Technical Architecture:

Project Design Phase-II Technology Stack (Architecture & Stack)

|  |  |
| --- | --- |
| Date | 2 November 2023 |
| Team ID | Team-591756 |
| Project Name | ECOMMERCE SHIPPING PREDICTION USING MACHINE LEARNING |
| Maximum Marks | 4 Marks |

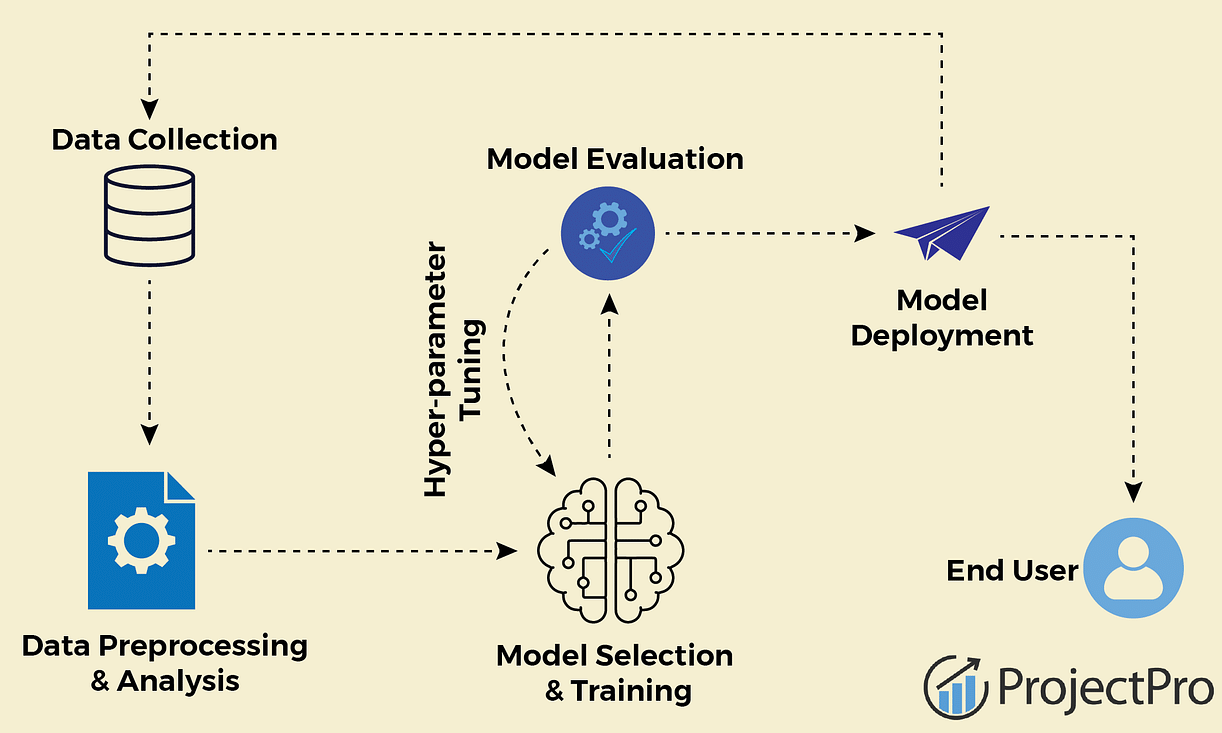


Table-1 : Components & Technologies:

|  |  |  |  |
| --- | --- | --- | --- |
| **S.No** | **Component** | **Description** | **Technology** |
| 1. | Data Collection | Collecting historical shipping data including features like product details, customer location, time of year, etc. | Ecommerce platforms, Web scraping tools, APIs. |
| 2. | Data Preprocessing | Cleaning and transforming raw data is crucial for machine learning models | Python libraries like Pandas and NumPy for data manipulation, Scikit-Learn for preprocessing. |
| 3. | Feature Engineering | Creating new features from the raw data to help improve the performance of the machine learning model | Domain knowledge, Python, Pandas. |
| 4. | Machine Learning Model | Training a machine learning model to predict shipping time based on the processed features. | Scikit-Learn, XGBoost, LightGBM, TensorFlow, PyTorch. |
| 5. | Model Evaluation | Evaluating the performance of the model using appropriate metrics. | Scikit-Learn's metrics module for Mean Absolute Error, Root Mean Squared Error etc |
| 6. | Hyperparameter Tuning | Optimizing the parameters of the machine learning algorithm to improve performance. | GridSearchCV or RandomizedSearchCV in Scikit-Learn |
| 7. | Model Deployment | Deploying the trained model for making predictions in real-time | Flask for creating APIs, Cloud platforms (e.g., AWS, Azure, GCP). |
| 8. | User Interface | Creating a user-friendly interface for users to interact with the shipping prediction system. | | Web development (HTML, CSS, JavaScript), mobile app development (e.g., React Native, Flutter) |
| 9. | Continuous Improvement | Regularly update and improve the system by collecting user feedback and enhancing the model. | Agile development practices, version control (e.g., Git). |

Table-2: Application Characteristics:

|  |  |  |  |
| --- | --- | --- | --- |
| **S.N**  **o** | **Characteristics** | **Description** | **Technology** |
| 1. | Data Preprocessing | Cleaning and transforming raw data is crucial for machine learning models. | Python libraries like Pandas and NumPy for data manipulation, Scikit-Learn for preprocessing. |
| 2. | Machine Learning Algorithms | Choosing the right algorithm for prediction. | Scikit-Learn for traditional machine learning algorithms, XGBoost or LightGBM for gradient boosting. |

|  |  |  |  |
| --- | --- | --- | --- |
| 3. | Model Evaluation | | Evaluating the performance of the model | Use Scikit-Learn's metrics module for accuracy, precision, recall etc. |
| 4. | Feature Selection | Dentifying the most relevant features for the prediction. | Techniques like correlation matrices, Recursive Feature Elimination |
| 5. | Hyperparameter Tuning | Optimizing the parameters of the machine learning algorithm to improve performance. | Optimizing the parameters of the machine learning algorithm to improve performance. |