A blockchain secured framework for enhanced crop protection

by

- Ayush
- Naman
- Evin

Why this project?

- Monoculture crops are prone to pests and harmful insects
- Pests reduce around 42 percent yield in the world
- Crop protection is important to satisfy high public demand
- Crop protection also implies using pesticides in the right proportion

cont...

- Climatic Conditions also affect crop productivity
- IPM technique are employed to reduce pests
- Agronomic practices divert pests attack

> It is imperative for the solutions to reach farmers across the world

CONCEPT

A blockchain based framework for crowdsourcing the data generated by farmer community to help in crop management

Key Ideas

- Efficient and tested solution for the farmers in need
- Incentives for providing solution
- Security of blockchain
- Freedom to choose solutions
- Neural Network facility from pest prediction

Farmer’s option

Taking help of ML node

- What if farmer do not know the disease
- Capture and upload image of crop
- ML node will identify the disease
- Pay a little incentive for help

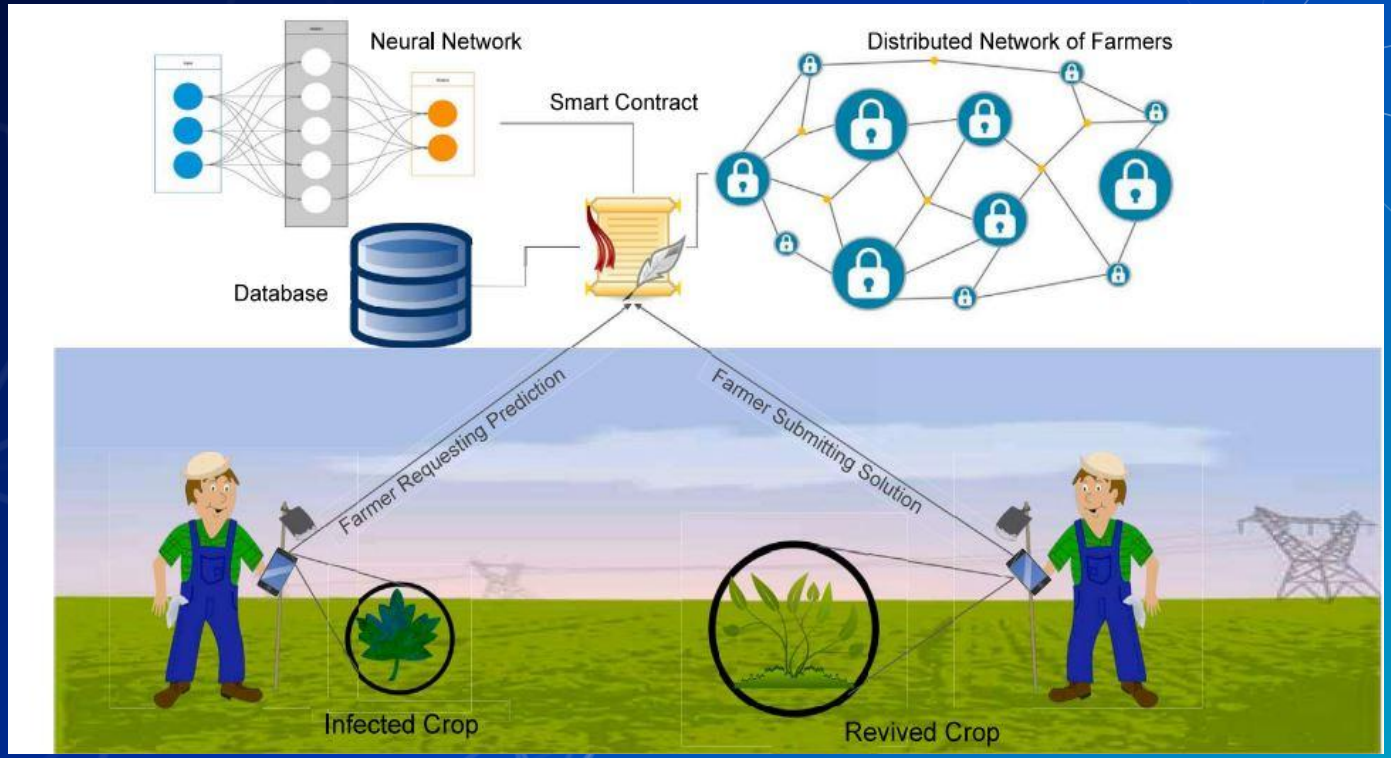
Finding a solution

- Enter disease
- Get the list of all available solutions
- Select a solution from the list
- Pay incentive to access full solution

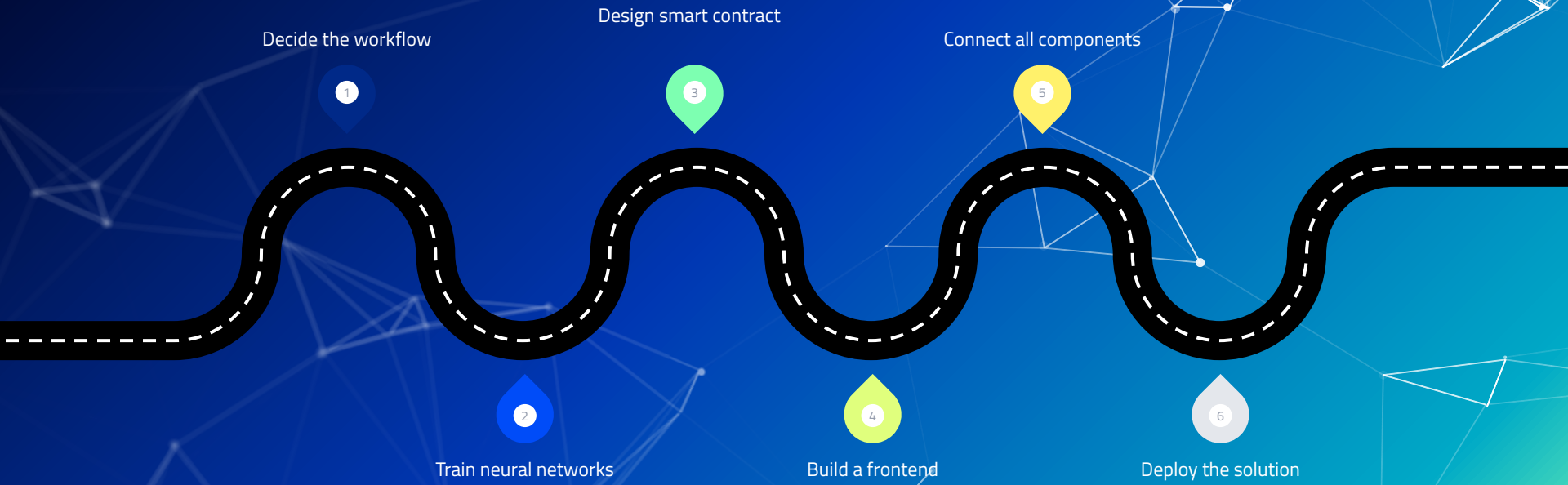
Uploading a solution

- In case someone invented a solution
- Upload the solution
- Get incentives when someone access it

Overview


Neural Network
Distributed Network of Farmers
Smart Contract
Database
Farmer Requesting Prediction
Farmer Submitting Solution
Infected Crop
Revived Crop

ROADMAP


Decide the workflow
1
Train neural networks
2
Design smart contract
3
Build a frontend
4
Connect all components
5
Deploy the solution
6

Training Neural Networks

- Dataset: Pest dataset with 9 classes, 350 image per class
- Convolution model with relu activation
- More than 652K parameters
- RMSProp optimizer with learning rate 0.001
- Test accuracy of more than 90%

Database

Solution ID	Disease ID	Infected Crop	Revived Crop	Solution	Account Address
					

Smart Contract (pseudo)

```
function pay_ml() payable public{
    MLnode.send(0.1);
}
```

```
function pay_for_solution() payable public{
    require(msg.value==10**18); //1 ether to pay directly for solution
}
```

```
function giveIncentive(address payable one) public payable returns(address){
    one.send(4*10**17); //0.4 ether
}
```

Future Scope

- Upvote and Downvote Solution
- Incentive refund in case of wrong ML prediction
- Checking uploaded solutions
- Launching solution on Android platform
- Crop selling facility

Conclusion

- The motivation of the users
- Privacy concerns connected to the service
- Increase in Crop production
- High number of participants due to incentivization