Software Requirements Specification

for

DiCRA Geospatial Platform

Version 1

mistEO Private Limited

Submitted to

India

19.03.2023

Privacy Policy

This document contains proprietary and confidential information. All data submitted to the recipient is provided in reliance upon the consent not to use or disclose any information contained herein except in the context of business dealings with mistEO Pvt. Ltd. (mistEO) and United Nations Development Programme - India, (UNDP). Proprietary and confidential information includes, but is not limited to, the following:

- Corporate, employee and infrastructure information about mistEO and UNDP-India
- mistEO and UNDP-India's project management and quality processes;
- Customer and project experiences provided to illustrate mistEO's and UNDP-India's capabilities;
- All commercial information;

Any disclosure of Confidential Information to, or use of it by a third party (i.e., a party other than to whom it is intended to) will be damaging to mistEO or UNDP-India. Ownership of all Confidential Information, no matter in what media it resides, remains with mistEO and UNDP-India. The recipient agrees not to duplicate or distribute or permit others to duplicate or distribute any material contained herein without the written consent of mistEO and UNDP-India.

Table of Contents

Revision History

Version	Date	Changes
1	23.03.2023	Approved for Release

1. Introduction

1.1 Purpose

UNDP has initiated the NextGenGov 'Data for Policy' initiative on Food Systems. The aim is to incorporate anticipatory governance models for future-fit food systems in various States of India using data-driven policymaking tools and ecosystem-driven approaches. UNDP is keen on augmenting learning capabilities, increasing the predictive or anticipatory capacity to feed-in to evidence-driven policies in various States, and create radical traceability and transparency across the system from producers to consumers by building provenance documentation around food that can help build trust in the system and at the same time nurture sustainable and healthy practices. The goal is to design, develop and demonstrate anticipatory governance models for food systems in various states using digital public goods and community-centric approaches to strengthen data-driven policy making.

The Food Systems Innovation platform is envisioned as a Digital Public Good that will strategically feed into data-driven decision making. The platform will have the capability to visualize and analyze high resolution geospatial data (both vector as well as raster layers). The digital platform will need to curate, integrate and visualize such critical datasets and assets to answer the basic question of - What is growing where? How much is there and the spatial and temporal changes across various indicators relevant to Agriculture and Food Systems. The platform should be able to visualize over time the changes that have happened to the agriculture ecosystem in terms of crop diversity, changes in soil/ground water, tree cover, and other indicators at higher resolution to support policy decisions. Such a synthesis of data and analytics can help identify farms which are doing exceptionally well (Positive Deviance) through which repositories of good practices and indigenous knowledge can be documented. This also helps in identifying farms that are not doing good as per the defined indicators (Negative Deviance).

1.2 Document and Intended Audience

The system requirements specification document describes what the system is to do, and how the system will perform each function. The audiences for this document include the system developers and the users. The system developer uses this document as the authority on designing and building system capabilities. The users review the document to ensure the documentation completely and accurately describes the intended functionality.

This document provides general descriptions of the system. The system developers should review the document to ensure there is adequate information for defining an initial design of the system. The users should review the document to affirm the features described are needed, to clarify features, and to identify additional features needed within the system.

1.3 Project Scope

In the first phase of digital technology platform development, a user-friendly web application must be developed using open software, open tools, open code and open APIs. The second phase the functionalities of the web application are as follows:

- Scale up the platform for six States of India;
- Integrate different formats of geospatial data at high-resolution (provided by UNDP & partner organizations) with ability to have API based access to various databases and local storage;
- Visualize geospatial data and render an interactive user experience and interface;
- Ability to download the data after successful registration on the web application;
- Ability to crowdsource use cases and projects that are actively using datasets curated in the platform;
- An authentication and authorization module to screen the crowdsourced projects and approve them to be visualized on the platform;

1.4 References

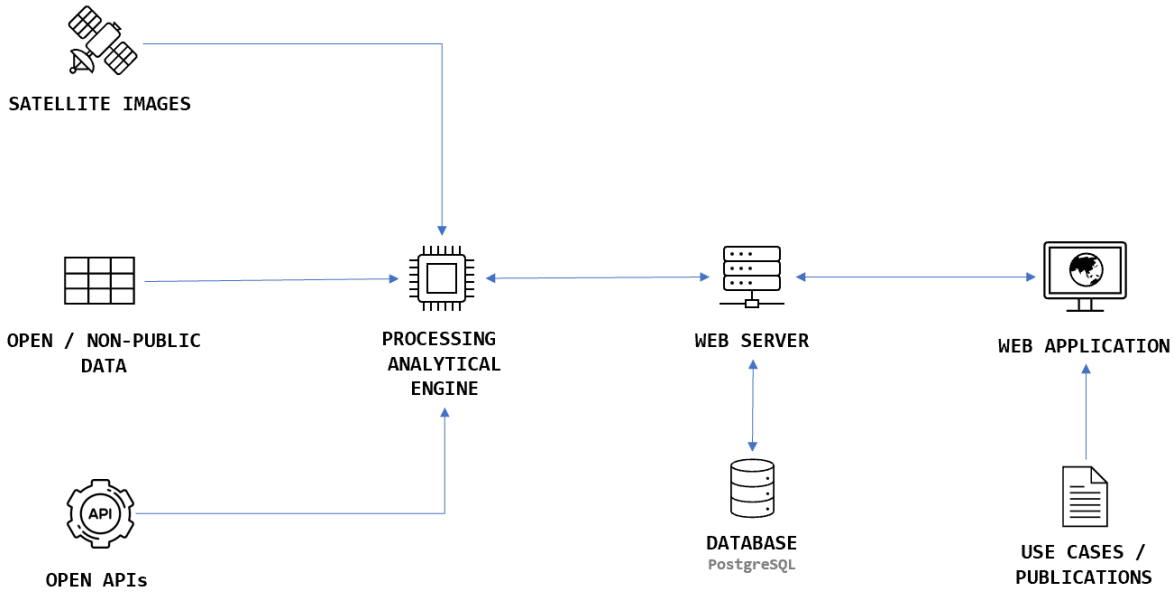
- https://github.com/undpindia/dicra
- https://github.com/UNDP-India/Data4Policy
- https://github.com/undpindia/dicra/blob/main/automation/setup/DiCRA_DATA_PIPELINE.pdf
- https://github.com/undpindia/dicra/blob/main/docs/manuals/installation-guide/MPL_Docker_Swarm_Setup.pdf

2. Overall Description

2.1 Product Perspective and Features

The Food Systems Innovation platform is envisioned as a Digital Public Good that will strategically feed into data-driven decision making in the State. The platform will have the capability to visualize and analyze high resolution geospatial data (both vector as well as raster layers). The digital platform will need to curate, integrate and visualize such critical datasets and assets to answer the basic question of - What is growing where? How much is there and the spatial and temporal changes within the State across various indicators relevant to Agriculture and Food Systems.

The platform should be able to visualize over time the changes that have happened to the agriculture ecosystem in terms of crop diversity, changes in soil/ground water, tree cover, and other indicators at higher resolution to support policy decisions. Such a synthesis of data and analytics can help identify farms which are doing exceptionally well (Positive Deviance) through which repositories of good practices and indigenous knowledge can be documented. This also helps in identifying farms that are not doing good as per the defined indicators (Negative Deviance).



2.2 User Classes and Characteristics

System shall have only one type of user, ie Public User / Citizen

User	Roles
Public User / Citizen	• These users includes citizens as well as policy makers • These users will have complete access to the system

2.3 Operating Environment

The rule for selecting hardware and software is that the components/application must be functionally efficient, capable of interfacing with other software, and easy to maintain.

- The System shall operate with all the leading Web browsers.
- The System shall operate from servers on Microsoft Azure

2.4 Design and Implementation Constraints

The goal of the web application is to be platform independent on the client side wherever possible. Therefore, the web applications will be implemented to run on the server side as much as possible. Also, it is required to test the application using different platforms, connection speeds, screen settings, colors/graphics, and browsers.

DIC-1 : The system shall be in compliance with all Accessibility, Web Design, and Security policies applicable.

DIC-2 : As part of standard operating procedures, a testing plan will be documented during the Design phase. The testing plan will be based on user roles, modules or use cases, required tasks and expected outcomes.

DIC-3 : The database structure should be complete as quickly as possible during the design stage but there should be room for modification without a large overhaul during later phases.

DIC-4 : The data/layers to be used should be finalized as fast as possible to avoid unexpected issues related to data processing.

DIC-5 : It may take some time to develop a data pipeline process for new datasets.

DIC-6 : Every component/libraries/data used in this project should be available as open source.

2.5 User Documentation

User documentation will consist of the several components usually expected of a modern web-based software application, including a tutorial, help pages, FAQs with an online request form, and a complete user's manual.

UDN-1 : A tutorial will provide a quick start, a walk-through of major system features, and a further reference source for the complete system features.

UDN-2 : A video explanation of the portal will be provided.

UDN-3 : The user's guide (or user manual) will contain sufficient information and instructions required to access and use the data system. It will include:

UDN-3.1 : Overview of the system features and architecture.
UDN-3.2 : Instructions for accessing the system.
UDN-3.3 : Samples of screens, where appropriate.

2.6 Assumptions and Dependencies

- It is anticipated that the data/layers required for this project will be provided by partner organizations of UNDP-India.
- The content required for the portal related to the project will be provided by UNDP-India.
- It is anticipated that there are no other user roles than public user / citizen.
- No authentication is required to access this platform.

3. System Features

Functional Requirements

3.1 Map view module

The main section on the portal.

FR-3.1-01 : Landing page

There should be a landing page to select the states and it should take the user to the corresponding states' portal.

FR-3.1-02 : Layer selection and view spatial data

The user should be able to select different layers to see it on top of the base map.

FR-3.1-03 : Location search

The user should be able to search the location through 3 different ways. Google Location API can be used here

a) Address Search - Nominatim API can be used.
b) Coordinate Search - Lat, Long Search
c) Administrative Name search

FR-3.1-04 : Option to choose Raster or Vector

The user should be able to switch between the raster or vector of a parameter layer as per its availability.

FR-3.1-05 : Map controls

The basic map control features like zoom-in, zoom-out & pan options should be available.

FR-3.1-06 : Map Legend

A colour palette like legend for a selected layer should be there. Min & Max Value or Low & High indication can be provided based on the layer.

FR-3.1-07 : Option to display partner logos

The logos of UNDP and the respective State Governments are considered as primary logos and will be placed at the top-right of the page. The logos of other partners can be placed at the bottom of the dashboard or at the 'about project' section.

FR-3.1-08 : Option to know more about the layer

The user will be able to see the details of the layer such as description, source, frequency etc by clicking on an info icon.

FR-3.1-09 : Option to download the data of selected layer

The download of selected layers can be triggered directly from the layer selection option.

FR-3.1-10 : Display value on hover

The display option of an attribute value on hover of the selected layer should be there.

FR-3.1-11 : Details section for a selected shape

There will be a modal visible on the right side of the dashboard. This will get triggered on click of a selected shape (active layer). The below mentioned information will be available.

- Details of selected shape
- Attribute value of selected shape
- Min, Max value
- Description of the layer
- Data Source
- Trend chart where it shows the temporal changes of the selected region of the parameter. (eg. NDVI value of a selected region)
- Option to download the current

FR-3.1-12 : Draw custom polygon (farm boundary)

The user should be able to draw a custom polygon and the details of the same should be displayed. The trend also needs a download option if available.

3.2 Data download

FR-3.2-01 : Data

The user should be able to download the selected data in different formats.

3.3 Use cases

FR-3.3-01 : Upload use cases

Option to upload use cases is required. The information like project details, data used, url and uploader's contact details will be collected.

FR-3.3-02 : Listing of Use cases

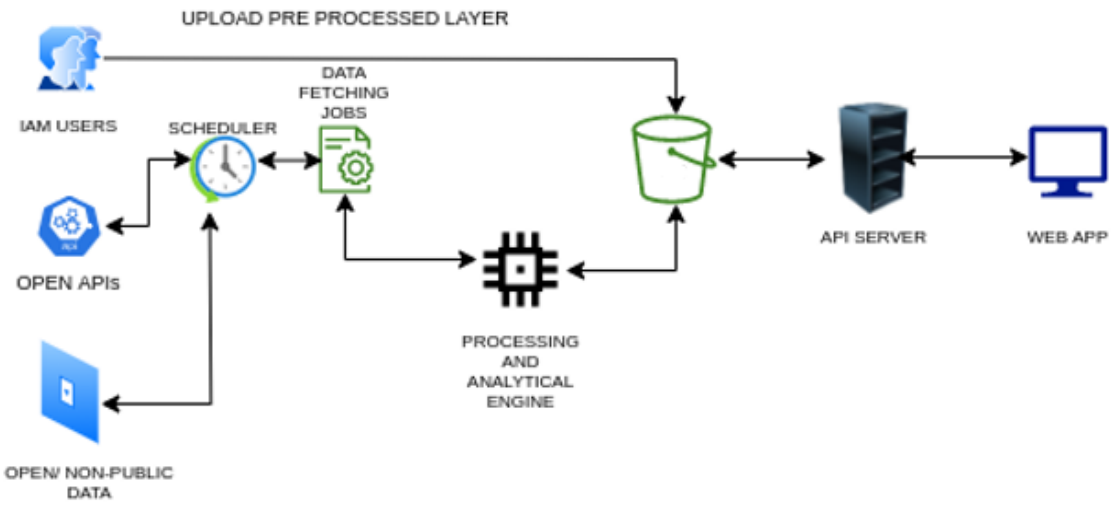
Option to list all approved use cases is required. The approval process for the same TBD.

FR-3.3-03 : Use Cases approval

The use case approval can be handled through updating the status on a configuration file. Based on the file, frontend loading can be decided.

3.4 Data Preprocessing & Engineering pipeline

The data preprocessing pipeline is dependent on the type & source of the data. The required process to be written to get the data. The data flow is explained below.



FR-3.4-01 : Fetch data functionality

Data/Layers come from different sources. So data extraction methods may differ and the different approaches to be handled as per the data.

FR-3.4-02 : Option to incorporate preprocessed data for partner organizations

There should be an option to update the required data/layers (preprocessed) to the storage. So that it can be consumed for the application.

FR-3.4-03 : Data download & process Scheduler

Develop a scheduler to fetch data in different frequencies and different sources.

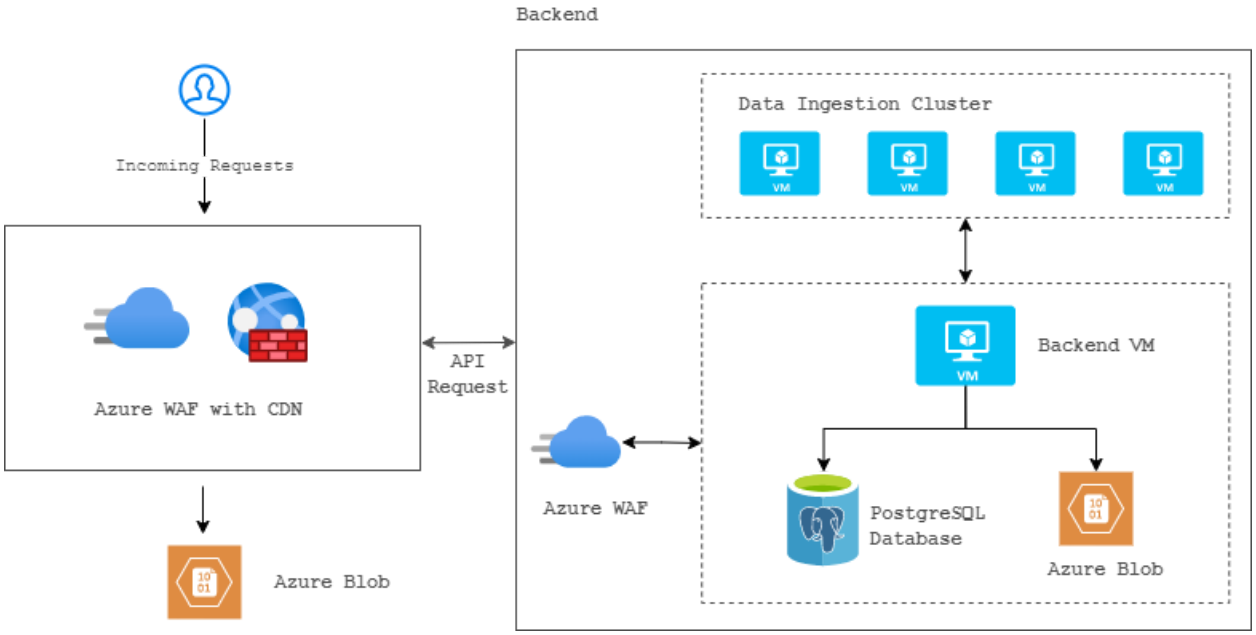
FR-3.4-04 : Data preprocessor

- Validation - Validating the format & projection
- Conversion - convert to the required format.

FR-3.4-04 : API Development

- API for add use cases
- API for list use cases
- API for getting zonal statistics against a shape
- API for getting zonal statistics against a shape - temporal trend
- API for getting layers for a user drawn location
- API for getting layer for current selected parameter
- API for download data for admin

4. Solution Architecture



The cloud architecture consists of the following components.

- Application Gateway WAF
- VM
- Azure Database for PostgreSQL
- Storage Account
- CDN

The application will be hosted under Microsoft Azure (India Region). The application deployed and the other services will be monitored.
Regarding the datastore, Azure Database for PostgreSQL will be used. The database is configured as global write and local read.

4.1 Security

The AWS services used for the application security

- WAF
- Security Group

4.1.1 Application Security

- Inbound traffic to the application will be whitelisted using the WAF and security groups. WAF will be only accepting the traffic from the pre-defined IPs.
- The Inbound traffic to the application will be used standard HTTP/HTTPS Ports for inbound traffic.
- All frontend forms will have captcha verification enabled.
- The data communication happens through an API that will have an encryption layer.
- The API server will verify the domain from which the request comes and act accordingly.

4.1.2 Database Security

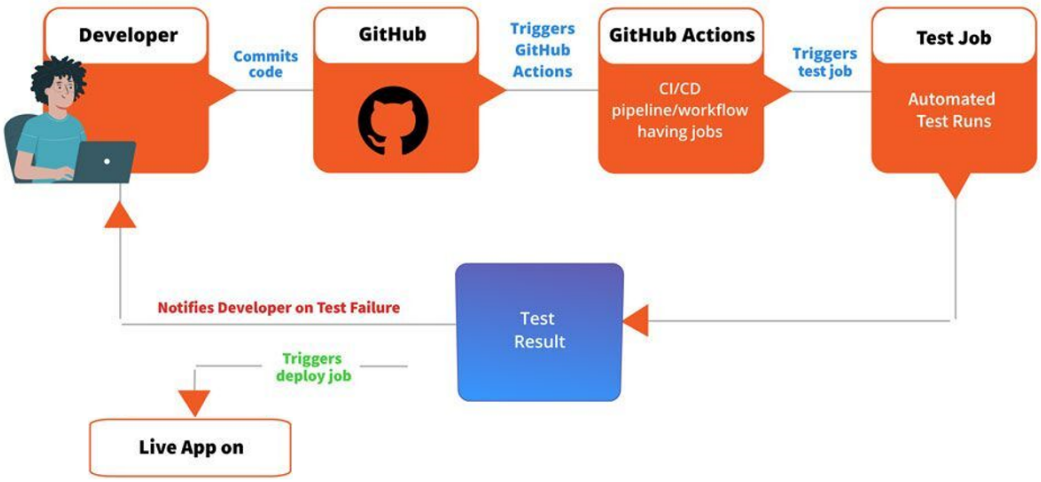
- The database access will be restricted to authenticated users.
- The database default PORT will be changed.
- TLS implementation.
- Encryption at rest is not considered since the application collects limited PII data.
- DB instance login will be restricted.

4.2 Implementation

The entire infrastructure will be provisioned as per the requirement. And the CI / CD will be handled using GitOps.

- Virtual machine B8ms is used to handle the data processing and API in the production environment.
- Virtual machine B4ms is used to handle the data pipeline
- Virtual machine B4ms is used for the backend in the development environment.
- The frontend deployment will be handled through Azure Storage Account Staic hosting and Cloudfront.
- Azure Storage Account is used to store raster and vector data
- Azure Database for PostgreSQL will be used as the database

4.3 CI / CD



The deployment of the apps to the infrastructure will be handled using the GitOps methodology. Github Actions will be involved in it.

Whenever a developer moves a feature for deployment by placing a merge request (MR) , Github CI pipeline will be triggered and it will run a set of pipelines.

The pipeline includes the

- **Dependency audit**: This step will check whether there are any vulnerabilities in the libraries we are using.
- **Build**: This step creates the build of the application and pushes to the server.

If any of the above steps fails , the whole process will be aborted.

Incase of production , the pipeline won't be triggered automatically, we need to trigger it manually.

4.4 Server Specification

The application will be hosted on Microsoft Azure and will also use other managed services from AWS

1. Virtual machine B4ms (4 vCPUs, 16 GB RAM) | 1 Nos - Dev Server

    Authentication method: .pem file
    Operating system: Ubuntu 18.04
    Inbound Ports: All traffic
    IP : Static IP

2. Virtual machine B4ms (4 vCPUs, 16 GB RAM) | 4 Nos - Data Pipeline

    Authentication method: .pem file
    Operating system: Ubuntu 20.04
    Inbound Ports: All traffic
    IP : Static IP

3. Virtual machine 1 B8ms (8 vCPUs, 32 GB RAM) | 1 Nos - Backend server

    Authentication method: .pem file
    Operating system: Ubuntu 18.04
    Inbound Ports: 443, 80
    IP: Static IP

4. IP Addresses

    2 Static IP for API server & dev server

5. Azure Monitor (Alternative for amazon cloudwatch)

Monitor web applications
Monitor Infrastructure
Based on various monitoring we can setup alerts

6. Notification Hubs (Alternative for amazon simple notification service SNS)

7.Azure Database for PostgreSQL

Version : 11
Tier : General purpose
Compute : Gen 5, 2 vCore
Storage : 30 GB
Additional Backup storage : 30 GB

Whitelist Backend server ip on this

8. Application Gateway - Web Application Firewall V2

Web application Firewall on Application Gateway.It helps to protect our web application or APIs against common web exploits and bots that may affect availability , compromise security, or consume excessive resources this will implement on both frontend and backend server

9. Content Delivery Network - Microsoft Classic

This can be used in Azure blob . by using its rule engine where we can configure URL rewrites which effectively matches patterns of the url and internally routes traffic elsewhere without affecting the client side URL

10.Storage Accounts

- Capacity 2 TB for storing layers data
  Type : Block Blob Storage
  Performance Tier: Standard
  Storage Account Type: General Purpose V2
  Access tier: HOT
  Redundancy: LRS

  Credentials needed to write data from the code
  connection string

storage account name
Container

- Capacity 1 GB for hosting front end app
  Type : Block Blob Storage
  Performance Tier: Standard
  Storage Account Type: General Purpose V2
  Access tier: HOT
  Redundancy: LRS

External Interface Requirements

4.5 User Interfaces

The user interface will be simple and consistent, using terminology commonly understood by the intended users of the system. The system will have a simple interface, consistent with industry standard interfaces, to eliminate the need for user training of infrequent users. For additional details see Appendix B-UI. User testing will be used to ensure the user interface is clear (simple, commonly understood vocabulary, intuitive to use without training), complete (users can perform all functions from the interface), and consistent (buttons and wording are the same throughout the system).

4.6 Software Interfaces

The system will use the standard OpenSource tools/libraries. This includes, but is not limited to, ReactJS, Python, Fast API, PostgreSQL, and Apache & NGINX server.

4.7 Communications Interfaces

The system will use open communication protocols. This includes, but is not limited to, HTTP protocol for communication with the web browser and the web server and TCP/IP network protocol with HTTP protocol.

5. Other Nonfunctional Requirements

5.1 Performance Requirements

Response time: The application shall show no visible deterioration in response time as the number of persons increases. Response times seen by end users for querying metadata should be on the order of a few seconds or less.

Loading speed: The application shall load as quickly as comparable productivity tools on whatever environment it is running in.

5.2 Safety Requirements

AWS is responsible for protecting the infrastructure that runs all of the services offered in the AWS Cloud. This infrastructure is composed of the hardware, software, networking, and facilities that run AWS Cloud services.

5.3 Security Requirements

The security requirements will have four primary components. They are authentication, confidentiality, integrity, and availability.

SCR-1: Authentication

Since it is an open source public platform, user authentication is not used. But all other security features such as captcha, encryption and blocking of bulk api requests will be enabled.

SCR-2: Confidentiality

Confidentiality security requirements from the client will be addressed.

SCR-3: Data Integrity

The integrity of data will be critical to its success as a product. The data used only from validated sources. Therefore, extensive data validation and review will be performed both before data are uploaded to the system and as part of the upload process. The system will need policy and procedures protecting the data from intentional or unintentional modifications, and to ensure accurate data are made available.

SCR-4: Availability

The fourth consideration for security requirements is availability. The system must be available to the intended audience 24 hours per day, 7 days a week with 99% availability and a tolerance of -5% (not less than 50% of working hours in any week). For this system, availability will be concerned with the reliability of the software and network components. Intentional “denial of service attacks” is not foreseen as a significant concern.

5.4 Software Quality Attributes

SQA-01: Portability

This database will be built for a particular system and may not be portable but results to queries will be portable between many environments.

SQA-02: Adaptability

Implementation of the application software/code and design of database structure should be flexible enough for the necessary change in the later phase.

SQA-03: Availability

Availability is defined here to mean the ability to use the system during its intended period of operation.

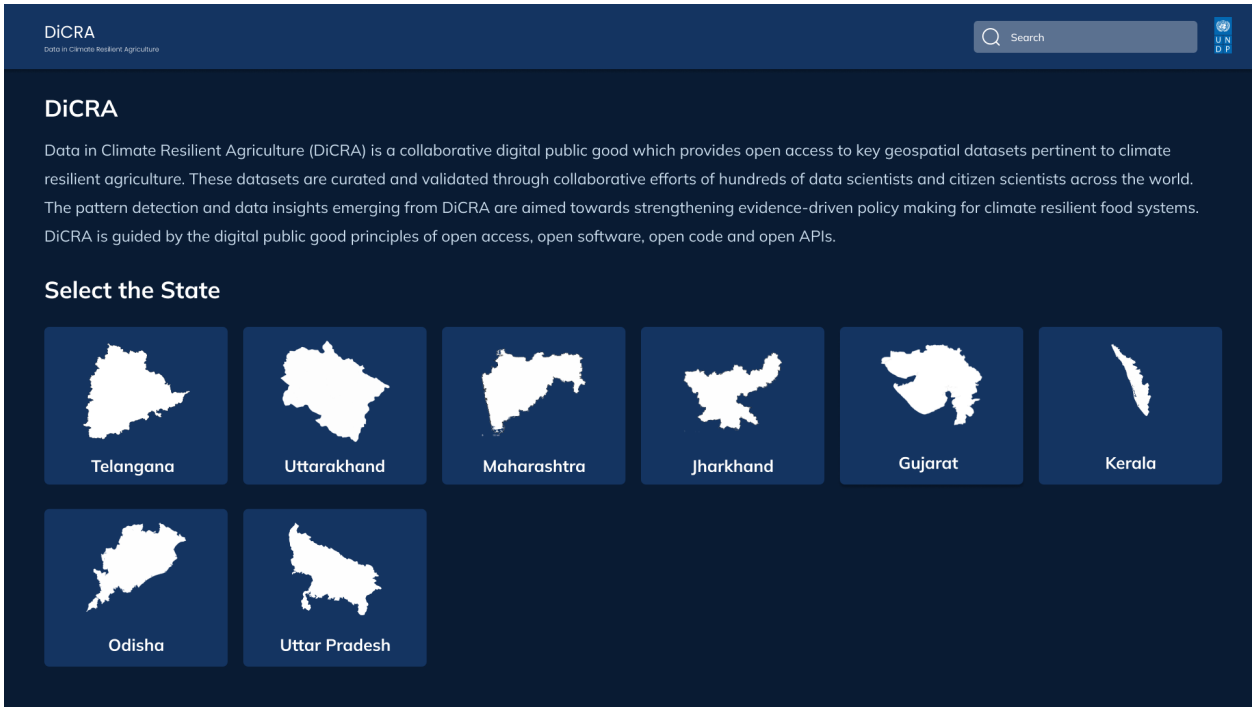
6. Appendix A: Acronyms

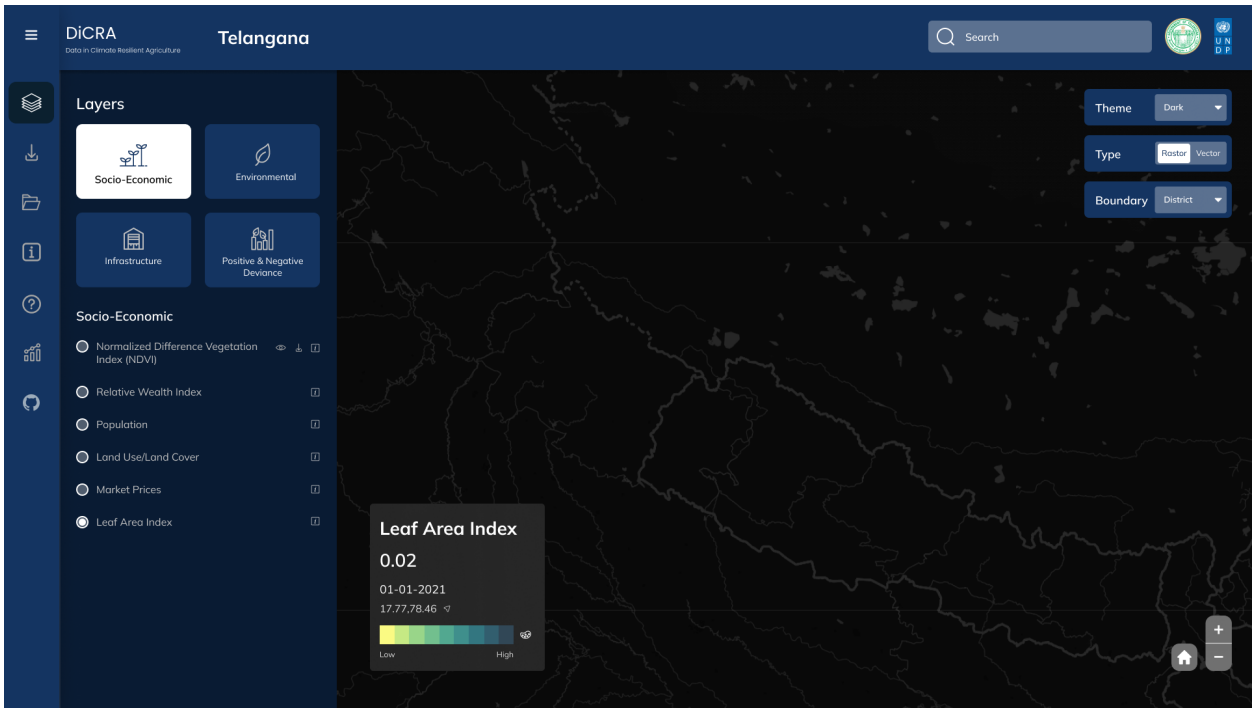
UNDP - United Nations Development Programme
MPL - mistEO Private Limited
GIS - Geographical Information System
UI - User Interface
HTML - HyperText Markup Language
HTTP - HyperText Transfer Protocol
RDS - Relational Database Service
WAF - Web Application Firewall
NDVI - Normalized difference vegetation index
AWS - Amazon Web Services
API - Application Programming Interface
CI - Continuous Integration
CD - Continuous Delivery/Continuous Deployment

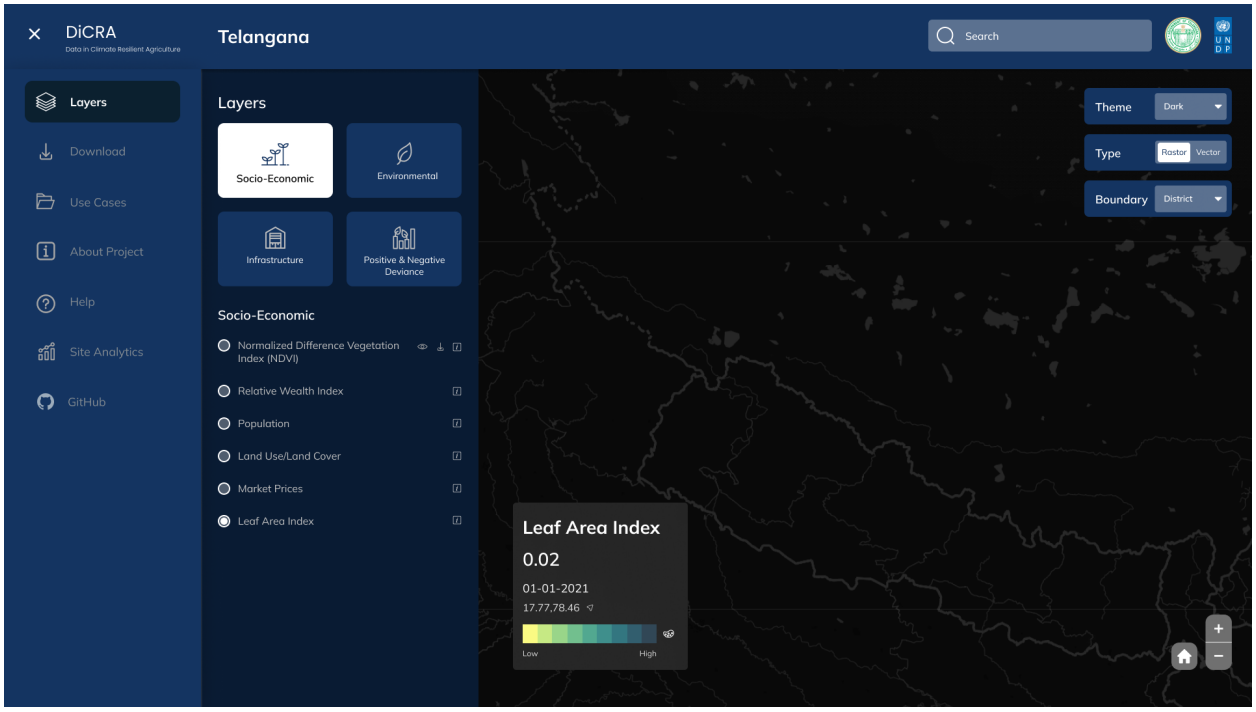
7. Appendix B: Datasets

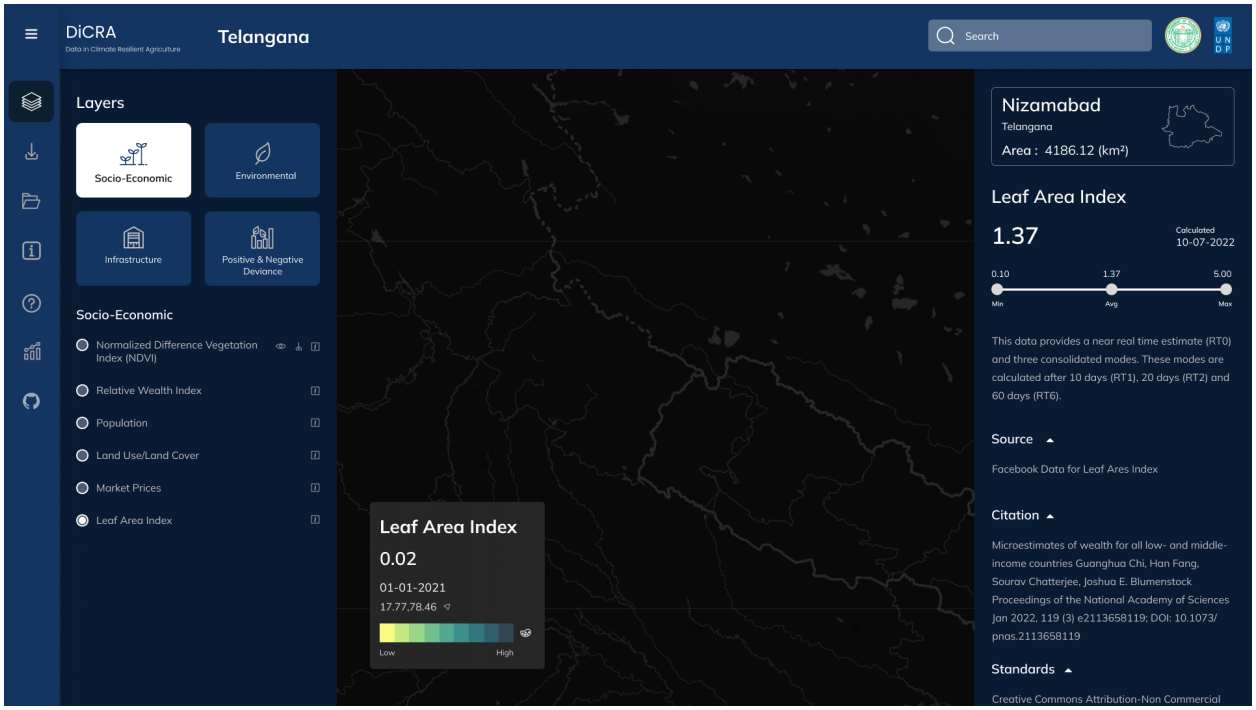
- Normalized difference vegetation index (NDVI)
- Normalized difference water index (NDWI)
- Leaf Area Index (LAI)
- Relative Wealth Index (RWI)
- Population
- Land Use / Land Cover (LULC)
- Soil Moisture
- Soil Organic Carbon
- Fire Events
- Market Price
- Weather
- Nitrogen Dioxide (NO2)
- Land Surface Temperature (LST)
- Particulate Matter (PM2.5)
- Warehouses Geolocation
- Positive and Negative Deviance

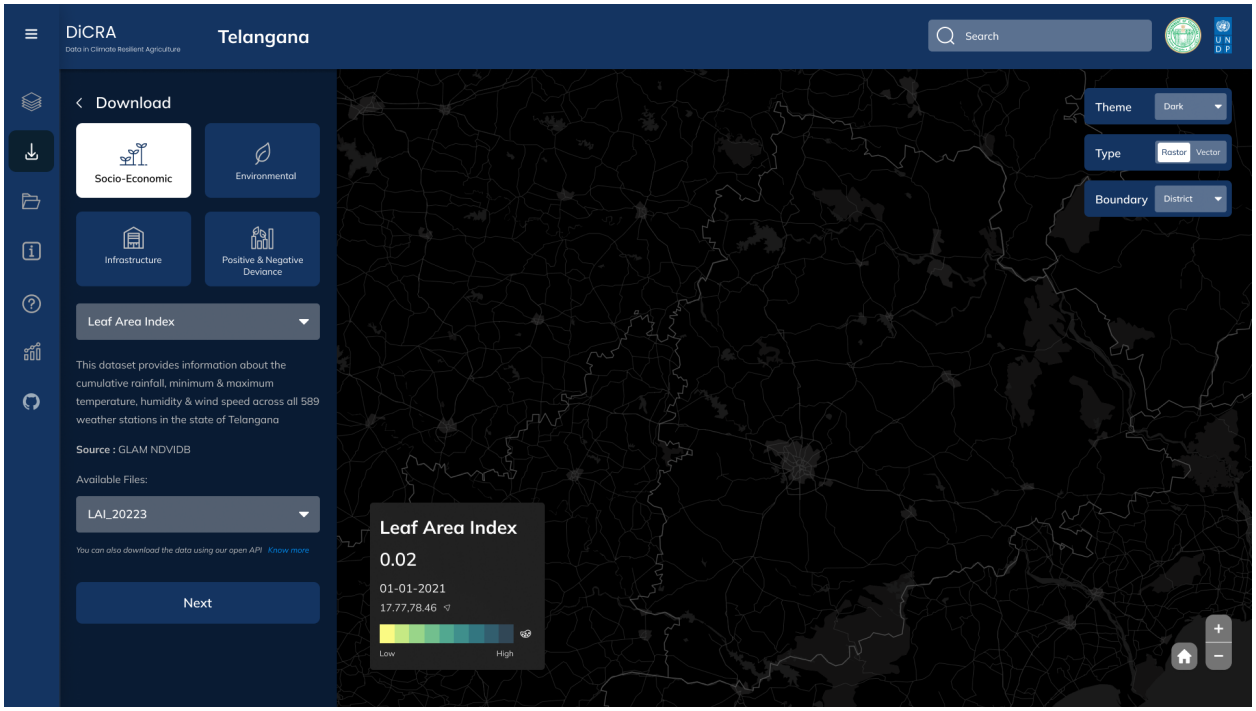
8. Appendix B: UI

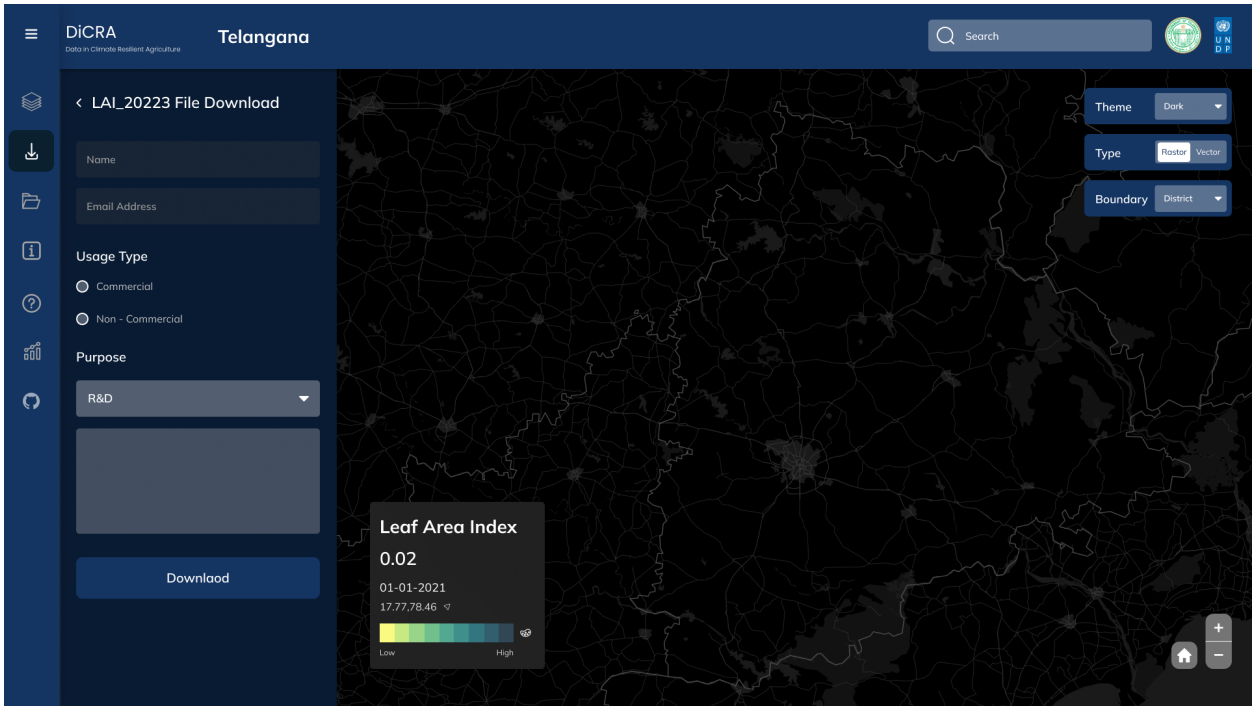


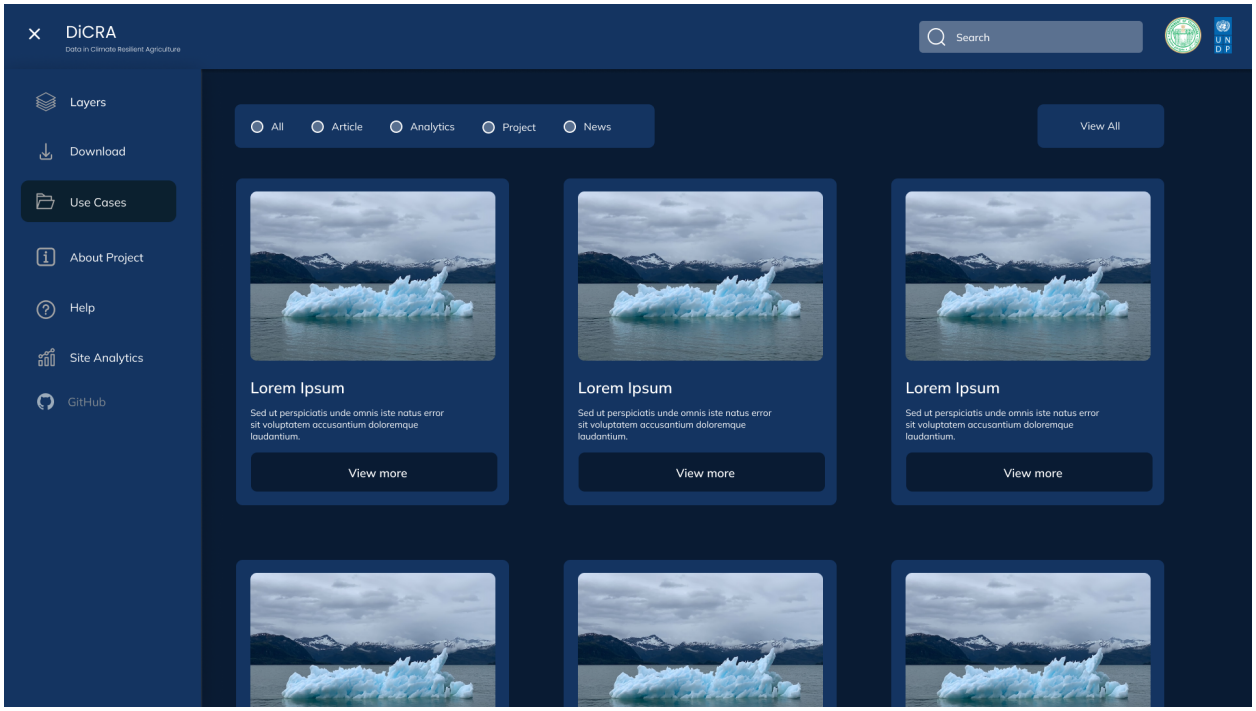

DiCRA
Data in Climate Resilient Agriculture
Telangana
Search
Layers
Socio-Economic
Environmental
Infrastructure
Positive & Negative Deviance
Socio-Economic
Normalized Difference Vegetation Index (NDVI)
Relative Wealth Index
Population
Land Use/Land Cover
Market Prices
Leaf Area Index
Theme
Dark
Type
Raster
Vector
Boundary
District
Leaf Area Index
0.02
01-01-2021
17.77,78.46
Low
High

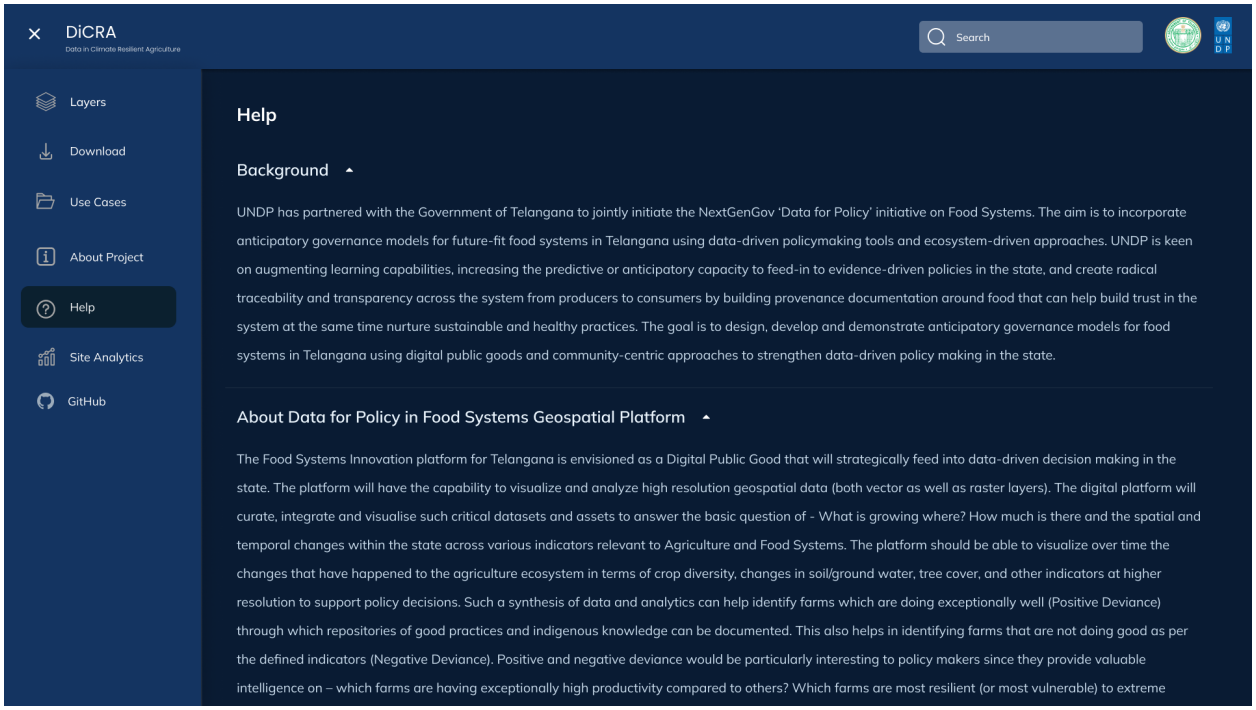

DiCRA
Data in Climate Resilient Agriculture
Telangana
Search
Layers
Download
Use Cases
About Project
Help
Site Analytics
GitHub
Layers
Socio-Economic
Environmental
Infrastructure
Positive & Negative Deviance
Socio-Economic
Normalized Difference Vegetation Index (NDVI)
Relative Wealth Index
Population
Land Use/Land Cover
Market Prices
Leaf Area Index
Theme
Dark
Type
Raster
Vector
Boundary
District
Leaf Area Index
0.02
01-01-2021
17.77,78.46
Low
High

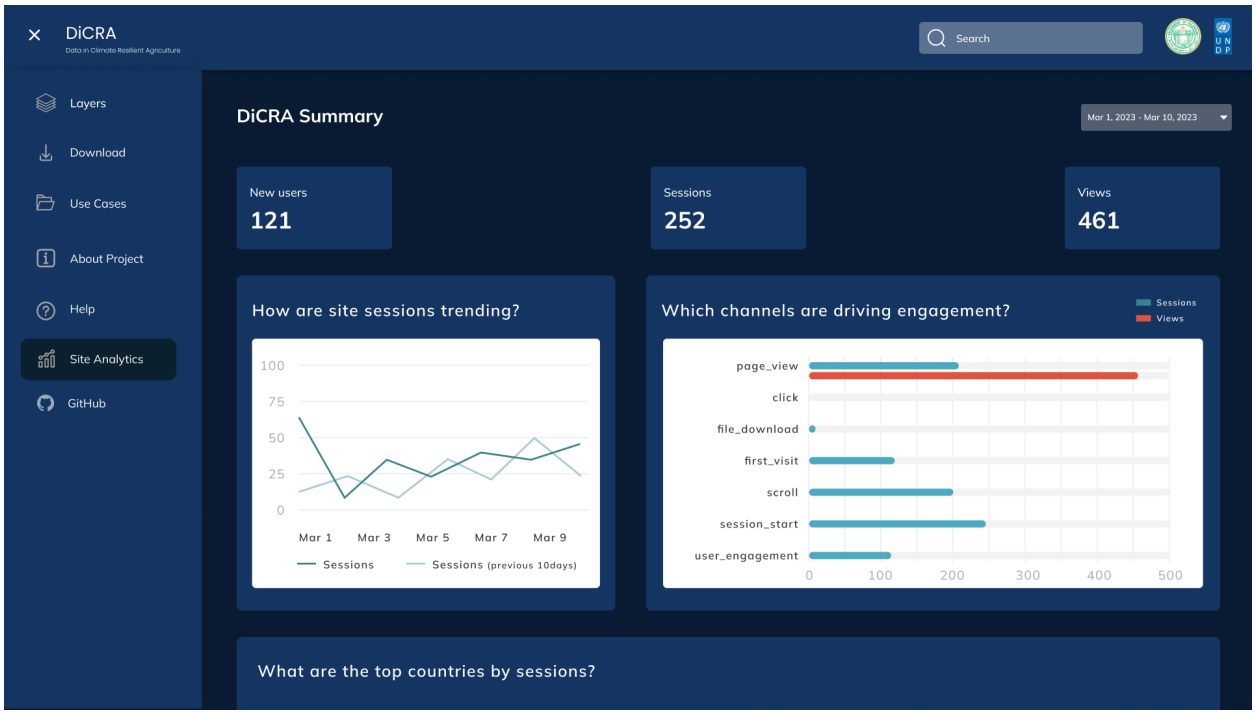

DiCRA
Data in Climate Resilient Agriculture
Telangana
Search
Layers
Socio-Economic
Environmental
Infrastructure
Positive & Negative Deviance
Socio-Economic
Normalized Difference Vegetation Index (NDVI)
Relative Wealth Index
Population
Land Use/Land Cover
Market Prices
Leaf Area Index
Leaf Area Index
0.02
01-01-2021
17.77,78.46
Low
High
Nizamabad
Telangana
Area : 4186.12 (km²)
Leaf Area Index
1.37
Calculated
10-07-2022
0.10
1.37
5.00
Min
Avg
Max
This data provides a near real time estimate (RT0) and three consolidated modes. These modes are calculated after 10 days (RT1), 20 days (RT2) and 60 days (RT6).
Source
Facebook Data for Leaf Ares Index
Citation
Microestimates of wealth for all low- and middle-income countries Guanghua Chi, Han Fang, Sourav Chatterjee, Joshua E. Blumenstock Proceedings of the National Academy of Sciences Jan 2022, 119 (3) e2113658119; DOI: 10.1073/pnas.2113658119
Standards
Creative Commons Attribution-Non Commercial

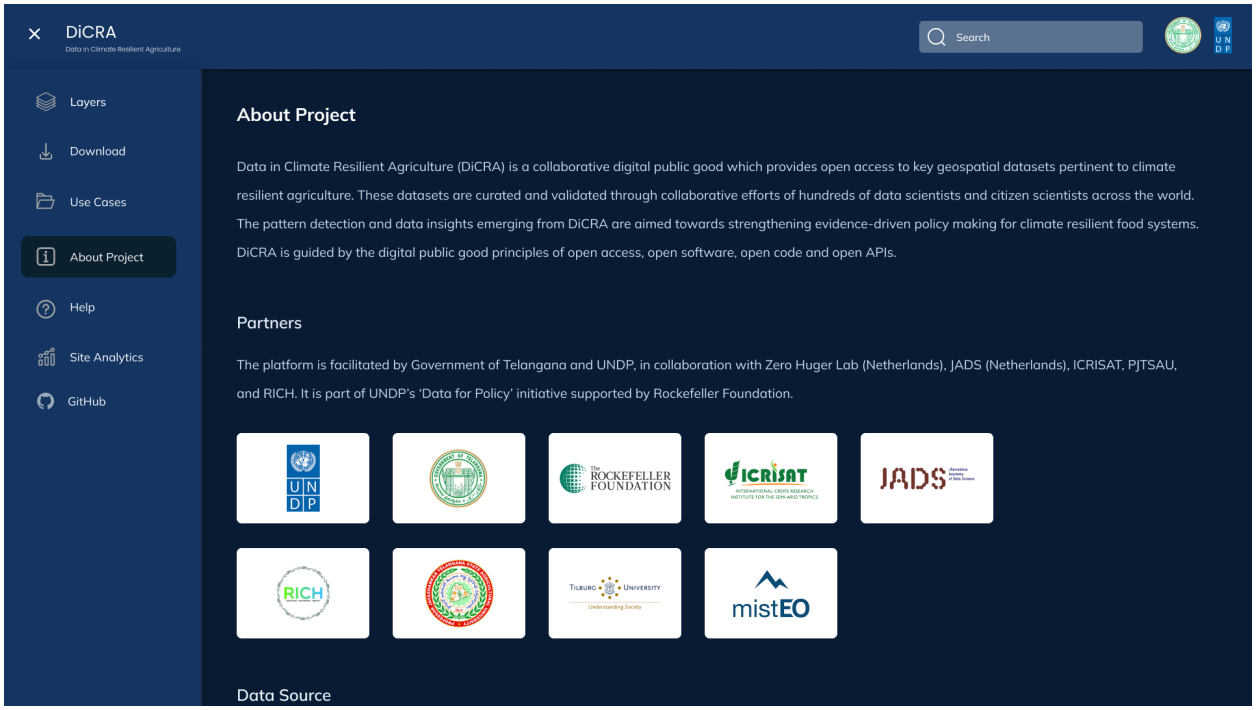

DiCRA
Data in Climate Resilient Agriculture
Telangana
Search
Download
Socio-Economic
Environmental
Infrastructure
Positive & Negative Deviance
Leaf Area Index
This dataset provides information about the cumulative rainfall, minimum & maximum temperature, humidity & wind speed across all 589 weather stations in the state of Telangana
Source : GLAM NDVIDB
Available Files:
LAI_20223
You can also download the data using our open API Know more
Next
Theme
Dark
Type
Raster
Vector
Boundary
District
Leaf Area Index
0.02
01-01-2021
17.77,78.46
Low
High

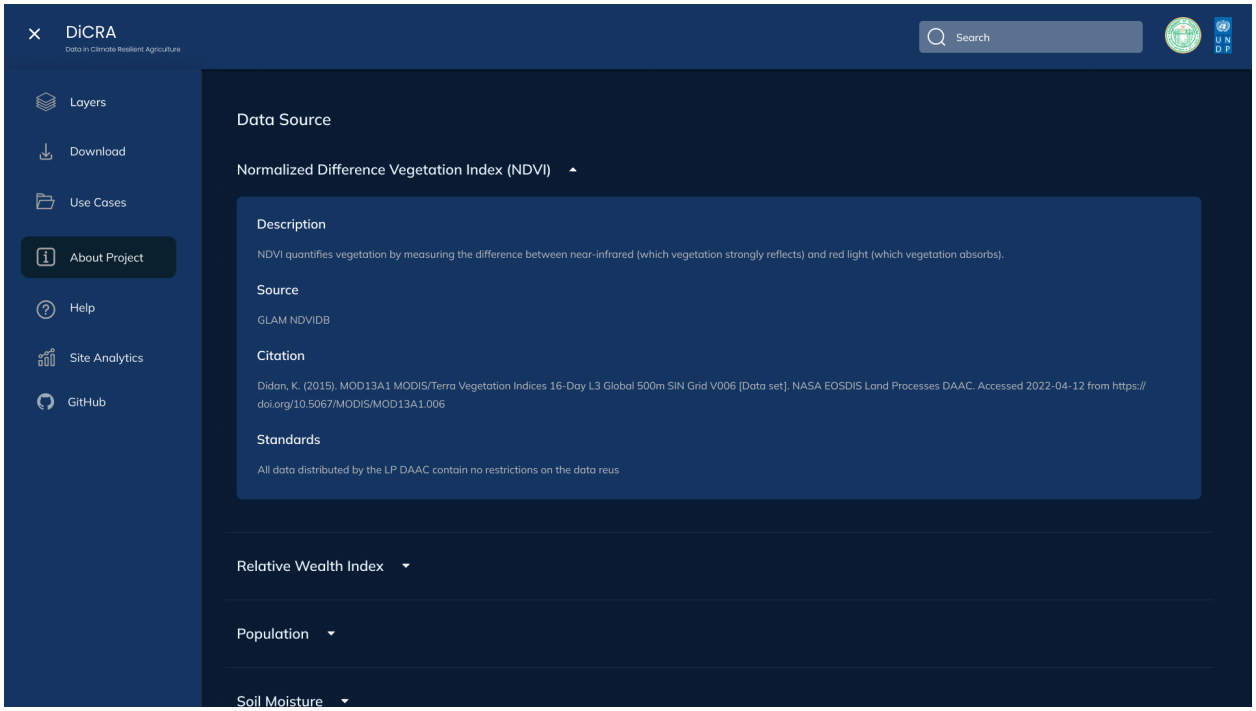

DiCRA
Data in Climate Resilient Agriculture
Telangana
Search
LAI_20223 File Download
Name
Email Address
Usage Type
Commercial
Non - Commercial
Purpose
R&D
Downlaod
Theme
Dark
Type
Raster
Vector
Boundary
District
Leaf Area Index
0.02
01-01-2021
17.77,78.46
Low
High


DiCRA
Data in Climate Resilient Agriculture
Search
Layers
Download
Use Cases
About Project
Help
Site Analytics
GitHub
All
Article
Analytics
Project
News
View All
Lorem Ipsum
Sed ut perspiciatis unde omnis iste natus error sit voluptatem accusantium doloremque laudantium.
View more
Lorem Ipsum
Sed ut perspiciatis unde omnis iste natus error sit voluptatem accusantium doloremque laudantium.
View more
Lorem Ipsum
Sed ut perspiciatis unde omnis iste natus error sit voluptatem accusantium doloremque laudantium.
View more


DiCRA
Data in Climate Resilient Agriculture
Search
Layers
Download
Use Cases
About Project
Help
Site Analytics
GitHub
Help
Background
UNDP has partnered with the Government of Telangana to jointly initiate the NextGenGov 'Data for Policy' initiative on Food Systems. The aim is to incorporate anticipatory governance models for future-fit food systems in Telangana using data-driven policymaking tools and ecosystem-driven approaches. UNDP is keen on augmenting learning capabilities, increasing the predictive or anticipatory capacity to feed-in to evidence-driven policies in the state, and create radical traceability and transparency across the system from producers to consumers by building provenance documentation around food that can help build trust in the system at the same time nurture sustainable and healthy practices. The goal is to design, develop and demonstrate anticipatory governance models for food systems in Telangana using digital public goods and community-centric approaches to strengthen data-driven policy making in the state.
About Data for Policy in Food Systems Geospatial Platform
The Food Systems Innovation platform for Telangana is envisioned as a Digital Public Good that will strategically feed into data-driven decision making in the state. The platform will have the capability to visualize and analyze high resolution geospatial data (both vector as well as raster layers). The digital platform will curate, integrate and visualise such critical datasets and assets to answer the basic question of - What is growing where? How much is there and the spatial and temporal changes within the state across various indicators relevant to Agriculture and Food Systems. The platform should be able to visualize over time the changes that have happened to the agriculture ecosystem in terms of crop diversity, changes in soil/ground water, tree cover, and other indicators at higher resolution to support policy decisions. Such a synthesis of data and analytics can help identify farms which are doing exceptionally well (Positive Deviance) through which repositories of good practices and indigenous knowledge can be documented. This also helps in identifying farms that are not doing good as per the defined indicators (Negative Deviance). Positive and negative deviance would be particularly interesting to policy makers since they provide valuable intelligence on – which farms are having exceptionally high productivity compared to others? Which farms are most resilient (or most vulnerable) to extreme


DiCRA
Data in Climate Resilient Agriculture
Search
Layers
Download
Use Cases
About Project
Help
Site Analytics
GitHub
DiCRA Summary
Mar 1, 2023 - Mar 10, 2023
New users
121
Sessions
252
Views
461
How are site sessions trending?
100
75
50
25
0
Mar 1
Mar 3
Mar 5
Mar 7
Mar 9
Sessions
Sessions (previous 10days)
Which channels are driving engagement?
Sessions
Views
page_view
click
file_download
first_visit
scroll
session_start
user_engagement
0
100
200
300
400
500
What are the top countries by sessions?


DiCRA
Data in Climate Resilient Agriculture
Search
Layers
Download
Use Cases
About Project
Help
Site Analytics
GitHub
About Project
Data in Climate Resilient Agriculture (DiCRA) is a collaborative digital public good which provides open access to key geospatial datasets pertinent to climate resilient agriculture. These datasets are curated and validated through collaborative efforts of hundreds of data scientists and citizen scientists across the world. The pattern detection and data insights emerging from DiCRA are aimed towards strengthening evidence-driven policy making for climate resilient food systems. DiCRA is guided by the digital public good principles of open access, open software, open code and open APIs.
Partners
The platform is facilitated by Government of Telangana and UNDP, in collaboration with Zero Huger Lab (Netherlands), JADS (Netherlands), ICRISAT, PJTSAU, and RICH. It is part of UNDP's 'Data for Policy' initiative supported by Rockefeller Foundation.
Data Source


DiCRA
Data in Climate Resilient Agriculture
Search
Layers
Download
Use Cases
About Project
Help
Site Analytics
GitHub
Data Source
Normalized Difference Vegetation Index (NDVI)
Description
NDVI quantifies vegetation by measuring the difference between near-infrared (which vegetation strongly reflects) and red light (which vegetation absorbs).
Source
GLAM NDVIDB
Citation
Didan, K. (2015). MOD13A1 MODIS/Terra Vegetation Indices 16-Day L3 Global 500m SIN Grid V006 [Data set]. NASA EOSDIS Land Processes DAAC. Accessed 2022-04-12 from https://doi.org/10.5067/MODIS/MOD13A1.006
Standards
All data distributed by the LP DAAC contain no restrictions on the data reus
Relative Wealth Index
Population
Soil Moisture