Symbiosis Skills and Professional University

Kiwale, Pune

PROJECT REPORT

On

"Food Delivery Time Prediction"

Submitted by

1. Manisha Shinde
2. Deepali Ghumare
3. Pallavi Kale
4. Akash Satapute
5. Vaibhav Hegadkar

DA- Batch FST02:DA03

Under The Guidance of

Trainer's Name: 1) Ritviz Singh

2) Shiv Patel

STUDENT DECLARATION AND ATTESTATION BY TRAINER

I'm Deepali Ghumare, Manisha Shinde, Pallavi Kale, Akash Satapute, Vaibhav Hegadkar, A student at **Symbiosis Skills and Professional University,** hereby declare and attest that we understand and agree to comply with the rules and regulation pertaining to the Food Delivery Time Prediction implemented by the intuition.

Name of Students: -

1) Manisha Shinde
2) Deepali Ghumare
3) Pallavi Kale
4) Akash Satapute
5) Vaibhav Hegadkar

Course: Data Associate

CERTIFICATE

This is to certify that the report entitled, **"Food Delivery Time Prediction**" submitted by

"**1) Manisha Shinde, 2) Deepali Ghumare, 3) Pallavi Kale, 4) Akash Satapute 5) Vaibhav Hegadkar**" to Symbiosis Skills and Professional University, Pune, Maharashtra, India, is a record of bonafide Project work carried out by him under my supervision and guidance and is worthy of consideration for the completion of certificate course in ''Data Associate".

Signature of Trainer Name of Trainer Date: / / 2024

Supervisor
Date:

Supervisor

ACKNOWLEDGEMENTS

We express our deep gratitude to everyone who played a pivotal role in the successful development of the **"Food Delivery Time Prediction"** by Team **Manisha Shinde**, **Deepali Ghumare**, **Pallavi Kale, Akash Satpute** and **Vaibhav Hegadkar**, with the unwavering support and guidance of Ritviz Singh and Shiv Patel at Symbiosis Centre of Distance Learning.

Our team members demonstrated exceptional dedication, contributing a rich array of skills that culminated in the creation of an innovative system seamlessly blending technology with real-world application.

A sincere thank you is extended to our esteemed project guides, **Ritviz Singh** and **Shiv Patel**, for their invaluable mentorship and technical expertise. Their guidance not only shaped the project's technical dimensions but also ensured its alignment with the esteemed academic standards set by Symbiosis Centre of Distance Learning.

Team -

1) Manisha Shinde
2) Deepali Ghumare
3) Pallavi Kale
4) Akash Satapute
5) Vaibhav Hegadkar

INDEX

Sr. No	TITLE	Page No
1	PLAN OF CAPSTONE PROJECT	1 to 2
2	OBJECTIVE OF THE PROJECT	3 to 4
3	INTRODUCTION	5 to 6
4	FUTURE PLANS	7 to 8
5	LEARNING FROM THE PROJECT	9 to 10
6	SUGGESTIONS	11
7	ANNEXURE	12 to 15
8	WEEKLY REPORT	16
9	CONCLUSION	17

1. PLAN OF CAPSTONE PROJECT

1.1) Purpose of Project: -

- Accuracy Improvement:

  Enhance the accuracy of delivery time predictions by incorporating advanced machine learning algorithms and real-time data analysis.

- Time-Saving Solution:

  Provide a time-saving solution for both customers and delivery personnel by optimizing route planning, reducing delays, and streamlining the overall delivery process.

- User-Friendly Implementation:

  Emphasize the creation of a user-friendly interface for widespread adoption, ensuring ease of use for both customers and delivery personnel within the food delivery ecosystem.

- Real-time Monitoring:

  Implement real-time monitoring capabilities, allowing stakeholders to track delivery progress instantly. This feature is crucial for identifying and addressing issues promptly.

- Data Analysis and Reporting:

  Incorporate data analysis and reporting functionalities to generate comprehensive reports on delivery patterns. This data-driven approach will assist in making informed decisions regarding resource allocation and performance optimization.

- Improved Security:

  Enhance the security of the delivery system by leveraging advanced authentication mechanisms. This includes ensuring the integrity of order data and protecting against unauthorized access.

1.2) Period of Project: -

For a small to medium-scale "Food Delivery Time Prediction" project, the estimated duration is generally within the range of 3 to 4 months. The timeline is influenced by factors such as project complexity, team size, available resources, and the specific features and functionalities included in the system. Efficient planning and execution of phases like requirements gathering, design, development, testing, and implementation are key to staying within this timeframe.

1.3) Problem Statement Detailing: -

i)Uncertain Delivery Times:

Customers often experience uncertainty regarding the exact arrival time of their food orders. Inaccurate delivery time estimates can result in frustration and a diminished user experience.

ii)Operational Inefficiencies:

Restaurants and delivery service providers face challenges in optimizing their delivery operations. Inefficient route planning, unpredictable traffic conditions, and varying order volumes contribute to delays and inconsistencies.

iii)Impact of External Factors:

External factors such as weather conditions, road traffic density, and special events (e.g., festivals) play a significant role in influencing delivery times. The current systems often struggle to adapt to these dynamic variables.

2. OBJECTIVE OF THE PROJECT

2.1) Core Objectives:

- **Automate Delivery Time Estimation:**
  Replace manual estimation processes with a robust, data-driven system for predicting delivery times accurately.

- **Enhance Accuracy and Reliability:**
  Develop algorithms that ensure precise estimation under varying conditions, accounting for factors like traffic, weather, and order volume.

- **Optimize Delivery Operations:**
  Streamline delivery operations by optimizing routes, managing workloads, and improving overall logistics for increased efficiency.

- **Ensure Security and Authentication:**
  Implement secure authentication mechanisms to ensure that only authorized individuals access delivery data and services.

- **Cost and Resource Efficiency:**
  Minimize manual efforts associated with delivery management, reduce labor costs, and streamline administrative processes related to order fulfillment.

2) General Objective:

- Automate Delivery Time Prediction.
- Increase Accuracy and Security.
- Save Time and Reduce Administrative Burden.
- Analyze Delivery Data for Insights.
- Optional Integration and Exploration.

3. INTRODUCTION

3.1) Key Components:

1. Python:

- **Purpose:** Utilized for its versatility in handling data processing and algorithm implementation.
- **Role:** Serves as the primary programming language for developing the system, leveraging libraries and frameworks for data manipulation and machine learning specific to food delivery.
-

2. CSV (Comma-Separated Values):

- **Purpose:** Used as a structured file format for storing and managing food delivery records efficiently.
- **Role:** Acts as a data storage solution, allowing easy organization and retrieval of information related to food delivery orders, locations, and other relevant details.

3. Machine Learning Algorithms

- **Purpose:** Explored for predictive modeling, focusing on accurate food delivery time estimation under varying conditions.
- **Role***:* Employs machine learning algorithms to analyze historical data from food deliveries, considering factors such as distance, traffic, order preparation time, and other variables influencing delivery times.

4. Real-time Data Integration:

- **Purpose:** Integrated to provide up-to-the-minute information on food delivery status and adapt predictions dynamically.
- **Role:** Enables the system to continuously receive and process real-time data, such as traffic updates, weather conditions, and current order preparation times. This ensures that the system can dynamically adjust delivery time predictions based on the latest information for a more accurate estimation.

3.2) Project Objectives:

4. **Accurate Time Estimation:** Develop a system that accurately predicts the delivery time for food orders.
5. Ensure the system can adapt and learn from past experiences, improving prediction accuracy over time
6. Enable the system to dynamically adjust predictions based on current conditions, improving the accuracy of delivery time estimates.
7. Improve the accuracy of predictions by incorporating data from various stakeholders involved in the delivery process.

7.1) Project Structure:

1. **Real-time Monitoring:**

•Enable real-time monitoring of delivery progress for stakeholders to track orders instantly.

2. **Data Analysis and Reporting:**

•Implement data analysis and reporting functionalities to generate insights into delivery patterns and trends**.**

Security Measures:

•Integrate advanced security measures to protect against unauthorized access and ensure data integrity.

3.4) Benefits:

- **Accuracy**: Utilize advanced algorithms to enhance the accuracy of delivery time predictions.

- **Efficiency**: Streamline delivery operations, reducing delays and optimizing resource utilization for increased efficiency.

- **Security**: Implement robust security measures to protect against unauthorized access and ensure the integrity of delivery data.

- **Convenience**: Provide a convenient and automated solution for both customers delivery person.

- **Data Management**: Stores attendance records securely and enables easy analysis.

- **Cost-Effective**: Leverages open-source libraries and technologies.

4.FUTURE PLANS

1.Performance Optimization:

- Optimize processing speed for real-time delivery time predictions, ensuring efficient system operation.

2.Enhanced Features:

- Explore features such as multi-camera support, advanced analytics, integration with existing systems, and cloud deployment for scalability.

3.Security and Privacy:

- Implement additional security measures to protect sensitive delivery data and ensure user privacy

4.User Experience:

- Develop a mobile app for customers and delivery personnel, enhancing ,accessibility and convenience.
- Accessibility Features: Incorporate accessibility features for users with different abilities.

5.Hardware Optimization:

- Investigate the use of specialized hardware to accelerate processing and optimize performance.

6.Continuous Improvement:

- Stay updated with advancements in delivery prediction technology, incorporating improvements and gathering user feedback for further enhancements.

5.LEARNING FROM THE PROJECT

•Technical Learnings:

1. Python Libraries:

❖Gained proficiency in Python, with a focus on libraries crucial for food delivery time
prediction, such as NumPy for numerical computations, Pandas for data manipulation, and Scikit-learn for machine learning model implementation.

❖Developed practical experience in integrating these libraries for real-world applications.

2. Integration Skills:

❖Developed practical experience in integrating various Python libraries to create a comprehensive and efficient food delivery time prediction system.

3. Data Processing and Feature Engineering:

❖**Data Handling Techniques:** Mastered data handling techniques, especially in the context of food delivery data. This includes preprocessing steps like cleaning, normalization, and feature scaling to prepare data for model training.

❖**Feature Selection:** Explored feature engineering methods to identify and select relevant features affecting delivery times, such as historical order data, traffic conditions, and distance metrics.

•Project Management:

1. Planning and Execution:

❖Enhanced skills in breaking down a project into manageable tasks and setting realistic timelines.

❖Practiced problem-solving and debugging techniques to overcome challenges encountered during development.

2. Testing and Evaluation:

❖Conducted comprehensive testing to assess system accuracy and efficiency.

❖Gathered user feedback to identify areas for improvement.

•Algorithm Understanding:

1. Linear Regression:

❖ **Purpose:**

- Linear Regression is a simple and interpretable algorithm used for modeling the relationship between the dependent variable (delivery time) and one or more independent variables (features like distance, traffic, etc.).

❖ **Key Features:**

- Assumes a linear relationship between features and the target variable.
- Provides coefficients that indicate the impact of each feature on the predicted outcome.

❖ **Considerations:**

- Suitable when there's a linear correlation between features and delivery time.
- Sensitive to outliers

2. Decision Tree Regressor:

❖**Purpose:**

•Decision Tree Regressor is a non-linear model that makes decisions based on a set of rules, splitting the data into subsets and predicting the average of each subset.

❖**Key Features:**

- Can capture complex non-linear relationships.
- Prone to overfitting, but this can be controlled through hyperparameter tuning.

❖**Considerations:**

•Suitable when the relationship between features and delivery time is non-linear or involves interactions.

3. Random Forest Regressor:

❖**Purpose:**

- Random Forest is an ensemble method that builds multiple decision trees and averages their predictions, providing better generalization and reducing overfitting.

❖**Key Features:**

- Combines the strength of multiple decision trees.
- Robust against overfitting compared to a single decision tree**.**

❖**Considerations:**

- Suitable when there are multiple features with varying importance.

4. Random Forest Regressor:

❖**Purpose:**

- XGBoost (Extreme Gradient Boosting) is a powerful and efficient gradient boosting algorithm that builds an ensemble of weak learners, typically decision trees.

❖ **Key Features:**

- Handles missing data well.
- Regularization to control overfitting.

❖ **Considerations:**

- Suitable for large datasets and achieving high predictive performance.

- **Specific Learning Outcomes:**

1. **Data Collection and Preprocessing:**

❖ Learned the importance of high-quality training data for accurate face recognition.

❖ Applied techniques for image resizing, cropping, and grayscale conversion.

2. **Model Training:**

❖ Understood the process of training a face recognition model using labeled images.

❖ Experimented with different parameters to optimize model performance.

3. **Attendance Marking:**

❖ Implemented logic to record attendance based on recognized faces.

❖ Used CSV files for data storage and retrieval.

4. **User Interface Design:**

❖ Created a user-friendly interface for registration, attendance tracking, and system management.

❖ Incorporated real-time feedback and guidance for users.

6. SUGGESTIONS

❖Multi-factor Predictors:

❖**Weather Integration:** Include real-time weather data to account for weather-related delays.

❖**Historical Order Patterns:** Leverage historical order data to identify patterns and predict delivery times more accurately.

❖Dynamic Order Adjustments:

❖**Order Volume Prediction:** Implement algorithms to predict peak order times and dynamically adjust predictions based on expected order volumes.

❖Route Customization:

❖**Customer Preferences:** Allow customers to specify preferences such as preferred routes or delivery time windows, influencing the prediction model.

❖Personalized Notifications:

❖**Customized Alerts:** Provide personalized notifications based on individual customer preferences, like preferred notification channels (SMS, app notifications, email).

❖Customer Interaction:

❖**Live Chat Support:** Integrate a live chat support system within the delivery tracking interface for real-time customer assistance.

❖**Order Modification:** Enable customers to modify their orders during transit and dynamically update delivery predictions.

❖Sustainability Metrics:

❖**Carbon Footprint Estimates:** Integrate features to estimate and communicate the environmental impact of delivery routes, contributing to sustainability initiatives.

❖Customer Feedback Loop:

❖**Feedback Integration:** Build a mechanism for customers to provide feedback on delivery times, helping improve the prediction model.

❖Adaptive Learning:

❖**Customer Behavior Analysis:** Utilize machine learning to analyze customer behavior and preferences, adapting the prediction model over time for individual users.

❖**Integration with existing systems:** Integrate the system with existing student databases or access control systems for seamless data exchange and automation.

❖**Performance optimization:** Optimize the system for real-time performance, especially face detection and

❖ **Real-Time Payment Status:** Integrate with payment systems to consider the processing time for payments, providing more accurate delivery estimates.

❖ Improve User Experience:

1.Customer-Facing App:

- **Order Tracking**: Develop a mobile app for customers to track their orders in real-time, including an estimated time of arrival (ETA).
- **Push Notifications**: Implement push notifications to keep customers informed about order status changes2

2.Delivery Personnel App:

- **Route Guidance:** Create a mobile app for delivery personnel with route optimization features, helping them navigate efficiently.
- Communication Channel: Include a communication channel between customers and delivery personnel for order-specific inquiries.

3.User-Friendly Interfaces:

- Intuitive Dashboard: Design a user-friendly dashboard for administrators to monitor overall system performance and make adjustments if needed.

4.Performance Metrics and Analytics:

- Analytics Dashboard: Develop an analytics dashboard for administrators to track performance metrics, such as prediction accuracy, delivery time variances, and customer satisfaction.

5.Customer Loyalty Programs:

- Rewards Integration: Implement features that allow customers to earn rewards or discounts for frequent orders or providing feedback.

6.Voice Assistants Integration:

- Voice-Activated Order Tracking: Explore voice-activated options for customers to check order status using virtual assistants.

7.Gamification Elements:

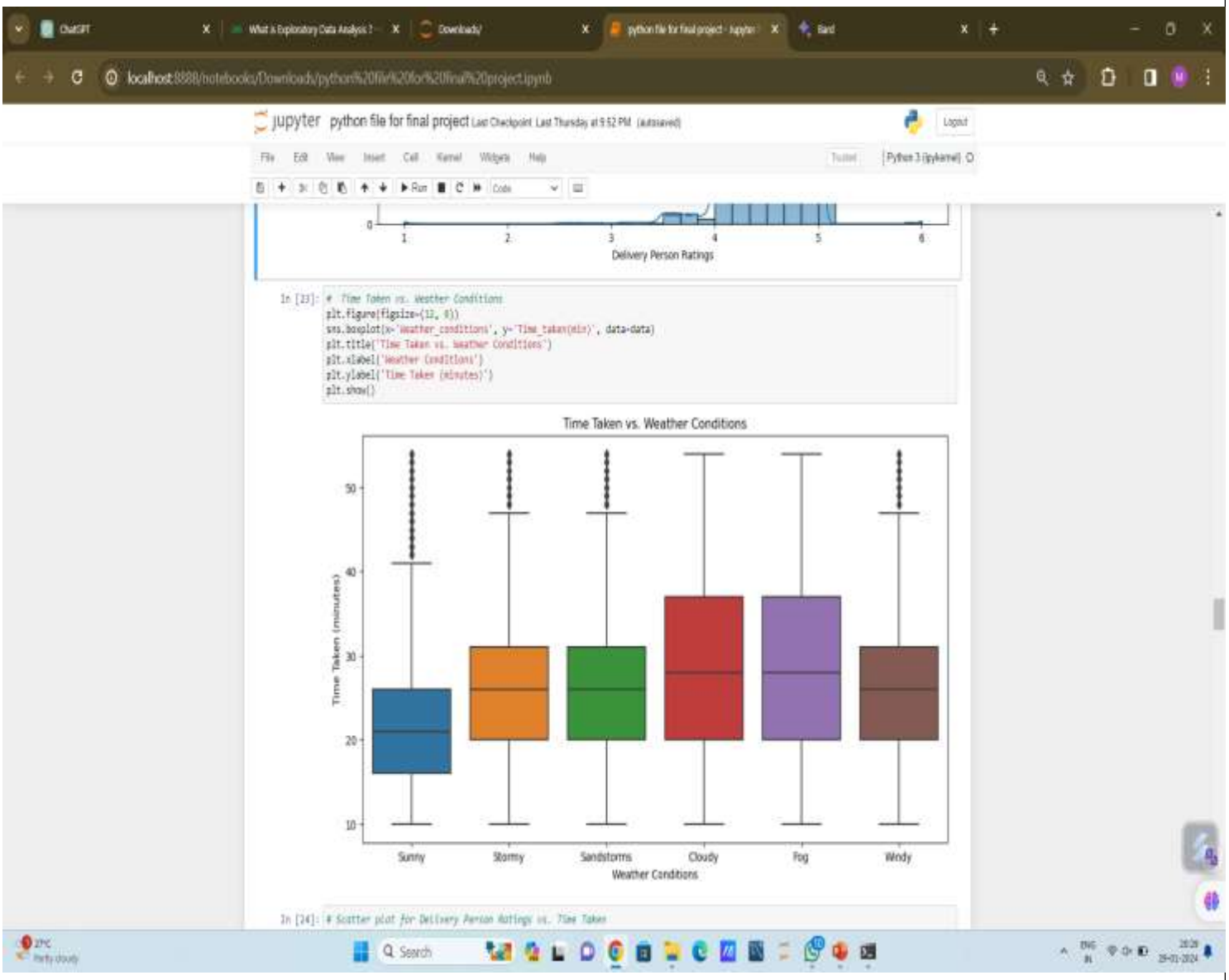
- Progress Bars and Achievements: Integrate gamification elements into the user interfaces to make the experience engaging, such as progress bars indicating order preparation stages.

8.Educational Content:

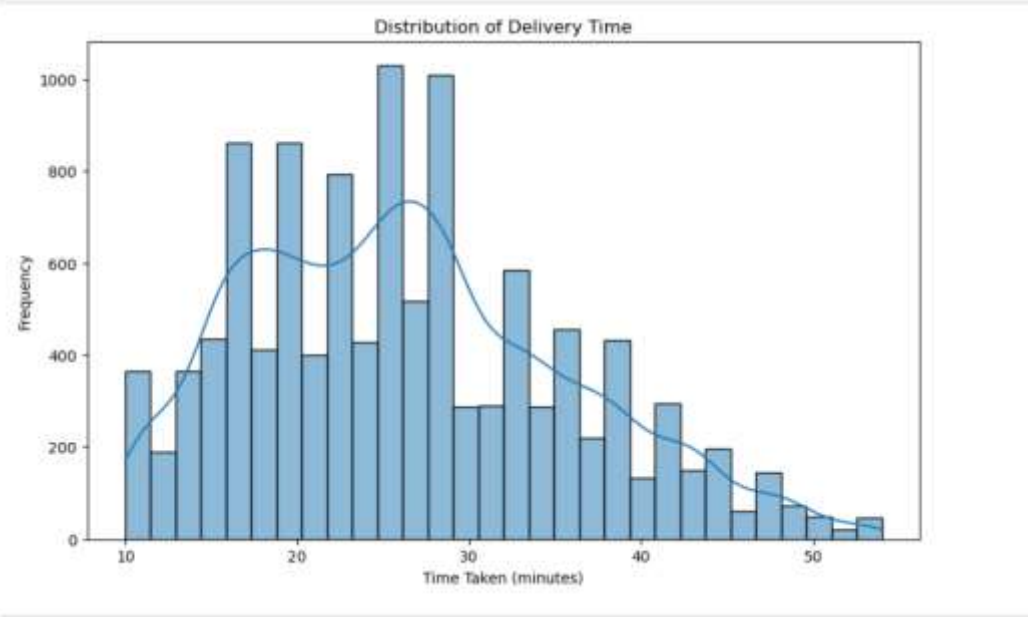
- Tips and Recommendations: Include educational content within the app, offering tips on how customers can optimize their orders for faster delivery

7. ANNEXURE

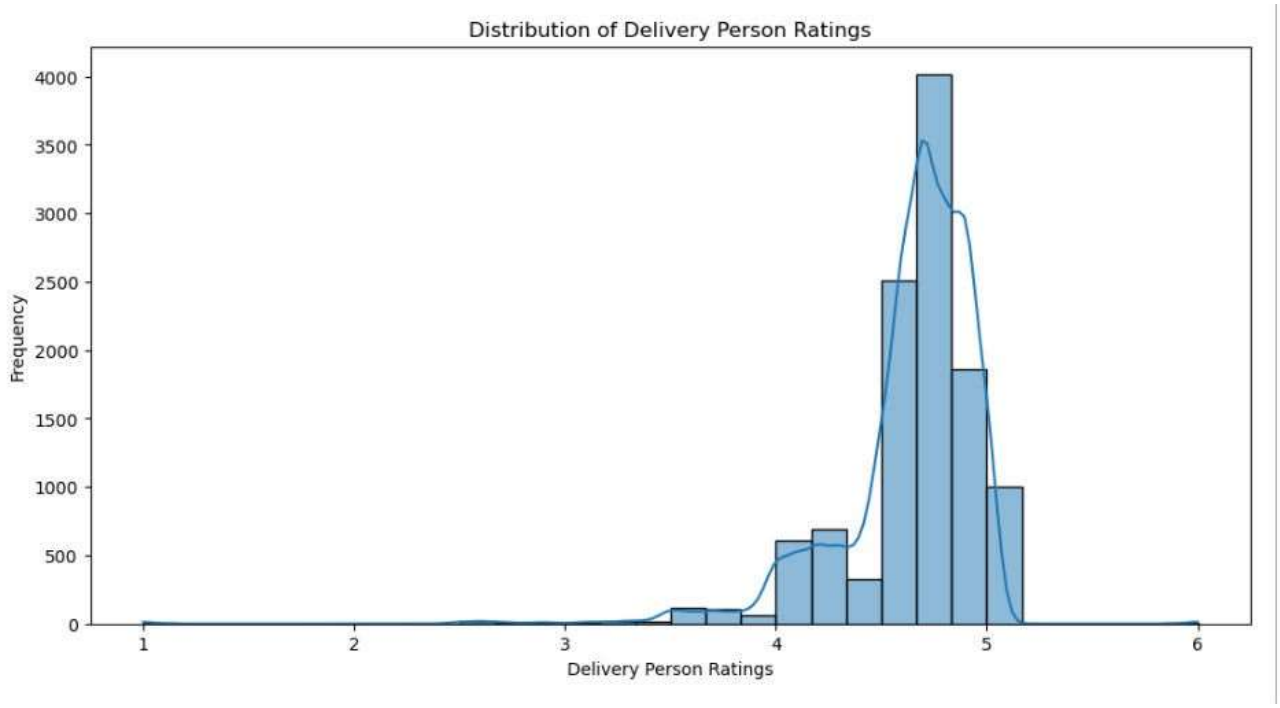
1) Weather Condition vs Time Taken



2) Delivery Time Distribution

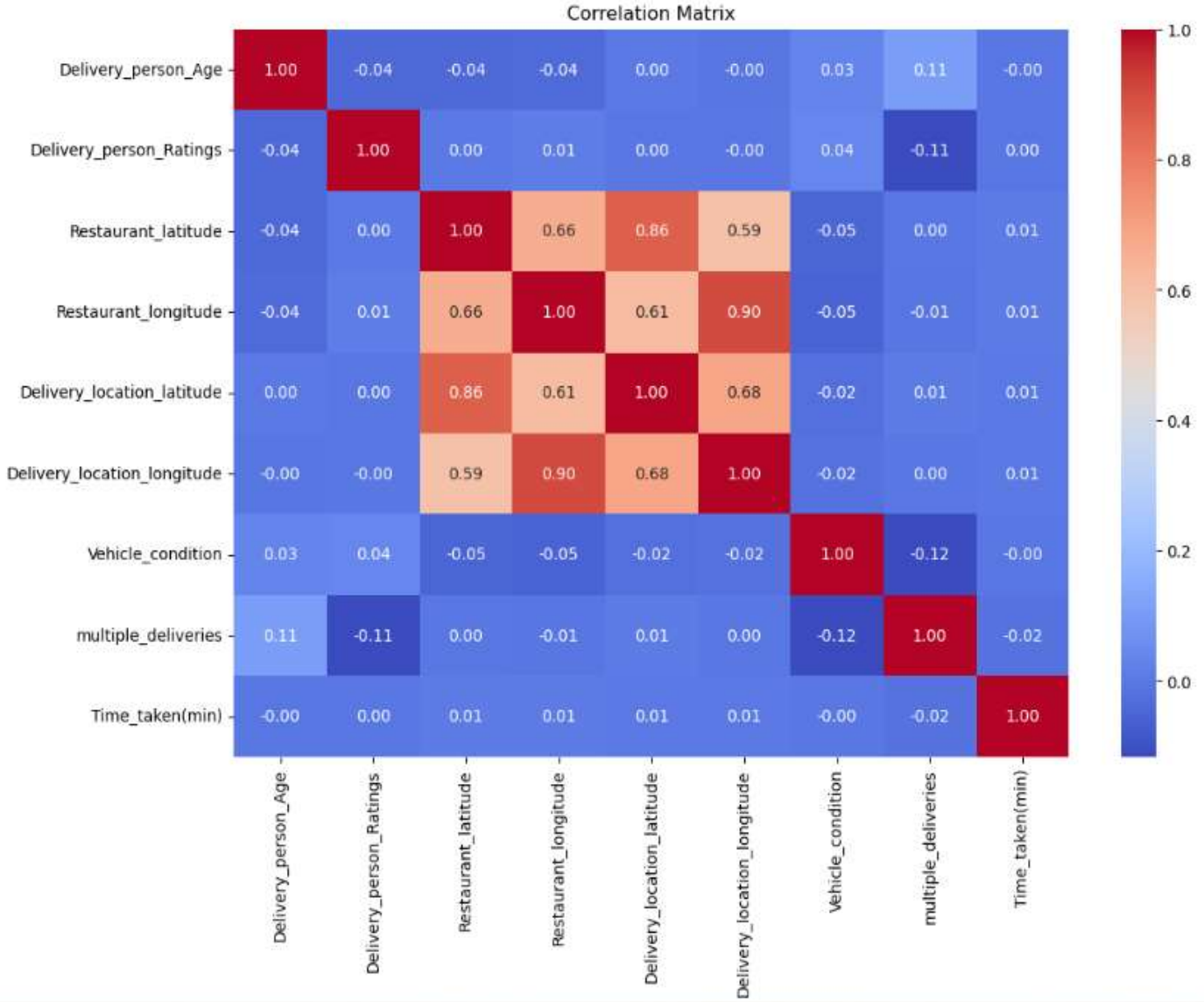


3) Delivery Persons Ratings



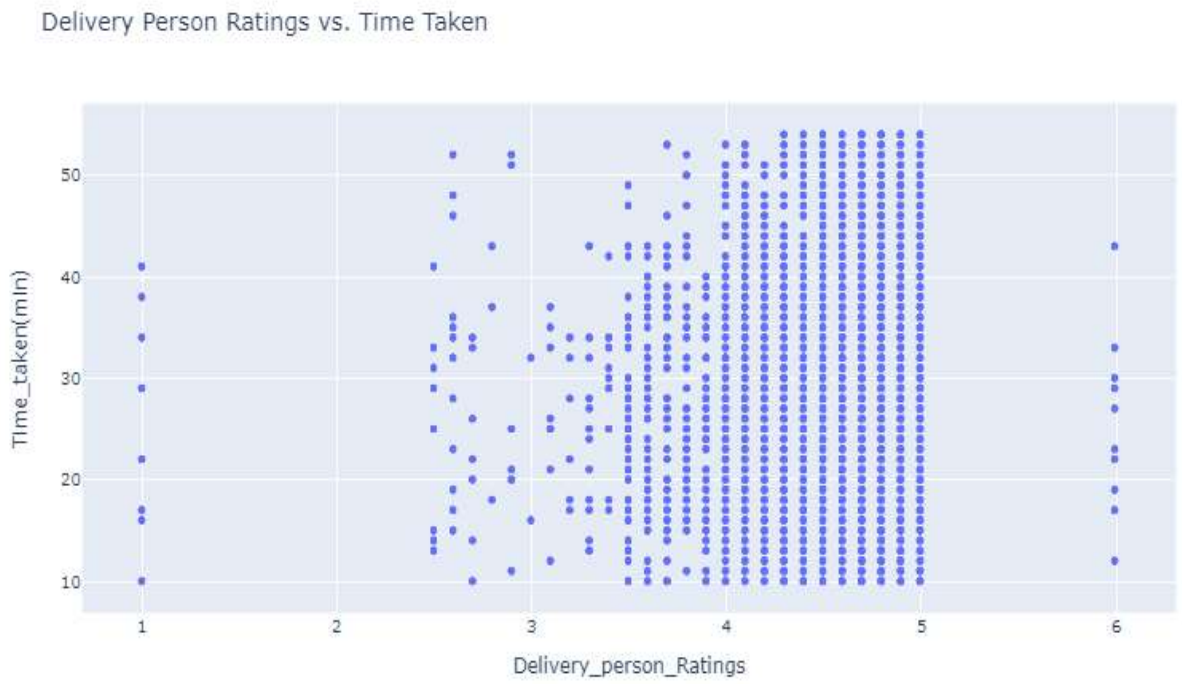
4) Heatmap

```
In [23]: # Correlation matrix
correlation_matrix = food_delivery_data.corr()
plt.figure(figsize=(12, 8))
sns.heatmap(correlation_matrix, annot=True, cmap='coolwarm', fmt=".2f")
plt.title('Correlation Matrix')
plt.show()
```



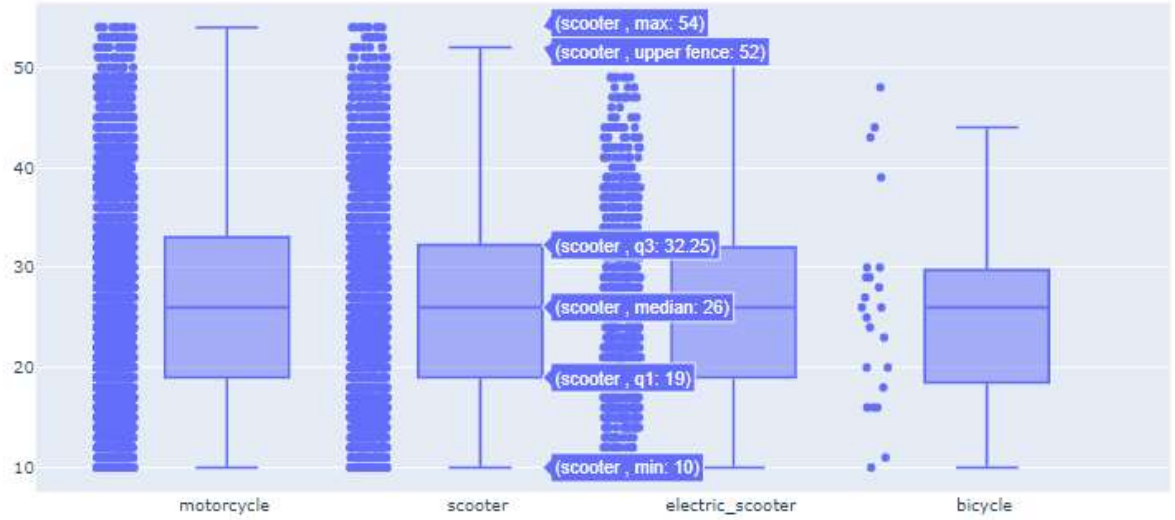
6) Graph

In [30]:
```
# Histogram for Delivery Person Age Distribution
fig2 = px.scatter(food_delivery_data, x='Delivery_person_Ratings', y='Time_taken(min)',
                  title='Delivery Person Ratings vs. Time Taken')
fig2.show()
```



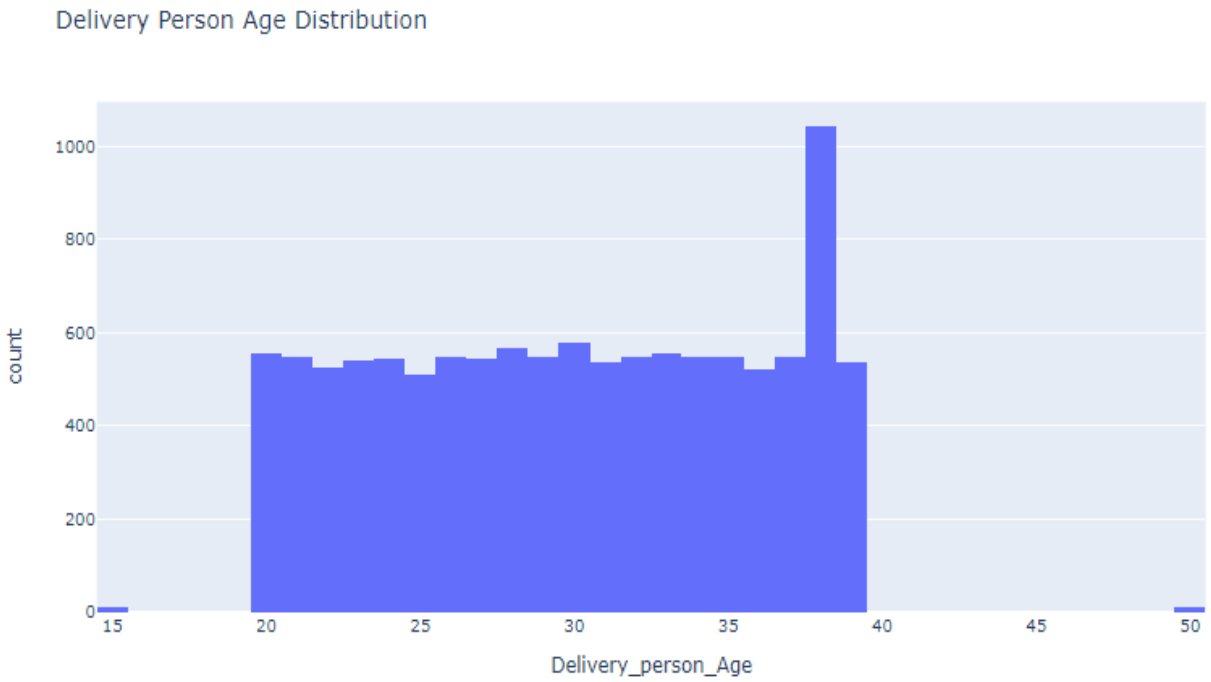
7) Graph

In [32]:
```
# Box plot for Time Taken by Vehicle Type
fig4 = go.Figure()
fig4.add_trace(go.Box(x=food_delivery_data['Type_of_vehicle'], y=food_delivery_data['Time_taken(min)'],
                      boxpoints='all', jitter=0.3, pointpos=-1.8, name='Time Taken by Vehicle Type'))
fig4.show()
```



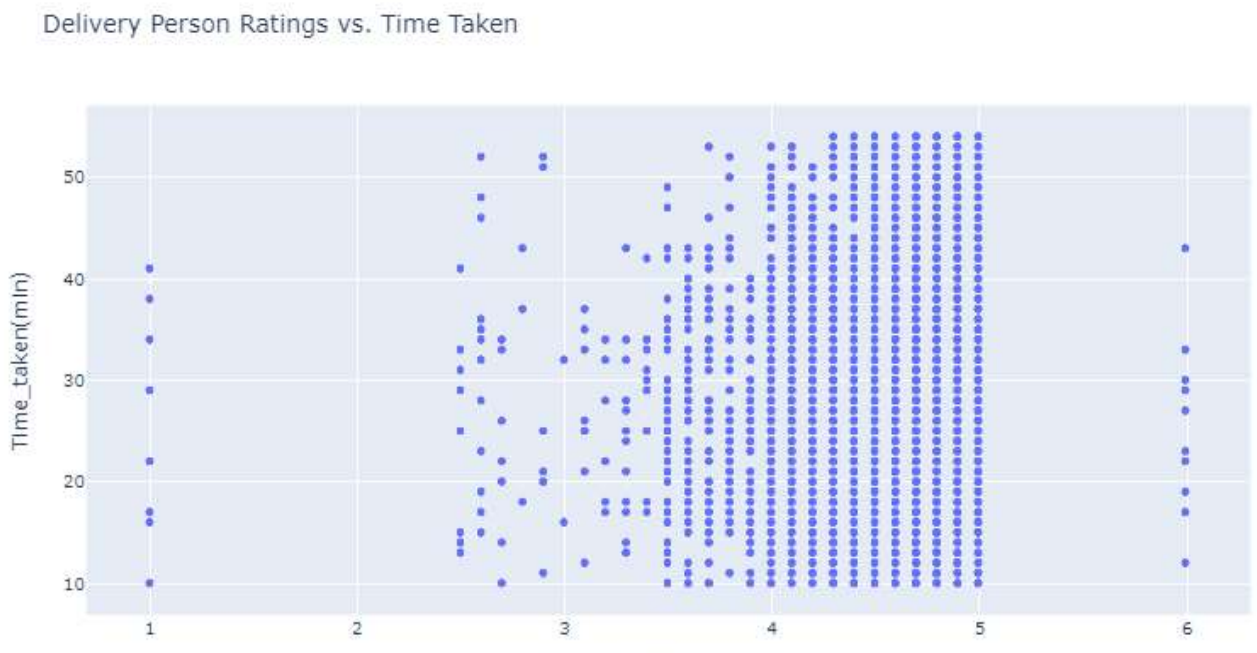
8) Graph

In [29]:
```
# Scatter plot for Delivery Person Ratings vs. Time Taken
import plotly.express as px
import plotly.graph_objects as go
import pandas as pd
fig1 = px.histogram(food_delivery_data, x='Delivery_person_Age', title='Delivery Person Age Distribution')
fig1.show()
```



9) Graph

In [30]:
```
# Histogram for Delivery Person Age Distribution
fig2 = px.scatter(food_delivery_data, x='Delivery_person_Ratings', y='Time_taken(min)',
                  title='Delivery Person Ratings vs. Time Taken')
fig2.show()
```



8. WEEKLY REPORT

➢ Planned Activities for Next Week:

❖ **Data Collection and Preprocessing:**
- Gather historical data on food deliveries.
- Organize information for model training.

❖ **Machine Learning Model Development**:
- Train models for accurate delivery time predictions.

❖ **Algorithm Implementation:**
- Develop algorithms considering distance, traffic, and real-time adjustments.

❖ **CSV Data Management:**
- Implement structured data storage for efficient delivery record management.

❖ **Functionality Design:**
- Design user-friendly features for reporting and analyzing delivery data.
-

❖ **Internal Testing:**
- Evaluate system performance and address potential bugs.

➢ **Documentation and Reporting:**
- Document algorithms, data structures, and system functionalities.
- Generate a report summarizing testing results and adjustments.

9. CONCLUSION

➢We aimed to make our food delivery service better by predicting delivery times accurately. Using data like delivery person details, location, order times, and more, we built a model. Here's what we found:

1. **Model's Performance:**

•Our chosen model did a great job predicting delivery times, showing it's accurate**.**

2. **Key Features:**

•Some factors, like traffic and weather, really matter in predicting delivery times. Knowing this helps us improve our delivery operations.

3. **Challenges:**

•While our model worked well, we found some challenges, like dealing with changing weather and traffic. Fixing these can make our predictions even better.