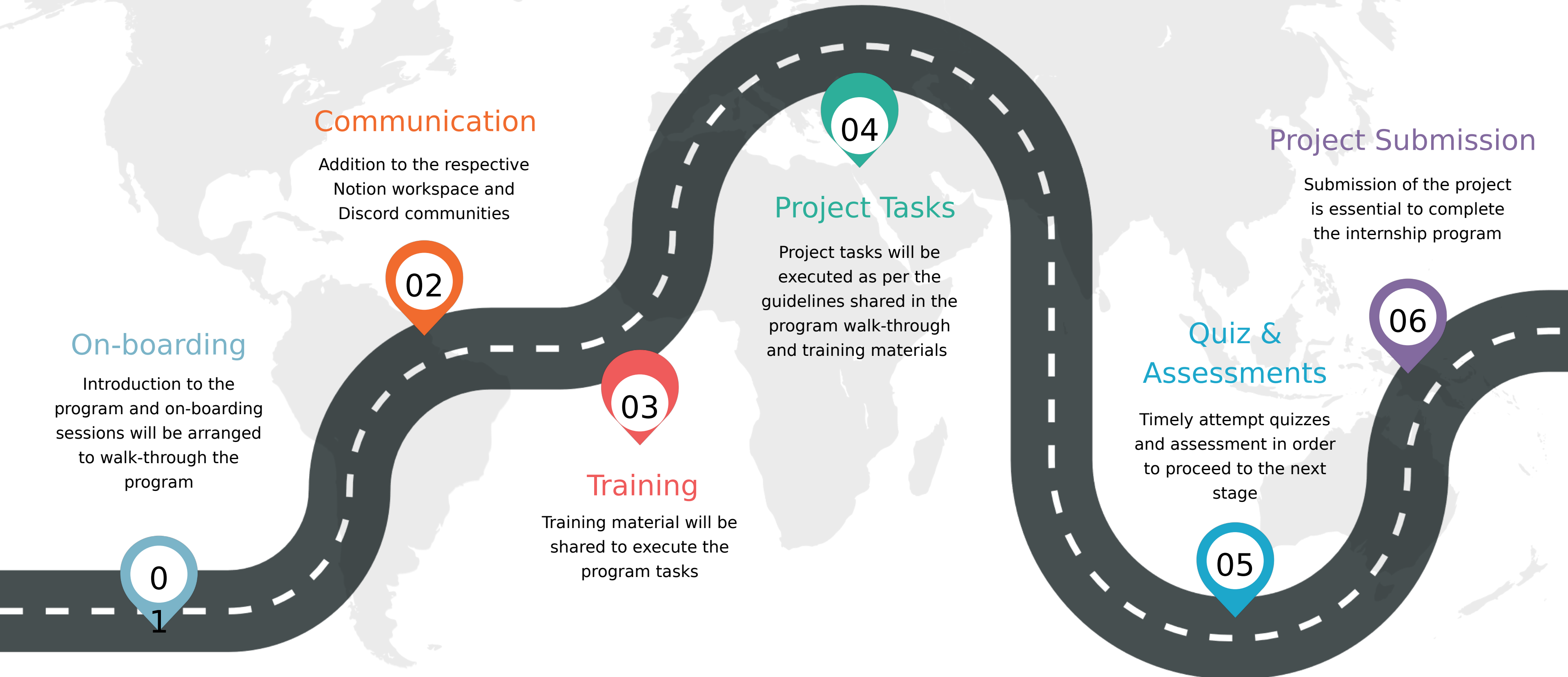




# 10P SHINE

INTERNSHIP PROGRAM

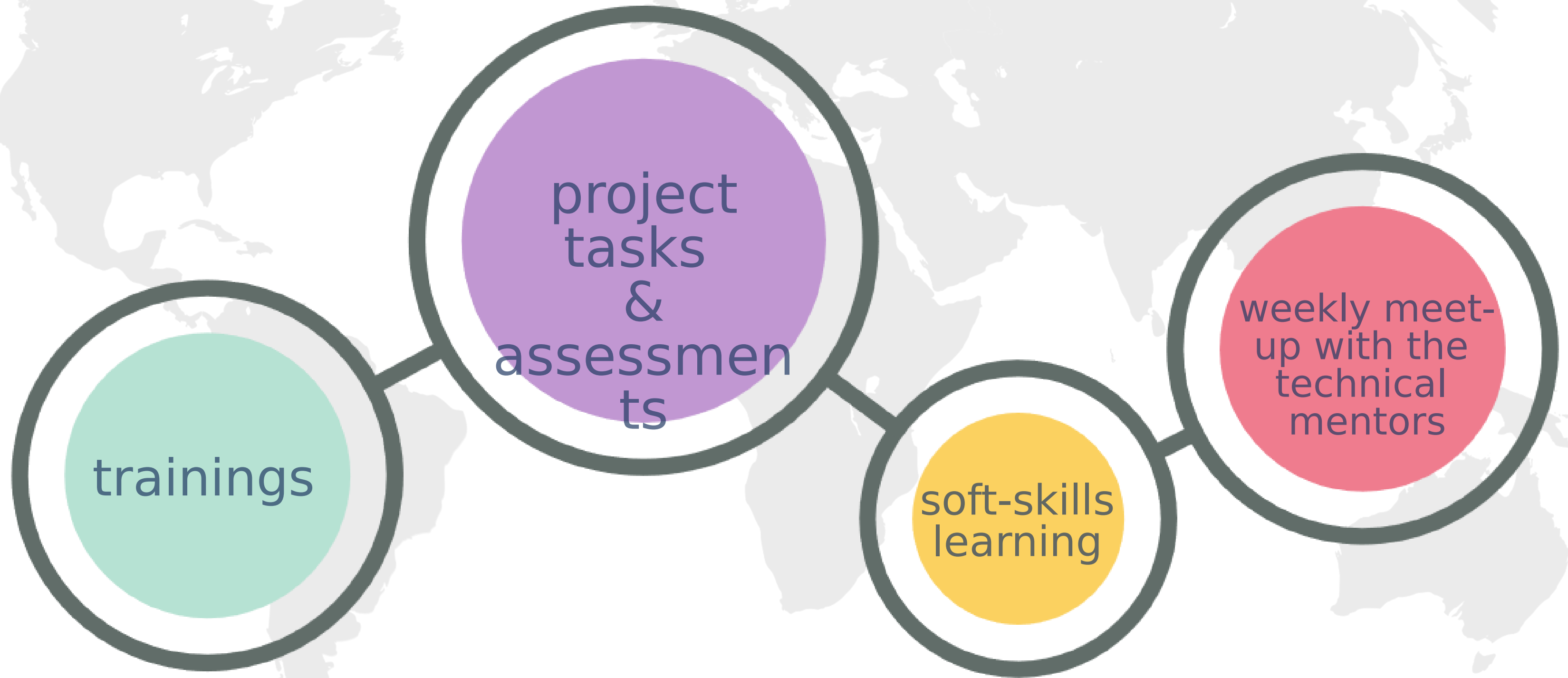
# WHAT TO EXPECT



# DAILY RUN-DOWN



# WEEKLY RUN-DOWN



# FACILITATORS



UMMARA AMIN  
ummara.amin@10pearls.com



SYED ANUS INTIZAR AHMED  
anus.intizar@10pearls.com



SAMIA ALI  
samia.ali@10pearls.com

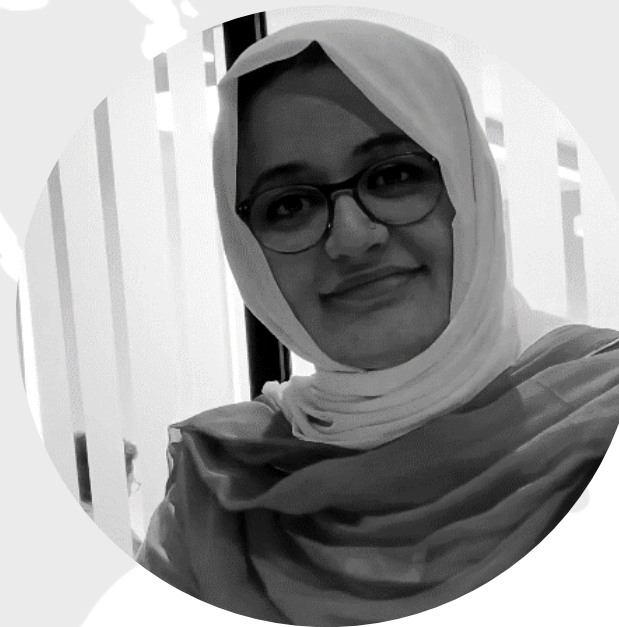


# WHO TO CONTACT?

## Data Sciences



POC:  
Ambreen Gul  
([ambreen.gul@10pearls.com](mailto:ambreen.gul@10pearls.com))



Technical Mentor:  
Syeda Zainab Yousuf Zaidi  
([zainab.yousuf@10pearls.com](mailto:zainab.yousuf@10pearls.com))

# NON-TECHNICAL TRAININGS

## Non-Technical Trainings

### Soft-Skills Learning

Topic 1

Topic 2

Topic 3

### HR Learning

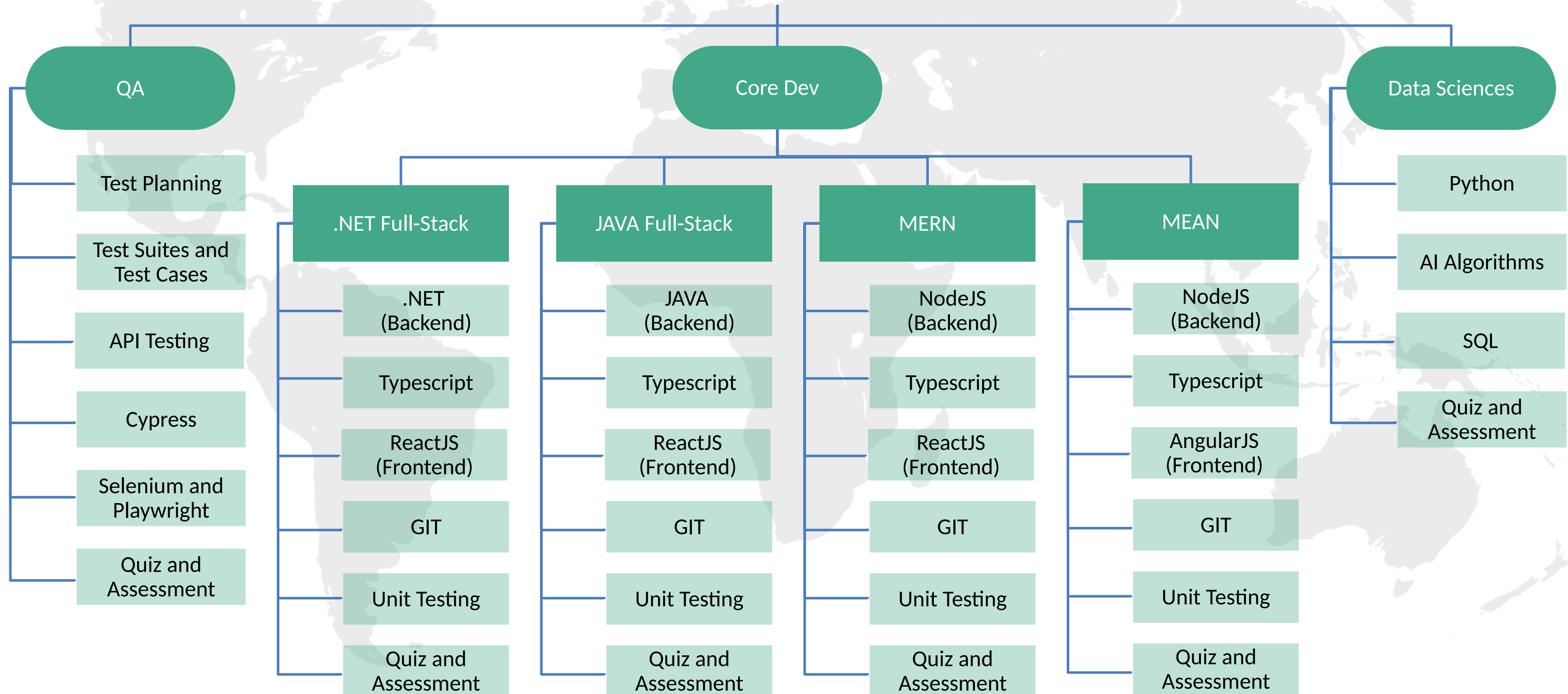
Topic 1

Topic 2

Topic 3

# TECHNICAL TRAININGS

## Technical Trainings





[illegible][illegible]



## PROJECT OVERVIEW

The objective of this project is to analyze customer churn data from a telecommunications company, develop a predictive model to identify customers who are likely to churn, and manage the data using SQL. The project will be divided into three main modules: Python, AI Algorithm, and SQL, each focusing on different aspects of the analysis and implementation.

# TECHNOLOGY STACK



Python: Jupyter Notebook, pandas, scikit-learn, matplotlib, seaborn, SHAP



SQL: MySQL or PostgreSQL



AI Algorithm: Logistic Regression, Decision Trees, Random Forest, Gradient Boosting, SVM



Data Visualization: matplotlib, seaborn



Database Management: MySQL Workbench or pgAdmin



Web Framework: Flask



API Development and Testing: Flask, Postman, curl



# DATASET OVERVIEW

Each row represents a customer, each column contains customer's attributes described on the column Metadata.

The data set includes information about:

- Customers who left within the last month – the column is called Churn
- Services that each customer has signed up for – phone, multiple lines, internet, online security, online backup, device protection, tech support, and streaming TV and movies
- Customer account information – how long they've been a customer, contract, payment method, paperless billing, monthly charges, and total charges
- Demographic info about customers – gender, age range, and if they have partners and dependents

# MODULE 1: PYTHON

## Tasks:

### 1. Data Preprocessing:

- Load the dataset and inspect the data.
- Handle missing values.
- Convert categorical variables into numerical format using encoding techniques (e.g., One-Hot Encoding).
- Normalize/Standardize the data if necessary.

### 2. Exploratory Data Analysis (EDA):

- Generate descriptive statistics.
- Visualize the data distribution for various features (e.g., histograms, box plots).
- Analyze the relationships between features and the target variable (Churn).
- Use correlation matrices to identify highly correlated features.

### 3. Feature Engineering:

- Create new features that might help improve the model's performance (e.g., interaction terms, polynomial features).
- Select important features using techniques like feature importance from tree-based models.

## Deliverables:

1. Jupyter notebook with data preprocessing steps.
2. EDA visualizations and insights.
3. Feature engineering process and final dataset ready for modeling.

# MODULE 2: AI ALGORITHM

## Tasks:

### 1. Model Selection:

- Choose a set of machine learning algorithms to evaluate (e.g., Logistic Regression, Decision Trees, Random Forest, Gradient Boosting, Support Vector Machines).
- Split the dataset into training and testing sets.

### 2. Model Training:

- Train multiple models and evaluate their performance using cross-validation.
- Use appropriate evaluation metrics (e.g., accuracy, precision, recall, F1-score, ROC-AUC).

### 3. Model Tuning:

- Perform hyperparameter tuning using techniques such as Grid Search or Random Search.
- Select the best model based on performance metrics.

### 4. Model Evaluation:

- Evaluate the final model on the test set.
- Generate a classification report and confusion matrix.
- Plot ROC and Precision-Recall curves.

### 5. Model Interpretation:

- Interpret the model's predictions using techniques like SHAP (SHapley Additive exPlanations) values.
- Identify the most important features driving the predictions.

## Deliverables:

1. Jupyter notebook with model training, tuning, and evaluation steps.
2. Final model saved for deployment.
3. Interpretation of the model's predictions.



# MODULE 3: SQL

## Tasks:

### 1. Database Schema Design:

- Design a relational database schema to store the telco customer churn data, including raw data and prediction results.
- Define tables for customers, services, billing, and churn predictions.

### 2. Data Ingestion:

- Write SQL scripts to create the database schema.
- Import the preprocessed dataset into the SQL database.
- Ensure data integrity and consistency with primary and foreign keys.

### 3. Querying the Data:

- Write SQL queries to retrieve customer information based on various criteria (e.g., tenure, monthly charges, contract type).
- Generate summary reports, such as the total number of customers, average tenure, and total revenue.

### 4. Storing Predictions:

- Create a table to store model predictions.
- Write a script to insert the model's predictions into the SQL database.
- Ensure predictions are linked to customer records via foreign keys.

### 5. Advanced SQL Analysis:

- Perform complex queries to analyze churn patterns and customer segments.
- Use window functions to calculate running totals and moving averages.

## Deliverables:

1. SQL scripts for creating the database schema.
2. Scripts for importing data and storing predictions.
3. SQL queries for data retrieval and analysis.
4. A report summarizing insights from the SQL analysis.

# MODULE 4: Model Deployment and API Consumption (Optional)

## Objective:

Deploy the trained model using Flask and create APIs to consume the model predictions.

## Tasks:

### 1. Flask Setup:

- Install Flask and set up the project environment.
- Create the necessary folder structure for the Flask application.

### 2. Model Integration:

Load the saved model into the Flask application.

- Implement endpoints for the API (e.g., /predict).

### 3. API Development:

- Develop API endpoints for the following functionalities:
- Prediction: Endpoint to receive input data and return predictions.
- Health Check: Endpoint to check the status of the API.

### 4. Testing and Validation:

- Test the API endpoints using tools like Postman or curl.
- Validate the predictions returned by the API.

### 5. Documentation:

- Document the API endpoints with details on how to use them.
- Provide example requests and responses.

## Deliverables:

1. Flask application code for model deployment.
2. API endpoints for model consumption.
3. Documentation for API usage.
4. Example scripts or Postman collection for testing the API.



# FINAL DELIVERABLES

1. Comprehensive Jupyter notebooks for each module.
2. SQL scripts for database management and analysis.
3. Flask application code for model deployment.
4. API documentation and testing scripts.
5. A final report summarizing the project findings and insights.

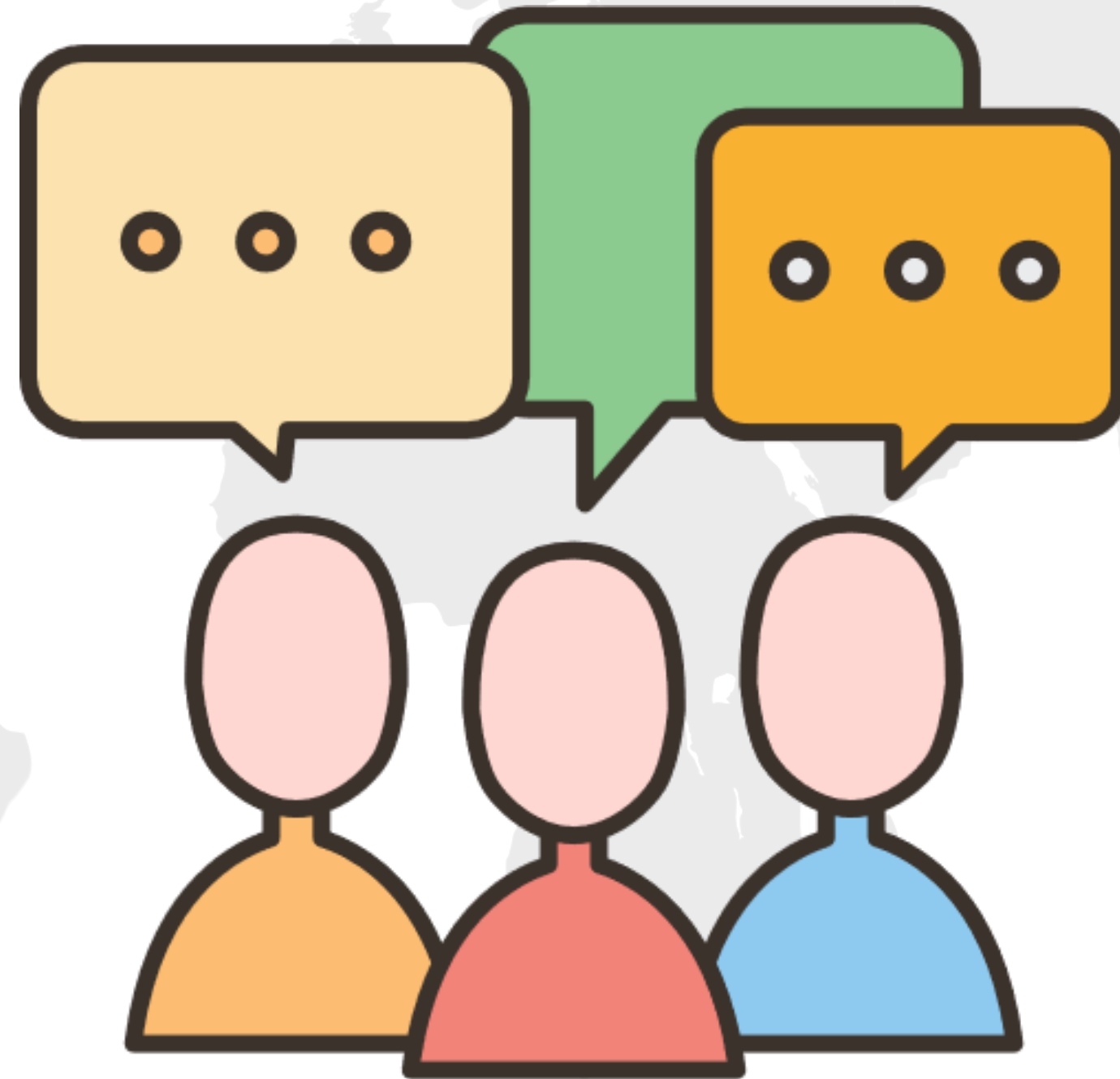
# THINGS TO REMEMBER



Medium of  
Instant  
Communication:  
Discord  
Community

Medium of  
Formal  
Communication:  
Notion  
Work-space

GOT QUESTIONS?



LETS DISCUSS

## OUR SOCIALS



@10pearlspk



/10Pearls.Pakistan



@10pearlspk



/10Pearls-Pakistan



[www.10pearls.com/career](http://www.10pearls.com/career)



[recruitment-Pakistan@10pearls.com](mailto:recruitment-Pakistan@10pearls.com)