Documentation:

NASA'S Europa Challenge is a unique platform to build apps which can benefit the society in various ways by using various technologies including NASA's open source virtual globe technology, WebWorldWind.

We also thank GODAN for giving us the opportunity to help the farmers throughout the world. This initiative will surely add to welfare of the society.

Description of Idea:

In FarmsNtech Webapp ,we are predicting the weather of 15 days with the help of API and Algorithms .We are using these predictions to tell suitability of weather for the crops and notify farmers about the same.

A farmer can register in our website ,choose a crop and start receiving the messages about the various weather conditions related to stages of crop.

The messages will consist of topics like "When to sow seeds? ", "How to manage the vegetative phase?", "How much water do they need from outer sources in case of inadequate rainfall? ".

A non-registered user can see everything on our website but will not get messages.

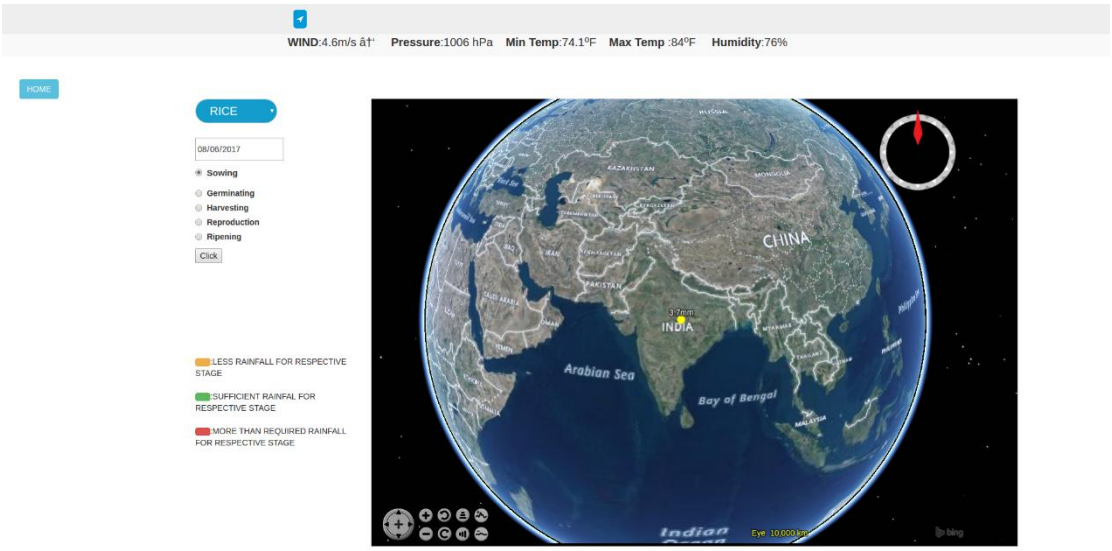
Technology Stack:

- Nasa WebWorldWind
- R
- HTML,CSS and its frameworks
- Javascript
- JSP
- Java

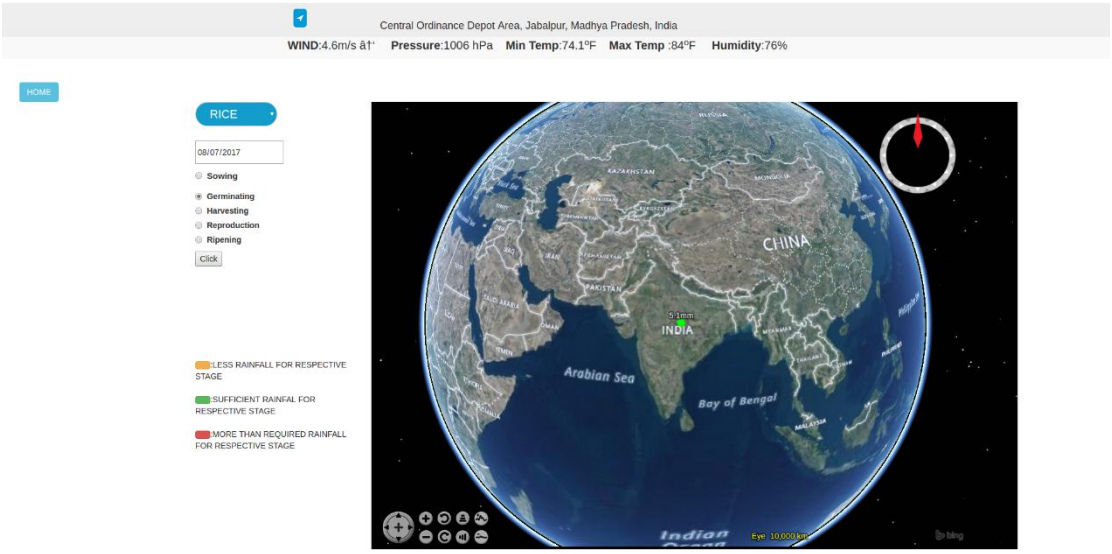
Using World Wind Web

We are providing weather prediction analysis of a particular Crop using World Wind Web.

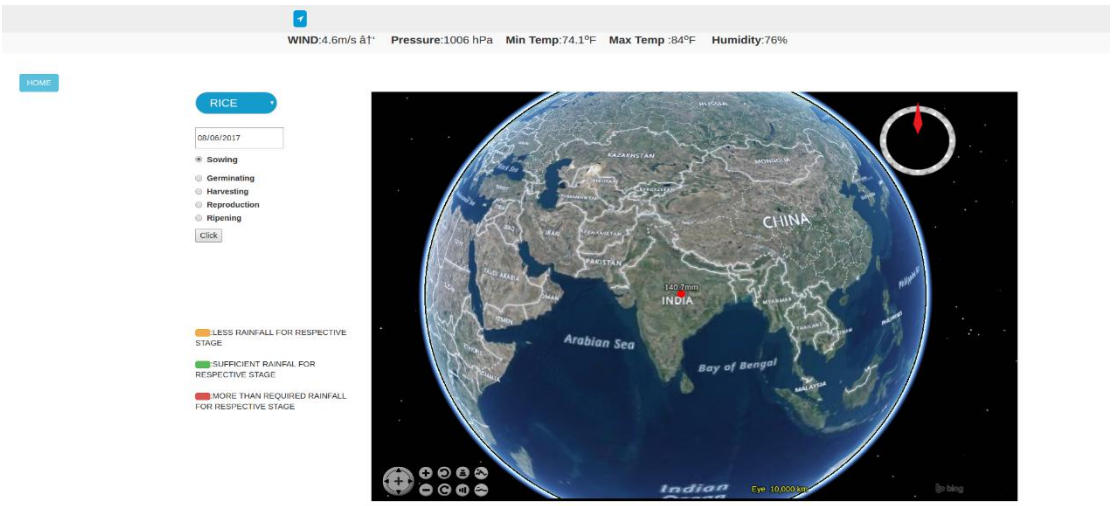
One can choose a crop and date, then he can see whether that day is suitable for performing that operation on crop or not based on a particular location.We are calling our algorithm to predict suitability of weather for a crop in Nasa world wind JS and the, preforming the required operations.



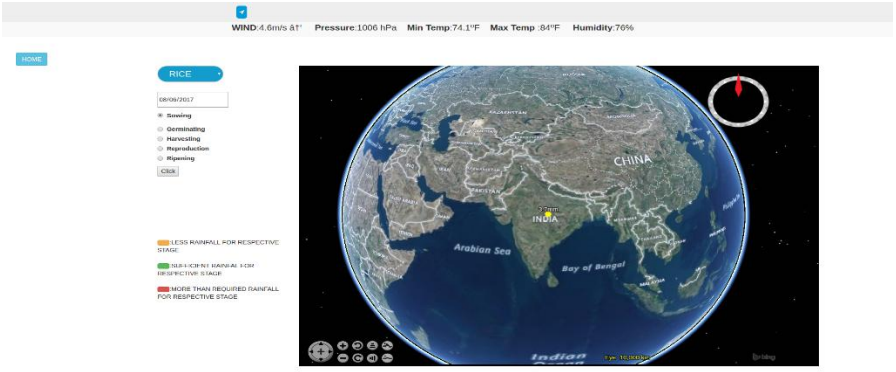
Yellow dot indicates that Rainfall on chosen date is less than required rainfall for selected stage of Rice crop.

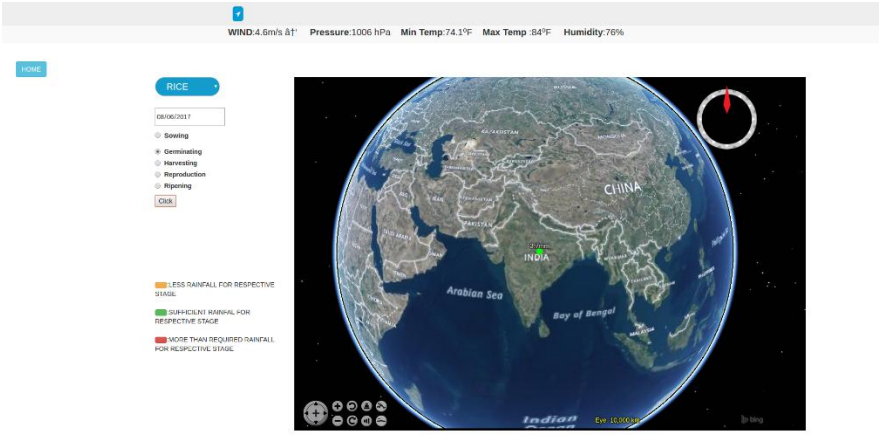


Green dot indicates that adequate weather conditions in this region for a given stage and crop.



Red dot indicates that the rainfall is very high in this region for a given stage and crop.





For a particular day, rainfall of a region will be same. That's why yellow and green buttons are showing same value. The different colour is due to their difference in ideal conditions.That's why if a certain temperature is suitable for one stage, other might not satisfy that temperature.

Algorithm for weather prediction

Our aim is to predict the weather information of next 15 days. We have used and extended Sliding Window Algorithm that predicts the weather data for the next day based on the last 1 year data.

Step 1. Take matrix "CD" of last seven days for current year's data of size 7×4.
Step 2. Take matrix "PD" of fourteen days for previous year's data of size 14×4.
Step 3. Make 8 sliding windows of size 7×4 each from the matrix "PD" as $W_1, W_2, W_3, \ldots, W_8$
Step 4. Compute the Euclidean distance of each sliding window with the matrix "CD" as $ED_1, ED_2, ED_3, \ldots, ED_8$
Step 5. Select matrix W_i as
W_i = Correponding_Matrix (Min.(ED_i))
$\forall i \in [1, 8]$
Step 6. For $k = 1$ to n
(i) For WC_k compute the variation vector for the matrix "CD" of size 6×1 as "VC".
(ii) For WC_k compute the variation vector for the matrix "PD" of size 6×1 as "VP".
(iii) $Mean_1$ = Mean (VC)
(iv) $Mean_2$ = Mean (VP)
(v) Predicted Variation "V" = $(Mean_1 + Mean_2)/2$
(vi) Add "V" to the previous day's weather condition in consideration to get the predicted condition.
Step 7. End

Fig. Sliding Window algorithm pic.

But as our requirement is to predict the weather data mainly minimum temperature ,maximum temperature, rainfall and humidity for the next 15 days, we have made some changes to the algorithm and implemented it with weather data of last 5 years for better accuracy in R language.

According to the algorithm, there will be sequence in the current year's weather pattern which is almost identical to some sequence in the weather pattern of the last year. To find this sequence we are using the Euclidean distance method. If the Euclidean distance of two matrices is minimum, then we can say that those two matrices are identical. So, we can use that sequence in the previous year to predict the weather of the current year.

```
x=1
edlist = list()
for(i in 1:8){
  y=x+27
  mdat2 <- matrix(c(res1[x:y]),nrow=7, ncol=4, byrow = TRUE)
  library(pdist)
  dists <- pdist(t(mdat), t(mdat2))
  A = as.matrix(dists)
  z = det(A)
  edlist = c(edlist,z)
  x=x+4
}

```

Fig. Packages used in R

We have made separate algorithms for last 5 years. i.e one prediction set of 15 days per algorithm, per year. Then we have made another algorithm in which we have taken 40% probability of happening last year's prediction, 25% probability of happening prediction of 2015, 15% probability of happening prediction of 2014 and 10% probability each of prediction of 2013,2012 years.

```
datalist1 = list()

for(i1 in 1:15){
  x = (((0.4)*dat1[i1,2])+((0.25)*dat2[i1,2])+((0.15)*dat3[i1,2])+((0.1)*dat4[i1,2])+((0.1)*dat5[i1,2]))
  y = (((0.4)*dat1[i1,3])+((0.25)*dat2[i1,3])+((0.15)*dat3[i1,3])+((0.1)*dat4[i1,3])+((0.1)*dat5[i1,3]))
  z = (((0.4)*dat1[i1,4])+((0.25)*dat2[i1,4])+((0.15)*dat3[i1,4])+((0.1)*dat4[i1,4])+((0.1)*dat5[i1,4]))
  w = (((0.4)*dat1[i1,5])+((0.25)*dat2[i1,5])+((0.15)*dat3[i1,5])+((0.1)*dat4[i1,5])+((0.1)*dat5[i1,5]))

  datafinal = data.frame(x,y,z,w)
  datalist1[[i1]] <- datafinal

}

```

Fig. Implementing 5 years of data analysis

Prediction Algorithm for suitability of Weather for Crop:

By using the above prediction we have written another algorithm which predicts suitable days for different stages in farming for a particular crop.. Ex: if we want to sow seeds now there should not be heavy rainfall for the next 7-10 days and temperature should also be in ideal range. This algorithm will predict the conditions according to date and stage and will display according to the picture below.

As the data (sample.csv) which the algo is using, needs to be updated every day with the current date weather,we have used dark sky API for current weather information and a scheduler has been scheduled that will update the csv file of history data with current weather data.

For historical weather data, we have used weather underground API.

For location,we used Google apis.

As the algorithm is implemented in R and we need to connect it with Java web project ,we have included Rserve.jar, Rengine.jar and JRI.jar in tomcat lib directory.

Then, run the library(Rserve) ,then execute the Rserve() cmd.This starts R server which will let you you use R written code in Java.

Rainfall and Wind Speed Graph

The rainfall and wind speed are plotted simultaneously where x-axis shows the dates and y-axis shows the levels of rainfall and wind speed. The graphs change on the basis of the current weather conditions.