

# INTRODUCTION

## WHAT IS A SMART CONTRACT ?

- PRE-WRITTEN LOGIC
- EXECUTED/RUN BY AND ON A NETWORK OF COMPUTERS
- USUALLY RESULTS IN STORING THE TRACK OF TRANSACTIONS

## WHY DO WE NEED SMART CONTRACT?

- BLOCKCHAIN WORKS ON A DISTRIBUTED TRUSTWORTHY CONSENSUS
- FOR SUCH A CONSENSUS WE NEED SOMETHING TO MAINTAIN CALCULATIONS

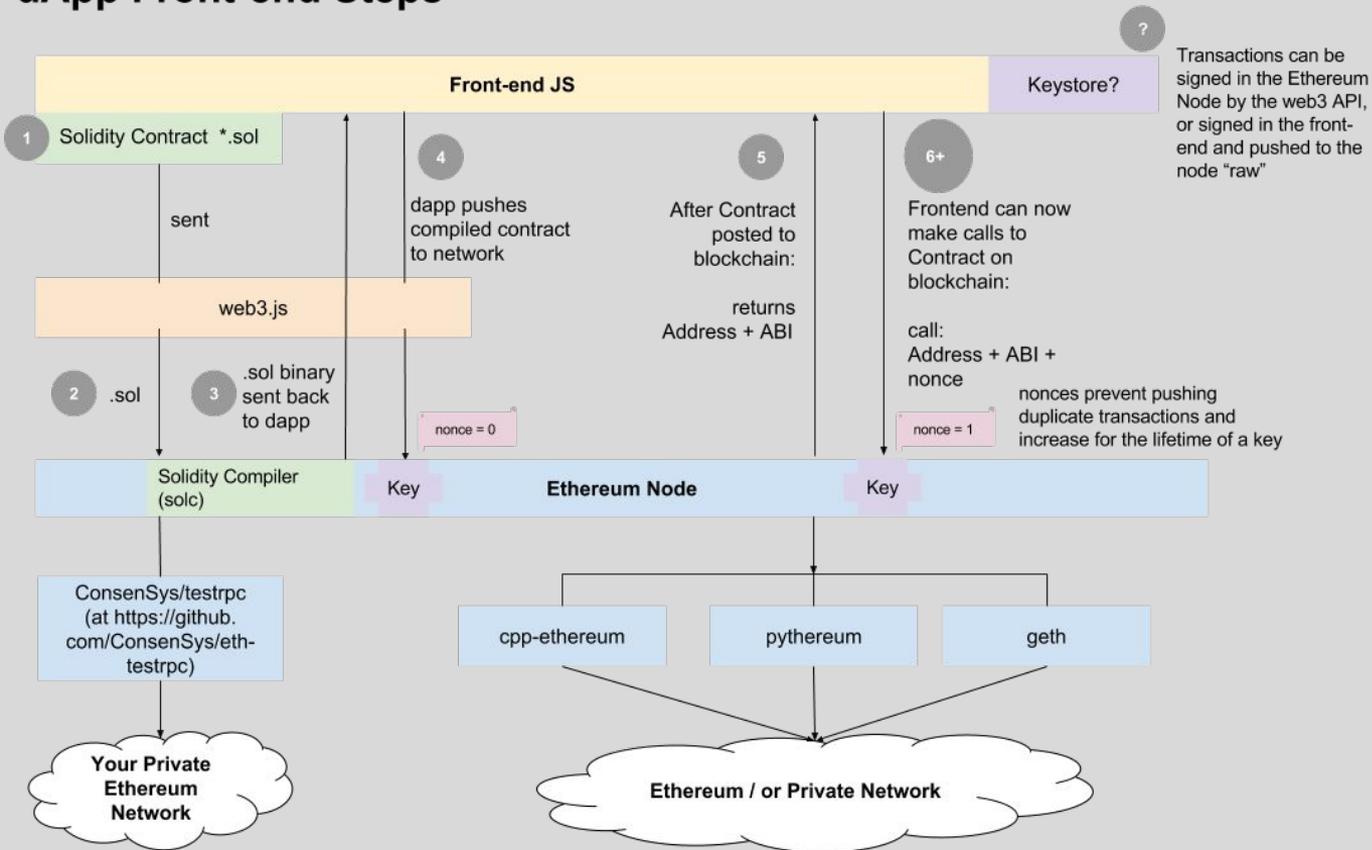
# WHY USE BLOCKCHAIN AND A SMART CONTRACT ?

- CONTROL OVER THE SYSTEM
- CODE PERFORMING TRANSACTIONS OVER THE SYSTEM
- TRANSPARENCY
- NO MUTUAL MISUNDERSTANDINGS

# THE GOAL OF THIS EXERCISE

- SET UP THE DEVELOPMENT ENVIRONMENT. [GANACHE, WEB3, SOLC]
- LEARN THE PROCESS OF WRITING A CONTRACT, COMPILING IT AND DEPLOYING IT.
- INTERACT WITH THE CONTRACT ON THE BLOCKCHAIN THROUGH A NODEJS CONSOLE.
- INTERACT WITH THE CONTRACT THROUGH A SIMPLE WEB PAGE TO DISPLAY THE VOTE COUNTS AND VOTE FOR CANDIDATES THROUGH THE PAGE.

# dApp Front-end Steps



A **Contract Creation Transaction** is shown in steps 1-5 at above.

An **Ether Transfer** or **Function Call Transaction** is assumed in step 6.

# SOME TOOLS THAT MAKE YOUR LIFE EASY!

- GANACHE CLI
- TRUFFLE/ EMBARK
- METAMASK
- REMIX IDE

TENDERWIZE

A SMART WAY FOR PUBLIC PROCUREMENT.

RS 5-7 LAKH CRORE EACH YEAR

OUR TAKE ON THIS.

```
graph LR; A[ORGANISATION RELEASES TENDER] --> B[CONTRACTORS APPLY. ONLY THE HASH OF THEIR SUBMISSION ARE STORED.]; B --> C[CONTRACTOR UPLOADS THE ORIGINAL APPLICATION FOR FURTHER SCRUTINY];
```

ORGANISATION  
RELEASES TENDER

CONTRACTORS APPLY.  
ONLY THE HASH OF THEIR  
SUBMISSION ARE  
STORED.

CONTRACTOR UPLOADS  
THE ORIGINAL  
APPLICATION FOR  
FURTHER SCRUTINY

```
graph LR; A[ORGANISATION RELEASES TENDER] --> B[CONTRACTORS APPLY. ONLY THE HASH OF THEIR SUBMISSION ARE STORED.]; B --> C[CONTRACTOR UPLOADS THE ORIGINAL APPLICATION FOR FURTHER SCRUTINY];
```

ORGANISATION  
RELEASES TENDER

CONTRACTORS APPLY.  
ONLY THE HASH OF THEIR  
SUBMISSION ARE  
STORED.

CONTRACTOR UPLOADS  
THE ORIGINAL  
APPLICATION FOR  
FURTHER SCRUTINY

AFTER DEADLINE EVERYONE SUBMITS ORIGINAL FILES

```
graph LR; A[ORGANISATION RELEASES TENDER] --> B[CONTRACTORS APPLY. ONLY THE HASH OF THEIR SUBMISSION ARE STORED.]; B --> C[CONTRACTOR UPLOADS THE ORIGINAL APPLICATION FOR FURTHER SCRUTINY];
```

ORGANISATION  
RELEASES TENDER

CONTRACTORS APPLY.  
ONLY THE HASH OF THEIR  
SUBMISSION ARE  
STORED.

CONTRACTOR UPLOADS  
THE ORIGINAL  
APPLICATION FOR  
FURTHER SCRUTINY

SINCE DATA STORED IN BLOCKCHAIN IS IMMUTABLE, SO PREVIOUSLY SUBMITTED HASH  
CAN'T BE ALTERED AND THEREFORE SUBMITTED FILE'S HASH SHOULD MATCH WITH  
PREVIOUSLY HASH SUBMITTED

# KEYS POINTS

- A PERMISSIONED BLOCKCHAIN WITH SEVERAL GOVERNMENT/ PUBLIC ORGANISATIONS AS NODES OF THE NETWORK.
- SCALABLE TO 100'S OF ORGANISATIONS.
- ENSURES A FAIR, SECURE AND CORRUPTION FREE PUBLIC PROCUREMENT.
- TRUSTABLE AS IT'S DECENTRALIZED, NO NEED TO TRUST A SINGLE PARTY.

# TECH STACK



TRUFFLE

