**Health Insurance Incentive Smart Contract**

**Blockchain technology used:** Solidity smart contracts on Ethereum blockchain.

**Use Case:** A health insurance company has to provide monthly incentive to its patients based on their daily number of footsteps. Therefore every end of the day health insurance provider is going to record the footstep details of its patients in the system. Assume footsteps are being recorded by some IOT means & then getting fed to our decentralized system at the end of day. Patients are going to store their 4 ether in contract in case if they are penalized (at the end of the month), contract should have sufficient amount to settle the penalty amount. In case they are not penalized, they will get incentive & their stored penalty amount (4 ether) back to their account at the end of the month. Insurance provider can provide this service to any number of patients.

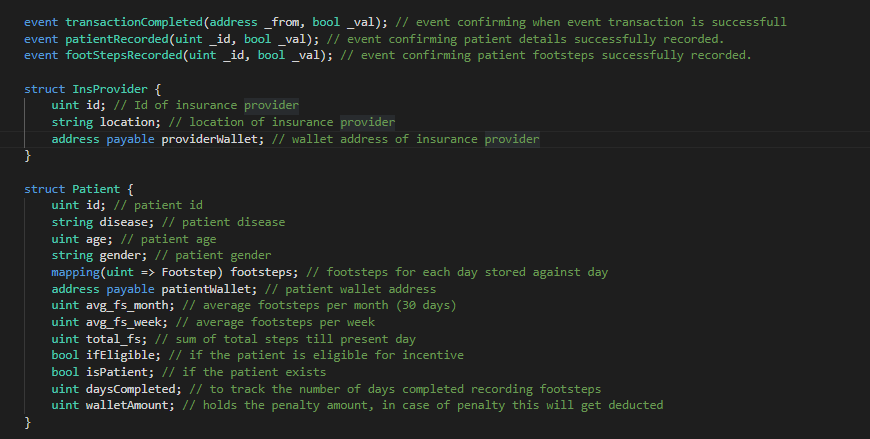
**Assumptions:**

1. Footsteps are being recorded by some IOT means & fed daily to the system at end of the day by insurance provider.
2. Assuming 30 days for each month.
3. Assuming 5000 steps as weekly threshold.

**Note –** Given Smart Contract can be tested locally using Ganache & truffle framework or on Remix IDE.

**Walkthrough:**

Here is our defined data structure:

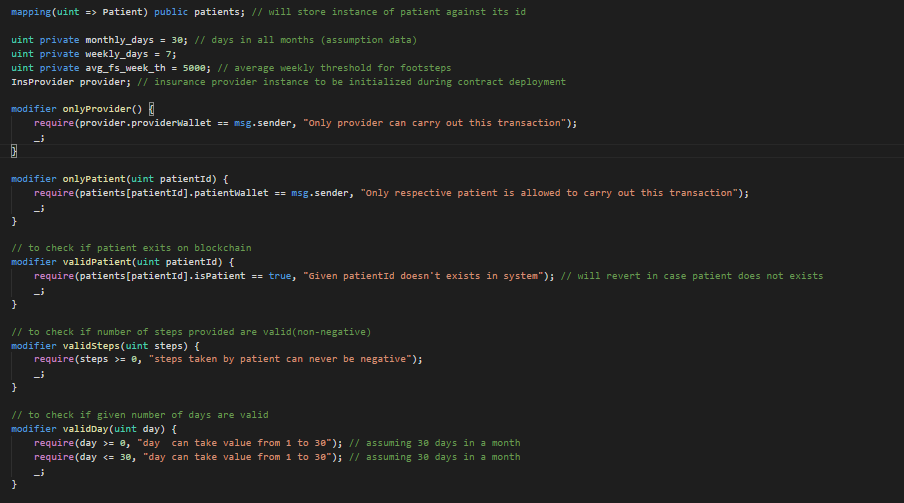


We will have various events confirming the transactions on blockchain.

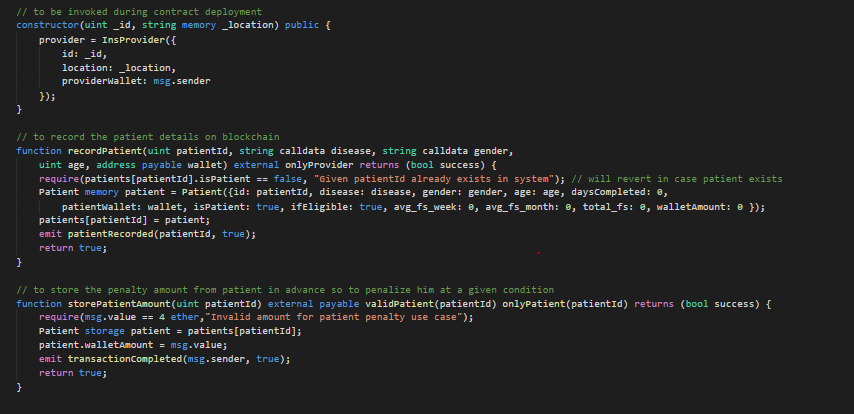
We will be maintaining struct for provider, patient & footsteps. Their address details are to be stored as part of the struct instance. Note that we are tracking various important parameters inside Patient struct such as avg\_fs\_month, avg\_fs\_week, total\_fs, ifEligible, isPatient, daysCompleted & walletAmount.

Insurance provider will be able to store the patient details on contract using the **Patient** struct.

Insurance provider will get initialized during Contract deployment when the constructor gets invoked.

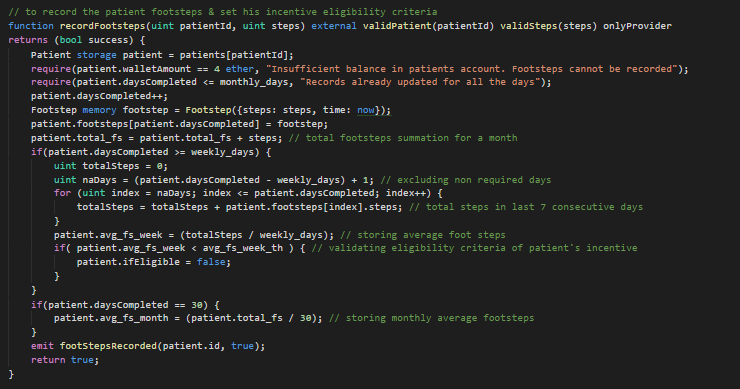


We also have mapping to store patient’s record against their id. Various modifiers have been defined to isolate & validate the various functionalities provided by the smart contract.

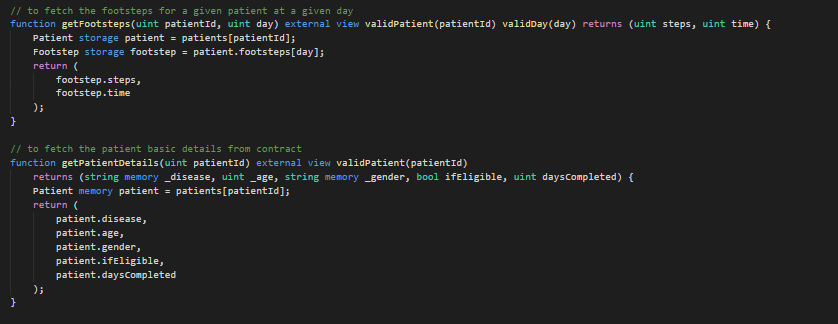


At the time of contract deployment on network, provider details to be initialized as constructor will get invoked.

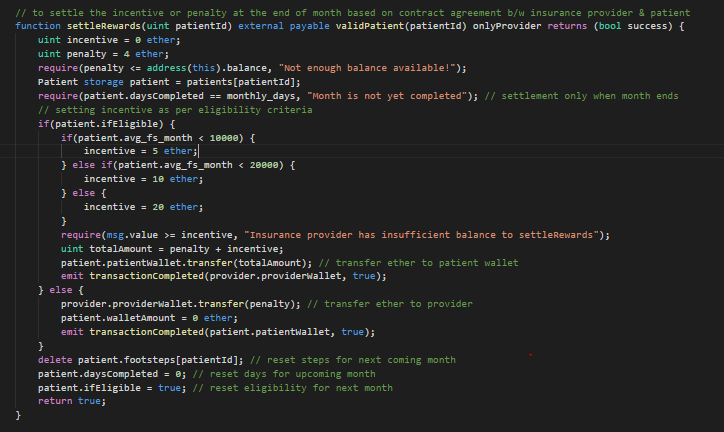
Provider will be able to call **recordPatient** method to record the patient details on blockchain. Each patient record will be an instance of Patient struct stored against patient Id in **patients** mapping. Before recording the recorded patient footsteps, we need to get the 4 ether penalty amount from patient’s wallet so that at the time of settlement, patient can be penalized if required. **storePatientAmount** method to be called by patient himself with penalty amount, passing their id as input argument. This will store penalty amount at contract address & will allow provider to record the patient’s footsteps.



**recordFootsteps** method to be called by provider with patientId & their respective footsteps count as input argument. We will validate first if patient has stored his penalty amount & also if all days records have not been stored. We will then store the footsteps records with current block timestamp. We will also sum up the total footsteps by combining records of previous days. If the record being fed is after 7 days we will check the average number of footsteps in last 7 days & will mark patient as ineligible accordingly. If the record being fed is for month’s last day we will also set the monthly average for patient.



Anyone can fetch the footstep & patient details from blockchain using methods: **getFootsteps** & **getPatientDetails**.



Provider can call **settleRewards** at the end of the month.

Here we will validate sufficient contract balance & if patient footsteps have been recorded for the complete month or not. If patient is eligible we will decide the incentive based on the given slab & will transfer him incentive + his stored penalty amount in contract. If patient is not eligible we will transfer the penalty amount from contract to Provider. Note that this step we kept at the end of month because we don’t want provider to penalize patient more than once in a month. We will also reset the patient footstep details & eligibility flag for the next month.

**Demo on UI:**

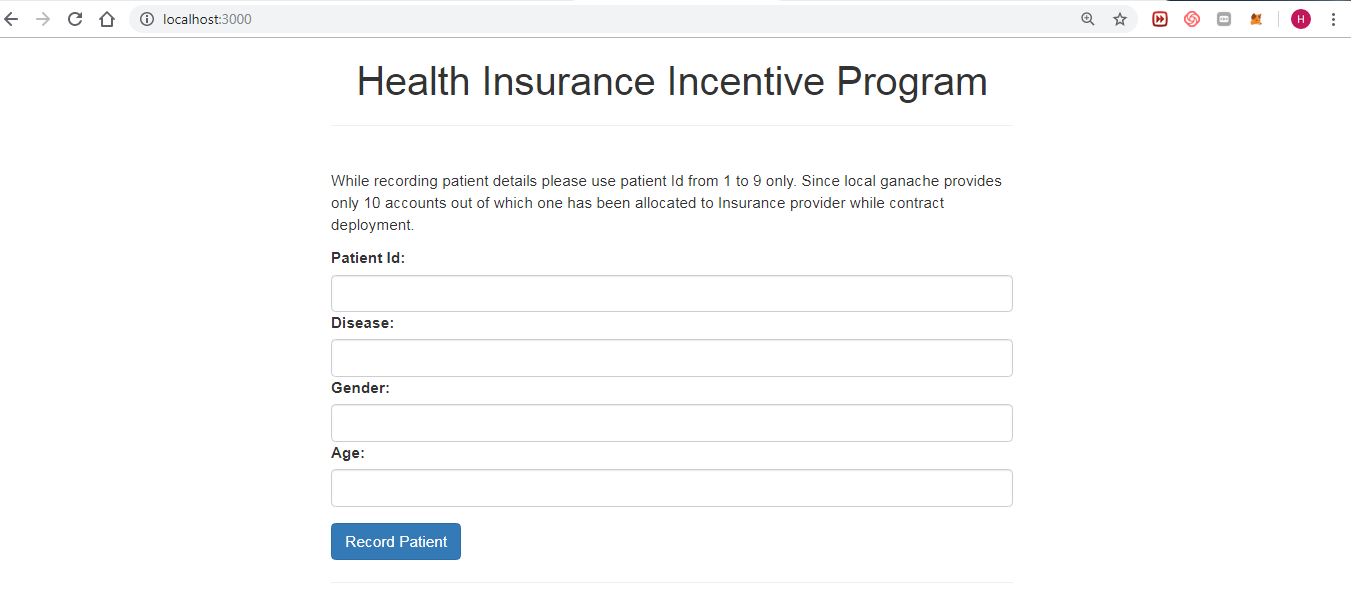
1. Install Truffle & Ganache on your machine.
2. Start the local Ganache on your machine at port 7545.
3. Ensure metamask extension is installed on your chrome browser. Connect metamask to <http://localhost:7545>. And import various accounts from Ganache to metamask.
4. Extract the source code in a specific folder.
5. Run the command ‘npm i’ in the root directory.
6. Run the command ‘truffle migrate --reset’. This will provide a build folder with deployed contract ABI.
7. After migration is complete, run command ‘npm run dev’.
8. Open the address: <http://localhost:3000> on chrome browser.

Please note that metamask is necessary to run the demo project.

Also metamask should be connected to local Ganache blockchain.

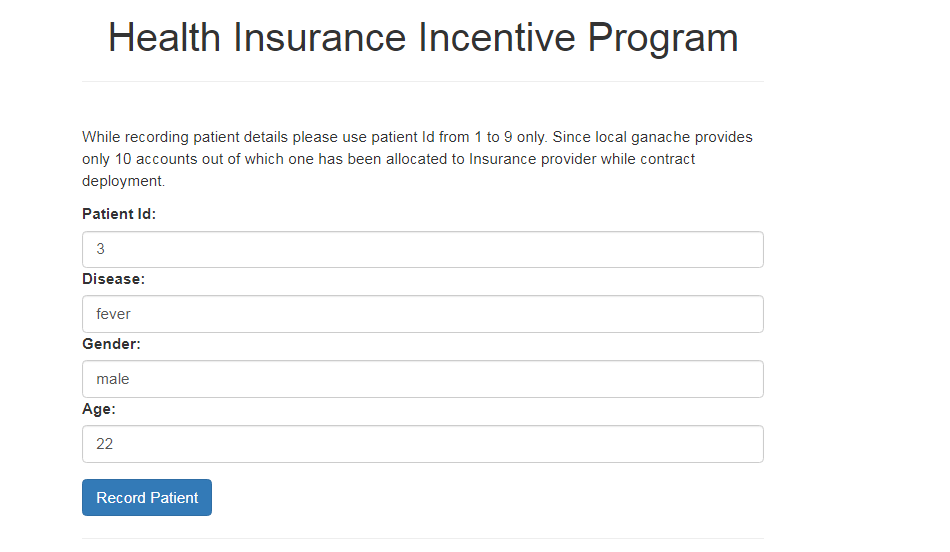
If running on windows do ensure the firewall settings aren’t interrupting with Ganache connection. If so, you may switch off the firewall from control panel for temporary.

Here is the first page of our app.

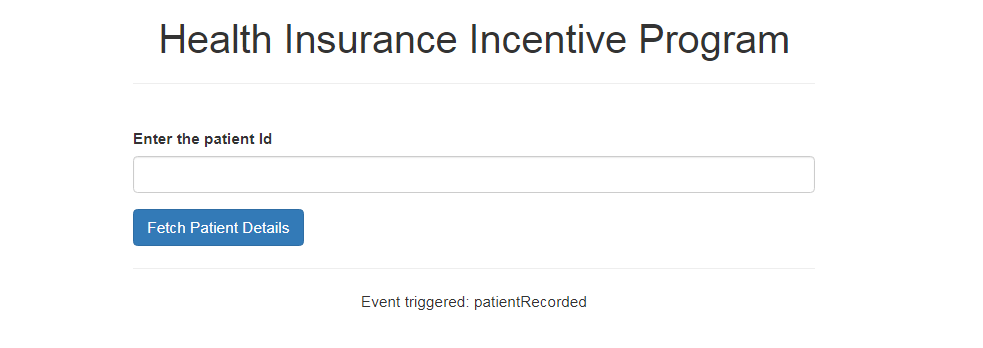


Enter the patient details and click on Record Patient button.

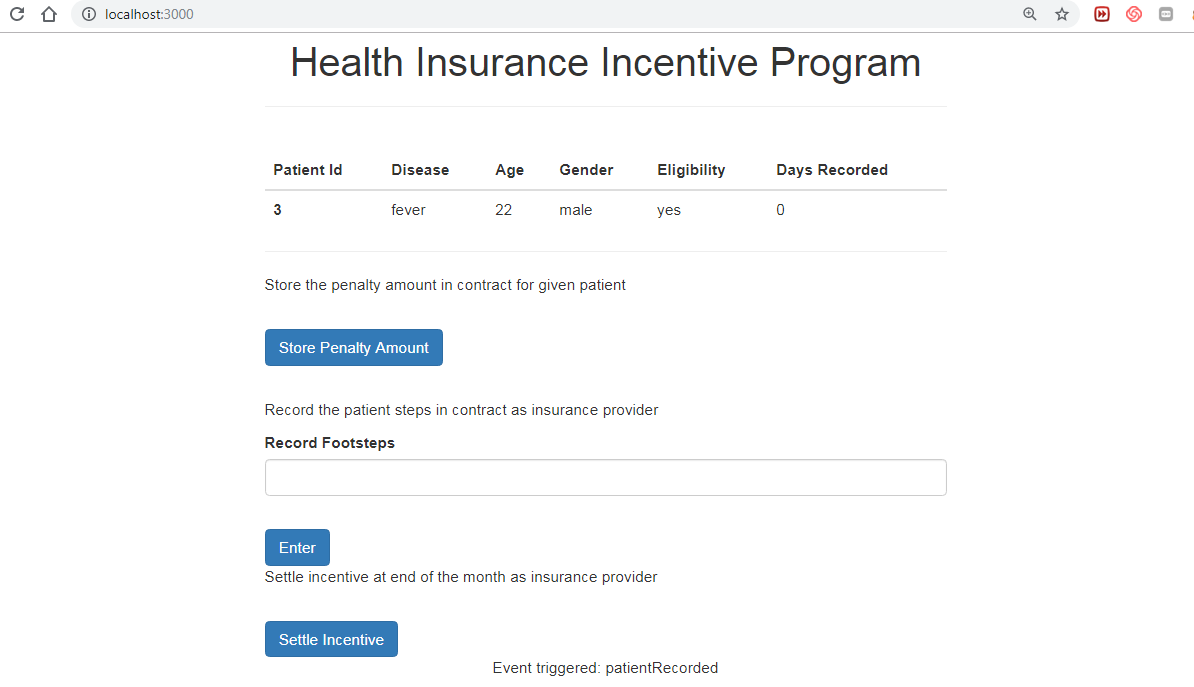
**Note:** For the demo please use patient Id 1 to 9 only. Since Ganache provides 10 accounts, out of which 1 is taken by Insurance provider, we have only 9 accounts for patients. Also note that local Ganache accounts index will be same as patient Id. (For ex: Ganache account no. 1 will belong to patient with Id 1).



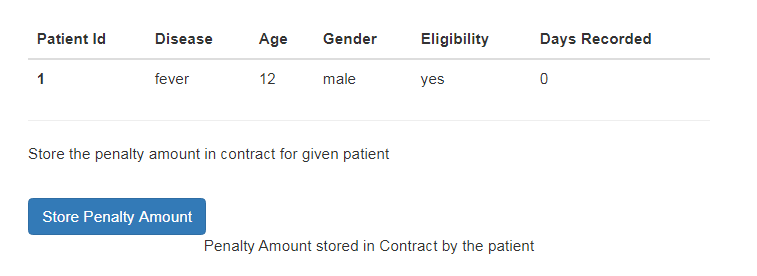
Once patient details have been recorded, following UI will get displayed.



Enter the patient Id & hit fetch button to get the patient details.

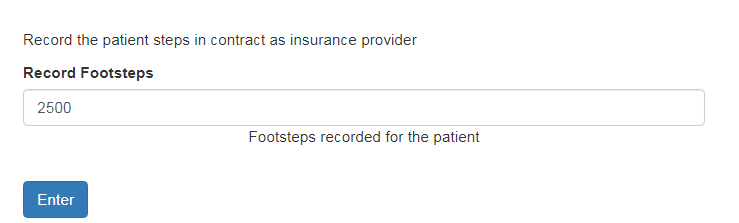


Switch your metamask account to Patient’s account (note that for given demo, the patient id from 1 to 9 will also be the index for Ganache accounts). And click on **Store Penalty Amount**, this will ensure penalty amount from patient’s wallet has been added to contract. Now Insurance provider is allowed to store the footstep details.



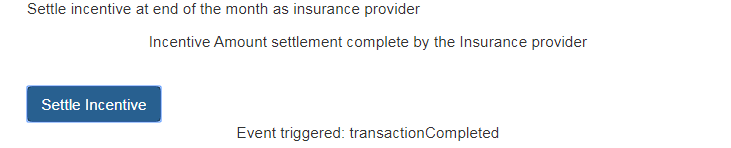
You will see a confirmation message as shown in above screenshot: **Penalty Amount stored in Contract by the patient**

Change the metamask account to the insurance provider account, enter the footsteps & click on enter button.



You will see a confirmation message as shown in above screenshot: **Footsteps recorded for the patient.**

After footsteps have been recorded for the **last day**, click on **settle Incentive** button.

****

You will see a confirmation message as shown above: **Incentive Amount settlement complete by the Insurance provider.**

You can check the incentive amount in patient’s wallet in metamask, or the penalty amount in insurance provider wallet.

**Please note any event triggered during transaction will be shown at the bottom center of the screen.**