



CANOPY SENTINEL

SATELLITE-BASED DEFORESTATION
MONITORING SYSTEM



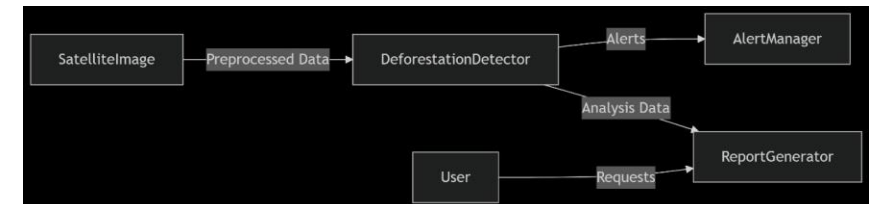
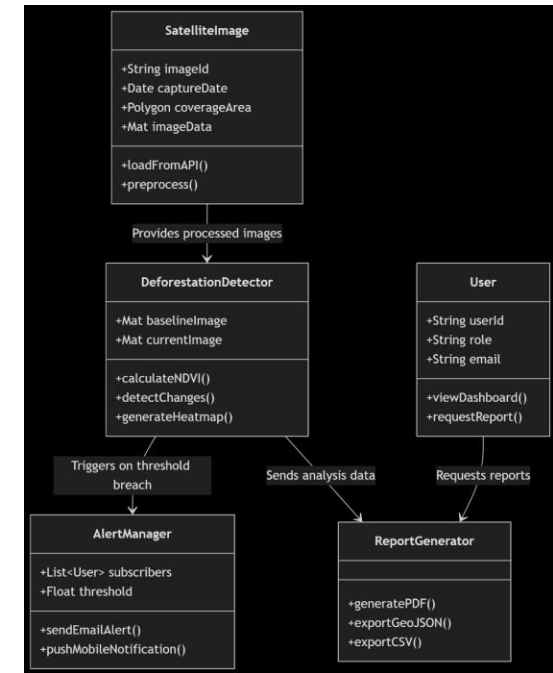
checklist

REVIEW 01

- Creating the new project with JDK & IDE setup ✓
- Define the project structure ✓
- Design the database schema for the project ✓
- Create a MySQL table ✓
- Implement jdbc for database connectivity ✓
- Create Model, DAO classes for the database operations ✓
- Aesthetics and Visual Appeal of the UI ✓
- Component Placement and Alignment in the UI ✓
- Responsiveness and Accessibility of the UI ✓

CLASS DIAGRAMS

- flowchart LR
- A[SatelliteImage] -->|Preprocessed Data| B[DeforestationDetector]
- B -->|Alerts| C[AlertManager]
- B -->|Analysis Data| D[ReportGenerator]
- E[User] -->|Requests| D





Detailed Class Diagram CODE

Class: SatelliteImage

- **Attributes:**

- String imageId
- Date captureDate
- Polygon coverageArea
- Mat imageData

- **Methods:**

- loadFromAPI()
- preprocess()

Class:

DeforestationDetector

- **Attributes:**

- Mat baselineImage
- Mat currentImage

- **Methods:**

- calculateNDVI()
- detectChanges()
- generateHeatmap()

Class: AlertManager

- **Attributes:**

- List<User> subscribers
- Float threshold

- **Methods:**

- sendEmailAlert()
- pushMobileNotification()

Class:

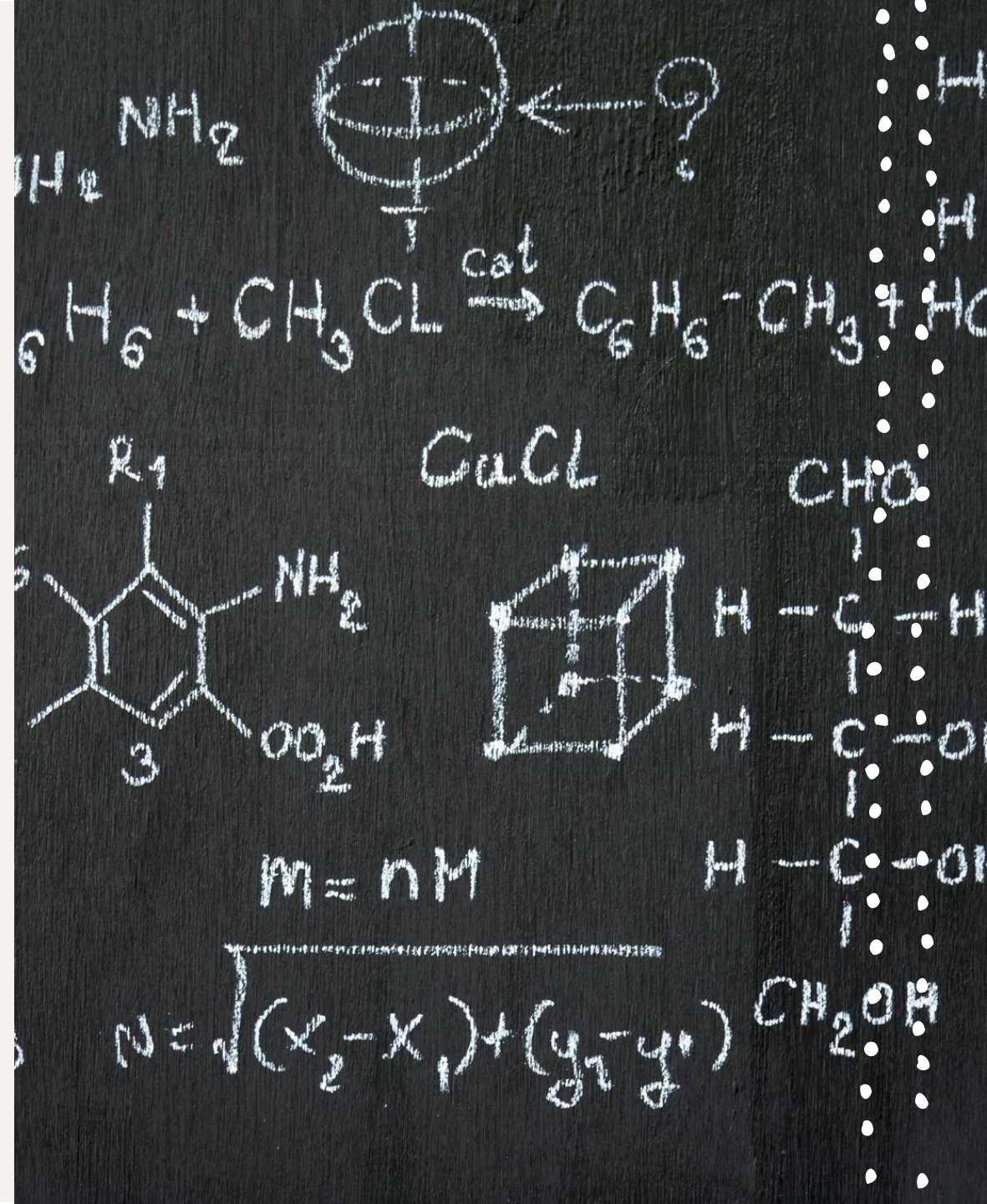
ReportGenerator

- **Methods:**

- generatePDF()
- exportGeoJSON()
- exportCSV()

Continued...

- **Class: User**
- **Attributes:**
 - String userId
 - String role
 - String email
- **Methods:**
 - viewDashboard()
 - requestReport()



FLOWCHART



SATELLITEIMAGE -->
DEFORESTATIONDETECTOR :
PROVIDES PROCESSED IMAGES



DEFORESTATIONDETECTOR -->
ALERTMANAGER : TRIGGERS ON
THRESHOLD BREACH



DEFORESTATIONDETECTOR -->
REPORTGENERATOR : SENDS
ANALYSIS DATA



USER --> REPORTGENERATOR :
REQUESTS REPORTS

CLASS DIAGRAMS CONTINUED

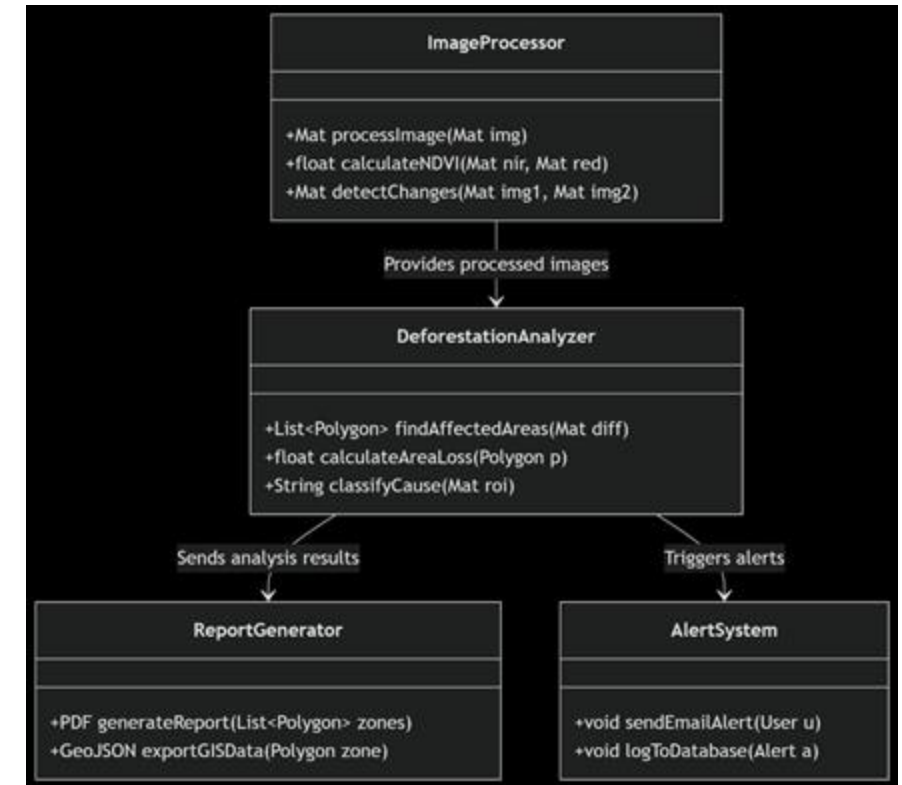
- **class: ImageProcessor**

- **Methods:**

Mat processImage(Mat img)

float calculateNDVI(Mat nir, Mat red)

Mat detectChanges(Mat img1, Mat img2)



Continued..

class: DeforestationAnalyzer

- **Methods:**
 - `List<Polygon> findAffectedAreas(Mat diff)`
 - `float calculateAreaLoss(Polygon p)`
 - `String classifyCause(Mat roi)`

Class: ReportGenerator

- **Methods:**
 - `PDF generateReport(List<Polygon> zones)`
 - `GeoJSON exportGISData(Polygon zone)`

Class: AlertSystem

- **Methods:**
 - `void sendEmailAlert(User u)`
 - `void logToDatabase(Alert a)`

FLOWCHART



IMAGEPROCESSOR -->
DEFORESTATIONANALYZER :
PROVIDES PROCESSED IMAGES



DEFORESTATIONANALYZER -->
REPORTGENERATOR : SENDS
ANALYSIS RESULTS



DEFORESTATIONANALYZER -->
ALERTSYSTEM : TRIGGERS
ALERTS

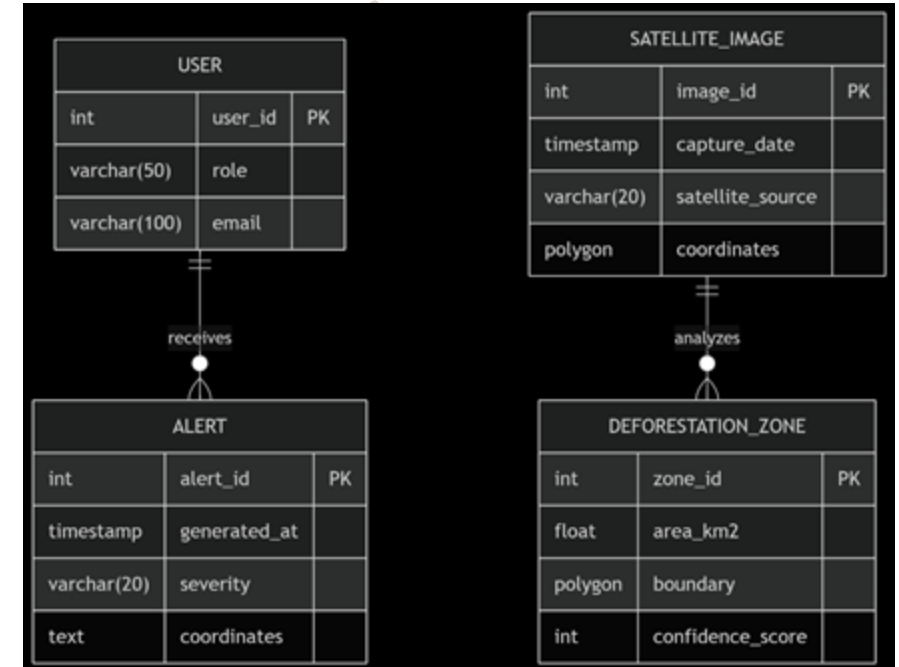
ER DIAGRAM & CODES

Entity: USER

- `user_id` (int, **PK**)
- `role` (varchar(50))
- `email` (varchar(100))

Relationships:

- **USER** || --o{ **ALERT** : *receives*



Continued..

Entity: SATELLITE_IMAGE

- image_id (int, **PK**)
- capture_date (timestamp)
- satellite_source (varchar(20))
- coordinates (polygon)

Relationships:

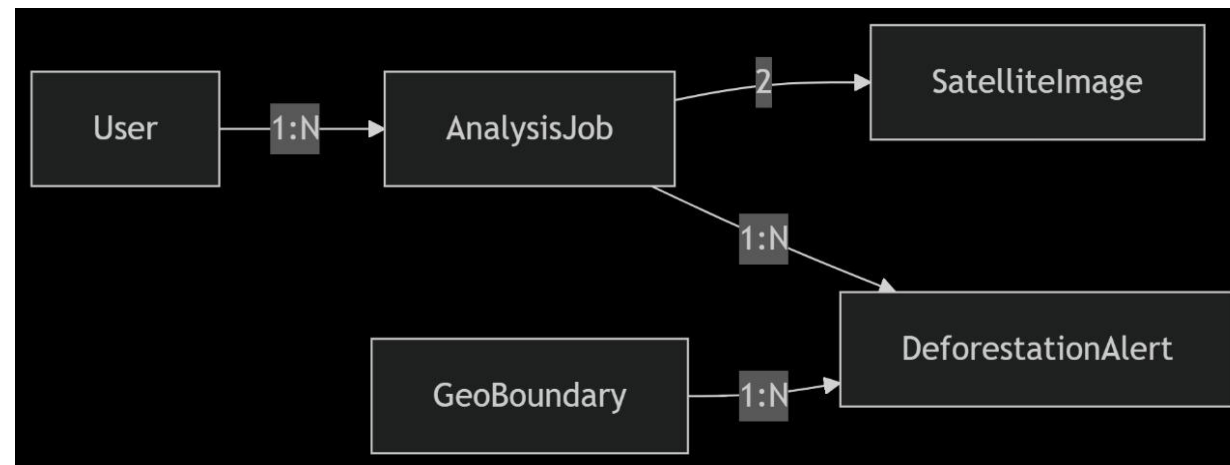
- **SATELLITE_IMAGE** || --o{ **DEFORESTATION_ZONE** : *analyzes*

Entity: DEFORESTATION_ZONE

- zone_id (int, **PK**)
- area_km2 (float)
- boundary (polygon)
- confidence_score (int)

Entity: ALERT

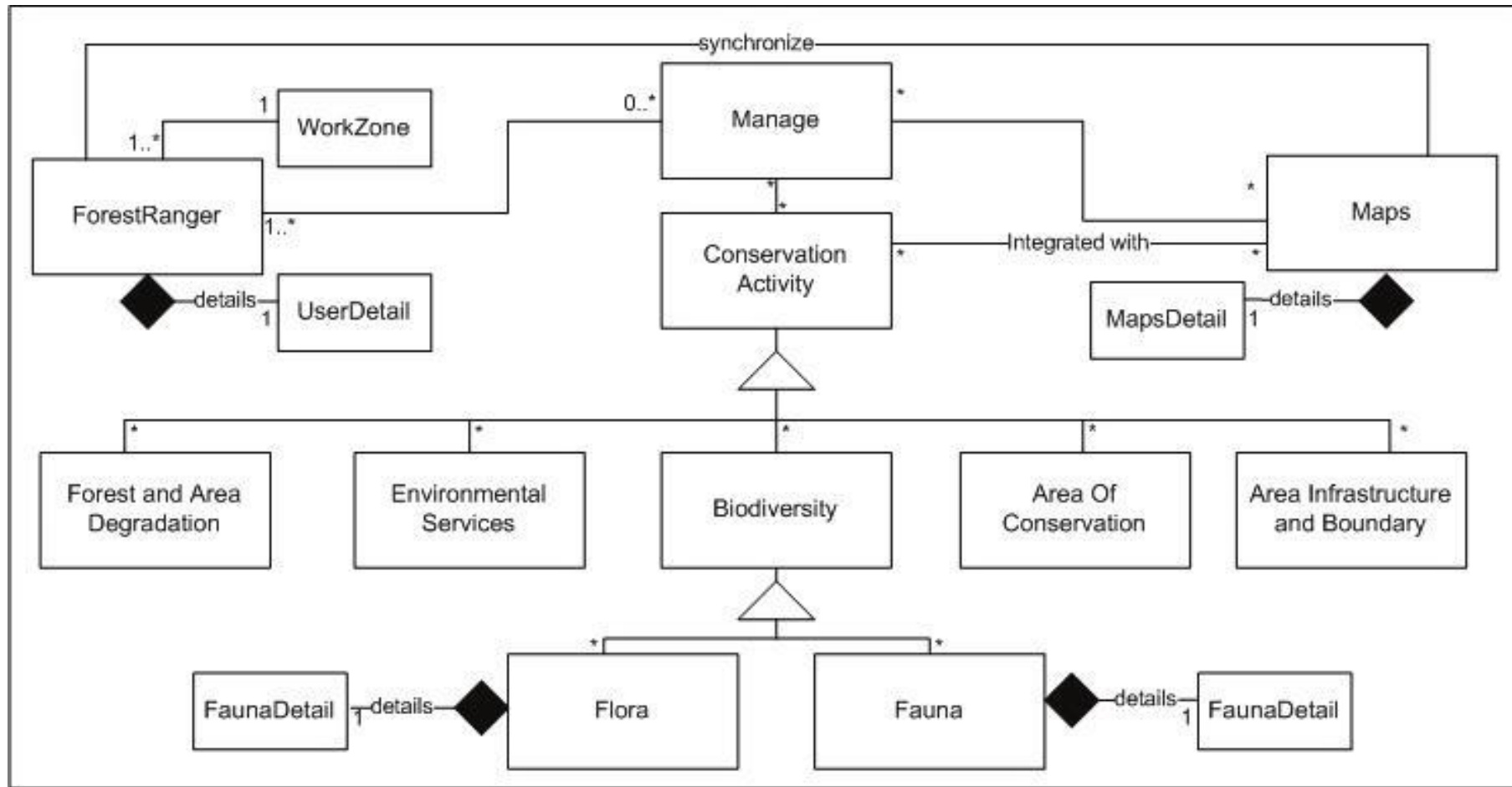
- alert_id (int, **PK**)
- generated_at (timestamp)
- severity (varchar(20))
- coordinates (text)





FLOWCHART

- flowchart LR
- User -->|1:N| AnalysisJob
- AnalysisJob -->|2| SatellitelImage
- AnalysisJob -->|1:N| DeforestationAlert
- GeoBoundary -->|1:N|
DeforestationAlert



UML DIAGRAM

APP UI DONE SO FAR

FIRST PAGE

INTERFACE



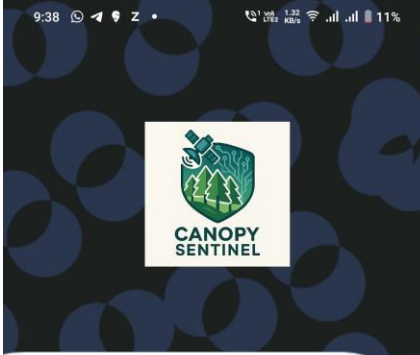
Welcome To Canopy Sentinal



**Satellite Based Deforestation
Monitoring System**

APP UI

SECOND LOGIN PAGE



The image shows a mobile app login screen for 'Canopy Sentinel'. At the top, there's a status bar with the time 9:38, signal strength, and battery level at 11%. Below the status bar is a dark blue header with a pattern of overlapping circles. In the center of the header is a white square containing the Canopy Sentinel logo, which features a green tree and the text 'CANOPY SENTINEL'. Below the header, the word 'Login' is written in a large, bold, black font. Underneath 'Login' is the text 'Sign in to continue'. There are two input fields: one for 'EMAIL' with a placeholder 'Email' and one for 'PASSWORD' with a placeholder 'Password'. Below the input fields is a green button with the text 'LOGIN'. At the bottom, there is a link that says 'Need an account?' followed by a 'SIGN UP!' button. A horizontal line is at the very bottom of the screen.

9:38 100% 1.27 4G LTE 11%

Canopy Sentinel

Login

Sign in to continue

EMAIL

Email

PASSWORD

Password

LOGIN

Need an account?

SIGN UP!

APP UI

THIRD PAGE
ACCOUNT PAGE

Create New Account



USERNAME

EMAIL

Password

RE-ENTER
PASSWORD

SIGNUP

[Login](#)

PROJECT TO BE CONTINUED...

A decorative horizontal line consisting of a series of small, light-colored dots that spans the width of the slide, positioned above the team name.

TEAM ALGOTITANS

VAYU NANDAN TRIPATHI
ALOK MISHRA
ANANYA MITTAL
APOORVA DWIVEDI