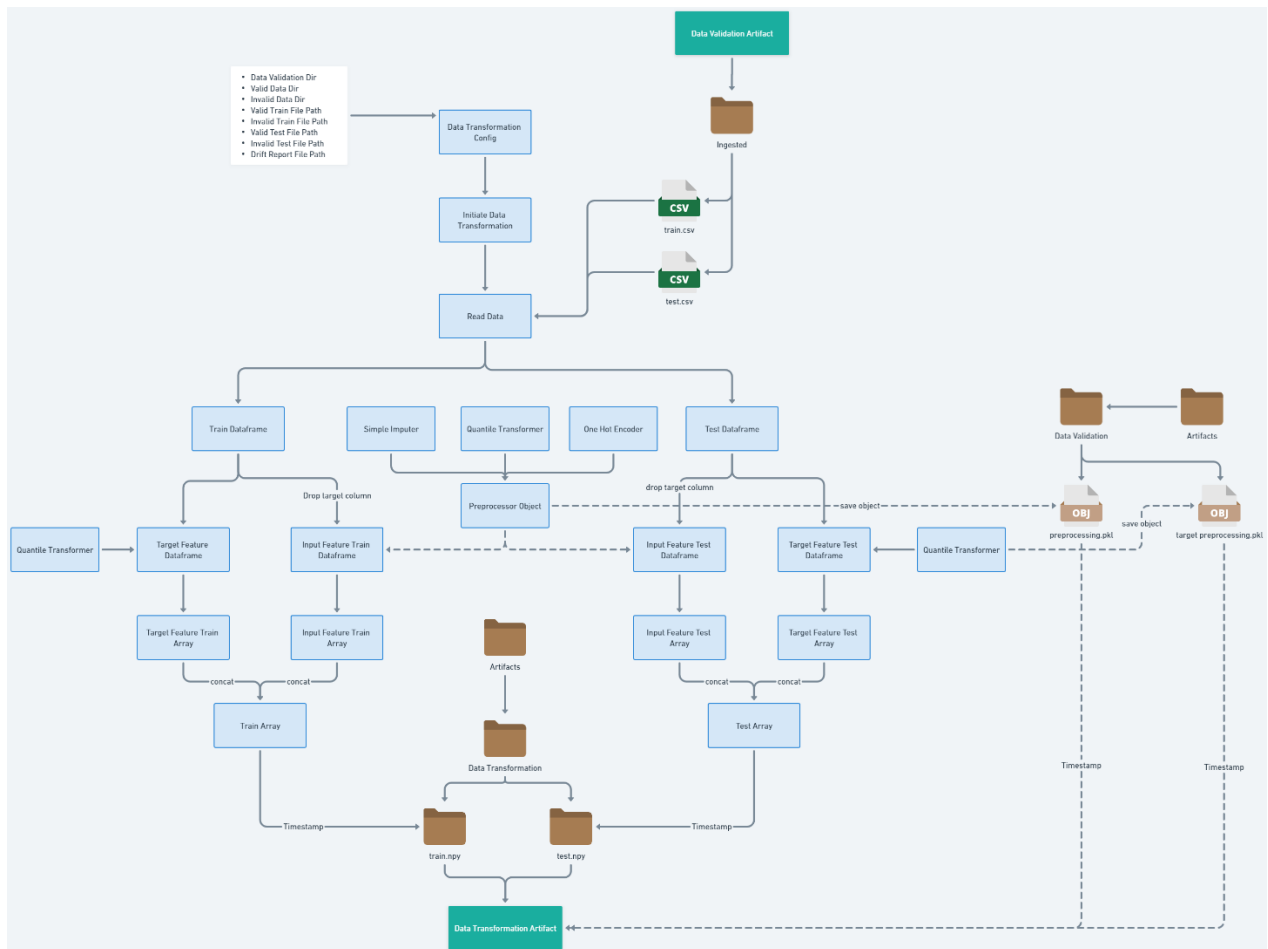
## Data Ingestion Component



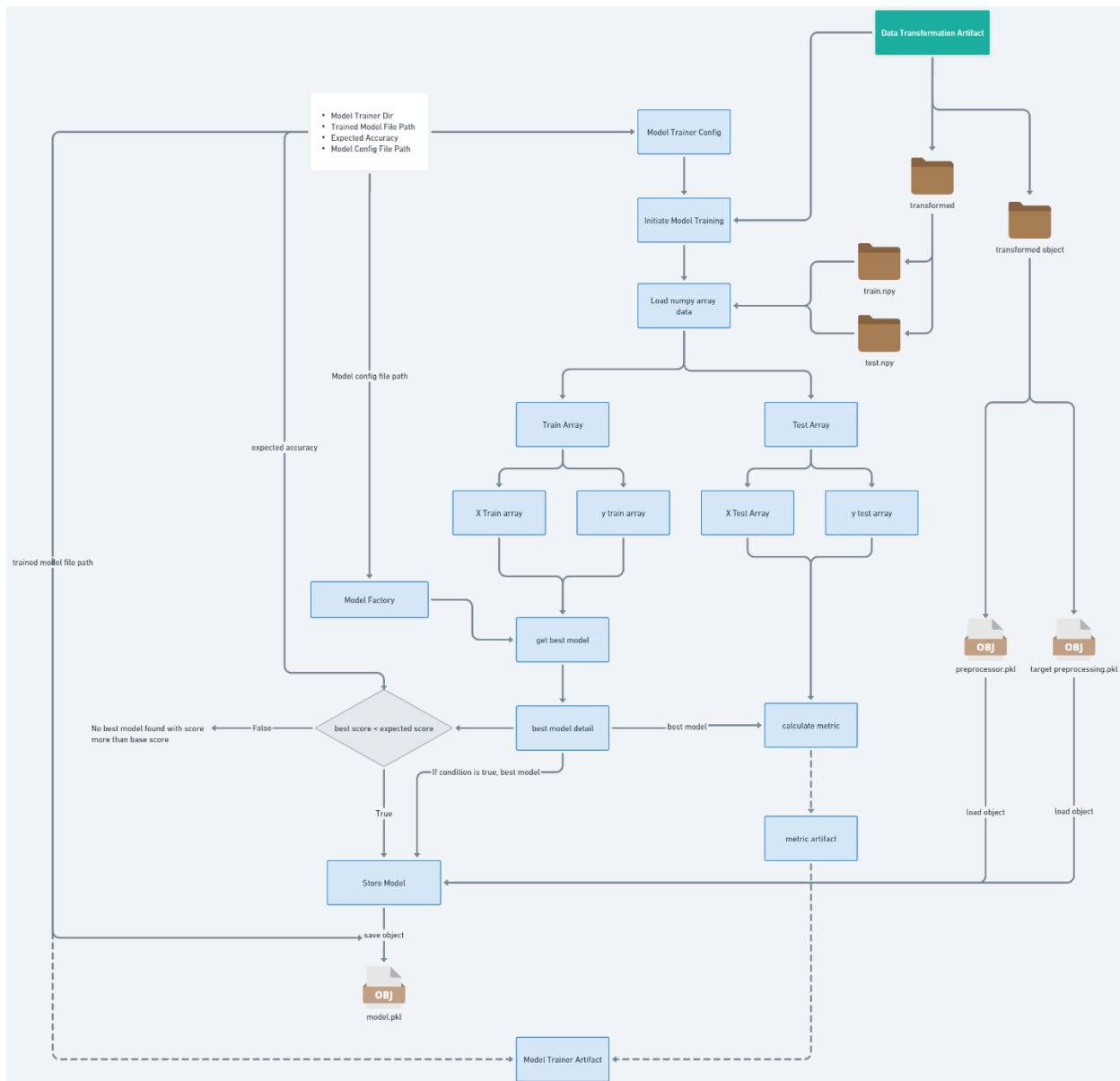
## Data Validation Component



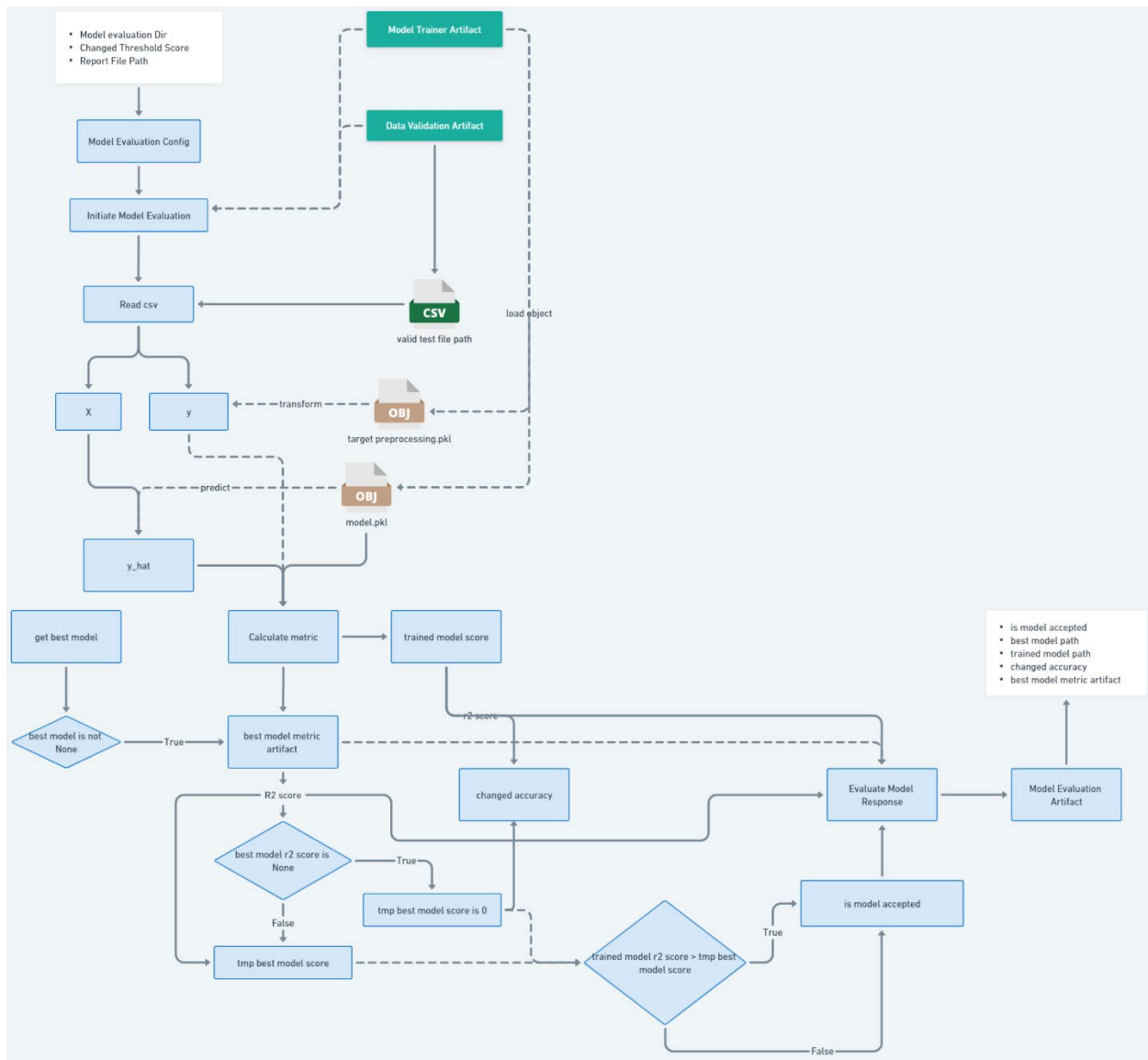
## Data Transformation Component



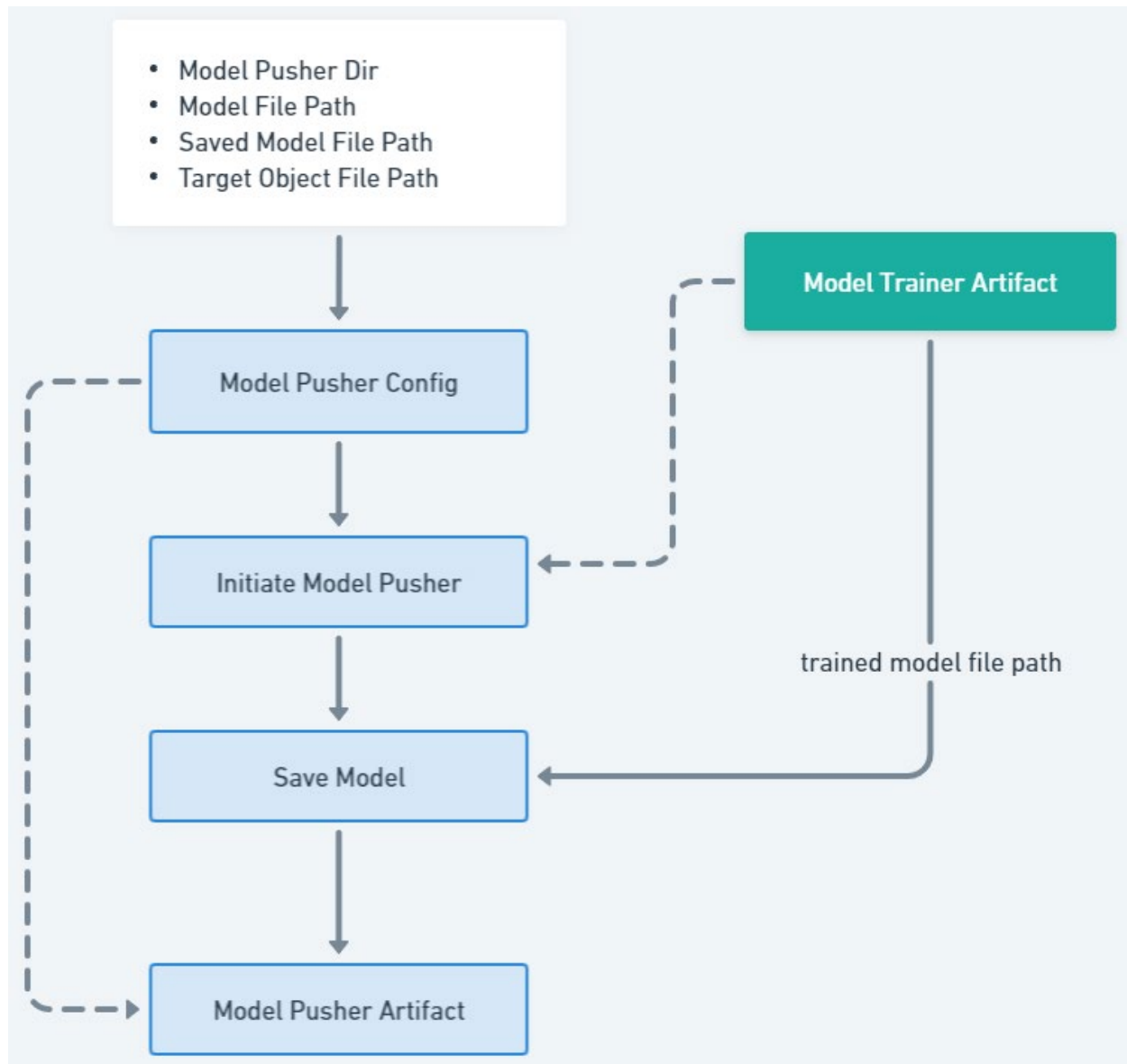
## Model Trainer Component



## Model Evaluation Component

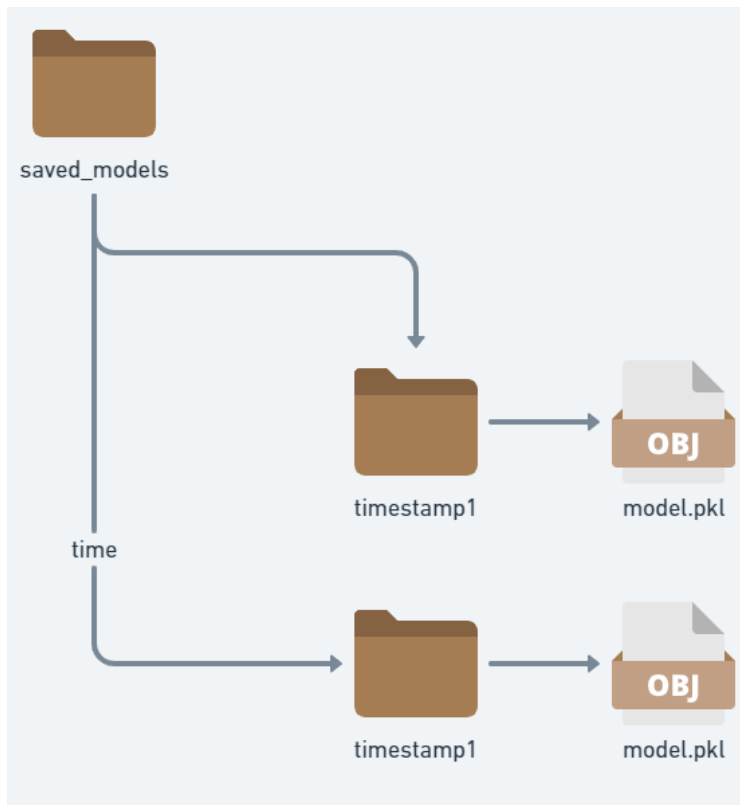


## Model Pusher Component





## Saved Models



- Components:
  - User Interface: Provides a user-friendly interface for inputting data and displaying the sales predictions.
  - Web Server: Handles user requests, communicates with the data processing components, and serves the user interface.
  - Data Processing Module: Responsible for data preprocessing, feature engineering, and encoding of categorical variables.
  - Model Training and Prediction Module: Trains the Gradient Boosting Regressor model on the preprocessed data and generates sales predictions.
  - Database: Stores the dataset, including historical sales data, product information, and outlet details.

### 3. Data Flow

- User Input: Users enter product details and outlet information through the user interface.
- User Interface to Web Server: The user interface sends the user inputs to the web server for processing.

- **Web Server to Data Processing Module:** The web server forwards the user inputs to the data processing module for preprocessing and feature engineering.
- **Data Processing Module to Model Training and Prediction Module:** The preprocessed data is passed to the model training and prediction module, which trains the Gradient Boosting Regressor model and generates sales predictions.
- **Model Training and Prediction Module to Web Server:** The sales predictions are sent back to the web server.
- **Web Server to User Interface:** The web server sends the sales predictions to the user interface for display to the user.

#### **4. Integration Points**

- **User Interface and Web Server:** The user interface interacts with the web server through RESTful API endpoints to send user inputs and receive sales predictions.
- **Web Server and Data Processing Module:** The web server communicates with the data processing module to send user inputs for preprocessing and feature engineering.
- **Data Processing Module and Model Training and Prediction Module:** The preprocessed data is passed from the data processing module to the model training and prediction module for training and prediction generation.
- **Model Training and Prediction Module and Web Server:** The sales predictions are sent back from the model training and prediction module to the web server.
- **Web Server and User Interface:** The web server sends the sales predictions to the user interface for display to the user.

#### **5. Deployment**

- **Deployment Environment:** The system is deployed using Docker containers to ensure portability and scalability.
- **Web Server Deployment:** The web server container is deployed on a server or cloud platform capable of hosting web applications.
- **Database Deployment:** The database, in this case, MongoDB, is deployed either locally or on a cloud-based database service.

#### **6. Performance Considerations**

- **Scalability:** The system should be designed to handle a large volume of data and concurrent user requests. The architecture should be scalable to accommodate increasing data size and user traffic.
- **Efficiency:** The data processing and model training modules should be optimized for performance to minimize processing time and resource utilization. Techniques like parallel processing and efficient algorithms should be employed.

- **Response Time:** The system should provide real-time or near real-time sales predictions to ensure a responsive user experience. Performance testing and optimization should be conducted to meet acceptable response time benchmarks.

## **7. Security**

- **Data Protection:** Measures should be taken to ensure the security and privacy of the data. This includes encryption of sensitive data, secure storage, and access control mechanisms to prevent unauthorized access.
- **User Authentication and Authorization:** The system should implement user authentication and role-based access control to ensure that only authorized users can access and interact with the system.
- **Secure Communication:** The communication between the user interface, web server, and other components should be encrypted using secure protocols like HTTPS to prevent data interception and tampering.

## **8. Maintenance and Support**

- **Monitoring:** The system should be equipped with monitoring tools to track system performance, detect anomalies, and generate alerts for proactive maintenance.
- **Logging and Auditing:** Detailed logging of system activities should be implemented to facilitate debugging, troubleshooting, and auditing of system usage.
- **Documentation:** Comprehensive documentation should be maintained, including system architecture, component specifications, deployment procedures, and user guides, to aid in system maintenance and future enhancements.
- **Version Control:** Source code and configuration files should be managed using a version control system to enable efficient collaboration, tracking of changes, and easy rollback if necessary.

## **9. Future Enhancements**

- **Integration with External Systems:** Explore the possibility of integrating the sales prediction system with external systems, such as inventory management or supply chain systems, to enable seamless data exchange and decision-making processes.
- **Advanced Machine Learning Models:** Investigate and incorporate advanced machine learning models, such as ensemble methods or deep learning algorithms, to further improve the accuracy and predictive capabilities of the sales prediction system.
- **Real-time Data Streaming:** Implement a data streaming pipeline to handle real-time data updates and provide up-to-date sales predictions based on the latest information.
- **Enhanced Visualization and Reporting:** Enhance the user interface to provide interactive visualizations, customizable dashboards, and comprehensive reports to help users analyze sales trends, identify patterns, and make data-driven decisions.

- **Integration with Customer Relationship Management (CRM) Systems:** Integrate the sales prediction system with CRM systems to leverage customer data and improve sales forecasting accuracy by considering customer behavior and preferences.
- **Automated Model Retraining:** Implement a mechanism to automatically retrain the sales prediction model periodically or when significant changes occur in the data, ensuring the model remains up to date and maintains its predictive performance.

## **10. Conclusion**

- The proposed architecture design for the Big Mart Store Sales Prediction project provides a solid foundation for accurate sales predictions and informed decision-making. The architecture design provides a scalable, modular, and efficient solution for the Big Mart Store Sales Prediction project. With the inclusion of future enhancements, the system can be further improved to meet evolving business needs and industry demands.
- By leveraging advanced machine learning techniques, real-time data processing, and integration with external systems, the sales prediction system can deliver actionable insights, optimize inventory management, and drive revenue growth for Big Mart.
- It enables seamless data flow, integration between components, and a user-friendly interface. The deployment using Docker ensures easy setup and deployment in different environments. The architectural design, combined with a strategic roadmap for future enhancements, ensures that the system remains adaptable, scalable, and aligned with Big Mart's long-term objectives.