1. Abstract
2. Target Audience
   1. Huge Legacy  
      We target people who has huge legacy, that the amount is big enough for them to be worried while they are alive. Or even be cautioned by others that they should do something about it.
   2. Complex Inherit Rules

The target could have complex inherit rules, including trust, complex percentage or even company board seats.

* 1. Complex Inherit Relationship  
     A lot of people might pop up and claim their rights when our TA passes away. Our TA could have a complex inherit relationship that often causes law suits or arguments.
  2. High Time Cost  
     Our TA are set to be busy and might have high time cost. If they spend a lot of time dealing with these, kinds of things will be costly.
  3. Privacy of Testament  
     There might be some requests that our TA don’t want to reveal their content of testament. Revealing their testament might cause unnecessary problems.

1. Pain Point
   1. Tampering

In our current law system, there are still a lot of rooms for malicious tampering. There are always risks that others can access the content of a testament. For example, if any testament is in digital form, the fire wall can never really guarantee it won’t be accessed.

* 1. Time-consuming

Because of our TA’s feature, it will cause big problems if the testament is distorted by others. The existing system that prevents these kinds of situations are highly time consuming. In fact, the more complex the method is, the more difficult others can distort the content.

* 1. Problems about Complex Rules

If anyone wants to set a complex rule of testament, it’s gonna take more time than a normal testament. And how to enforce the rules in the future is also a big problem. If others do not follow, it’s going to be a lot of endless law suit.

1. Solution
   1. Pain point 1: Tampering versus Untamperable on Mechanism Level

Block Chain technology is based on a data structure built up by hash pointers and hash pointers can keep the all information on the same blockchain related tightly to each other. That means if you want to make any piece of information tampered and maintain the validation of the information, you must try to make all corresponding data changed the same time in a designate way. This is usually impossible.

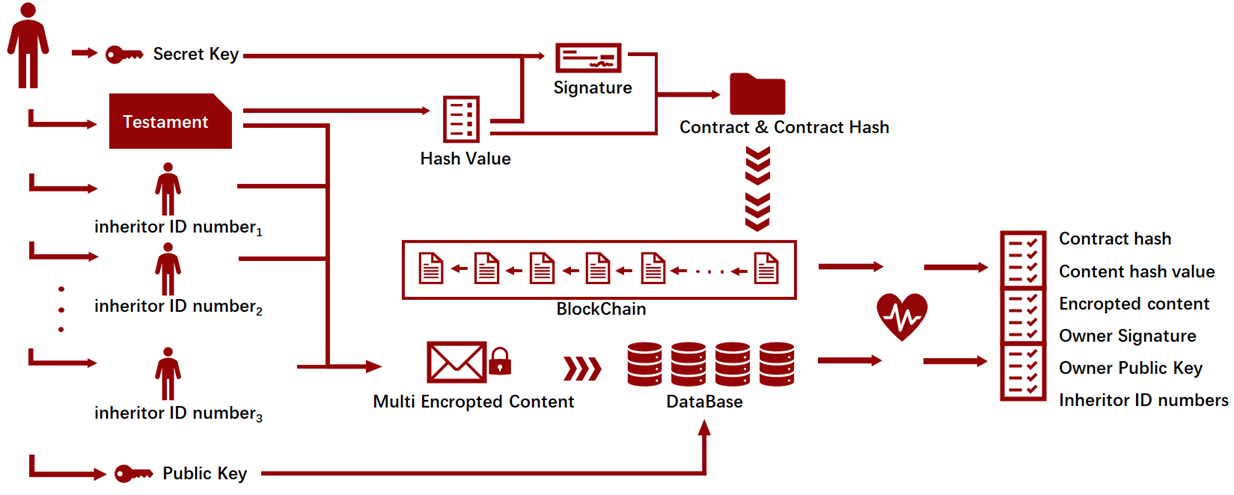
* 1. Pain point 2: Time-consuming versus Highly Efficient Online System

Block chain is deployed on information systems, you can finish all things with a terminal. The development of information technology and user interface design makes it easier and easier for customers to operate machine and manage personal data. Blockchain as a new technology is also an helpful interface to greatly improve the efficiency of information process.

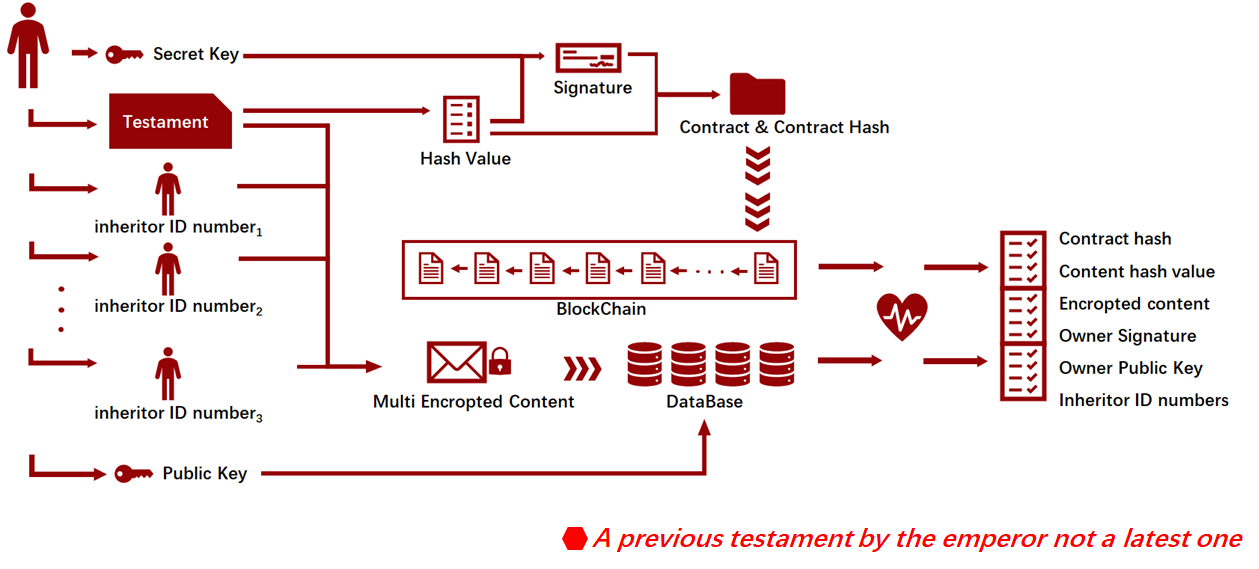
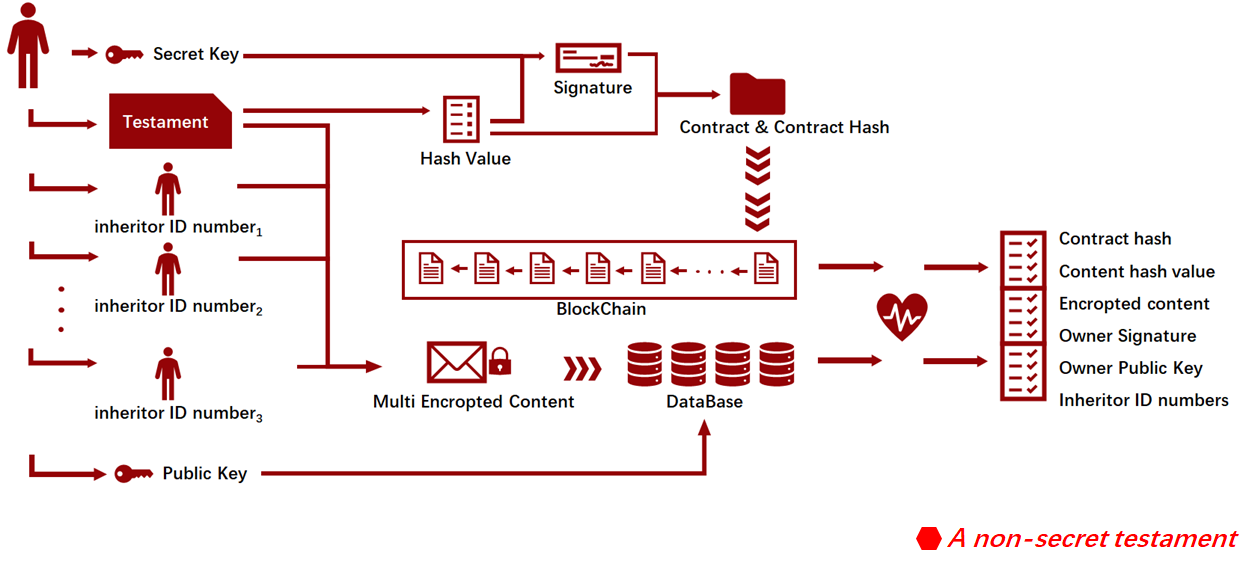
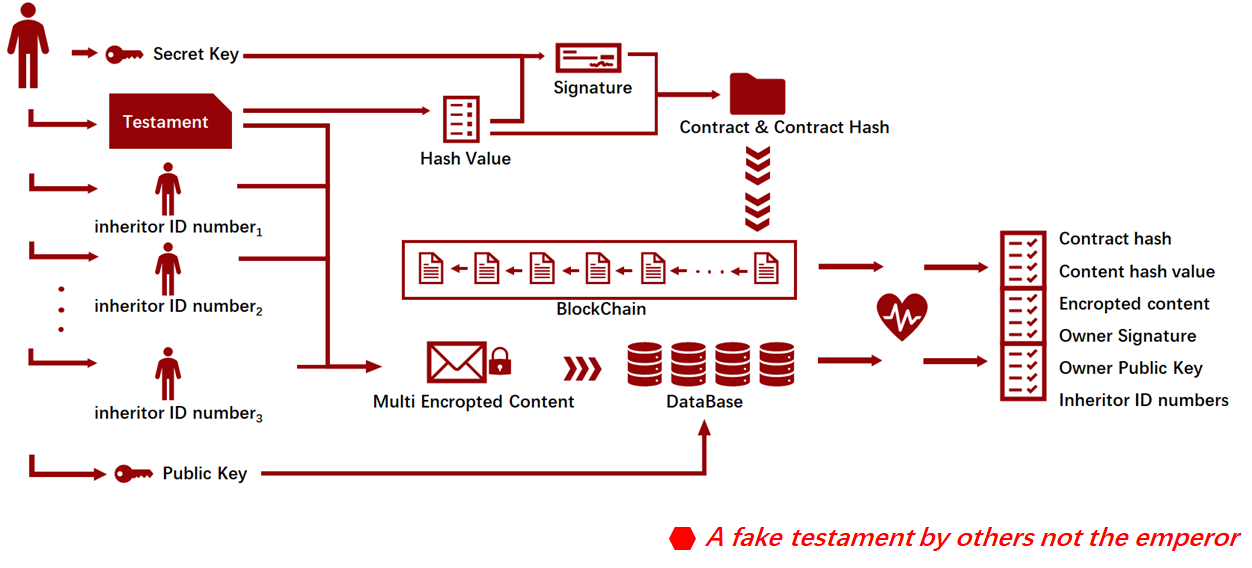
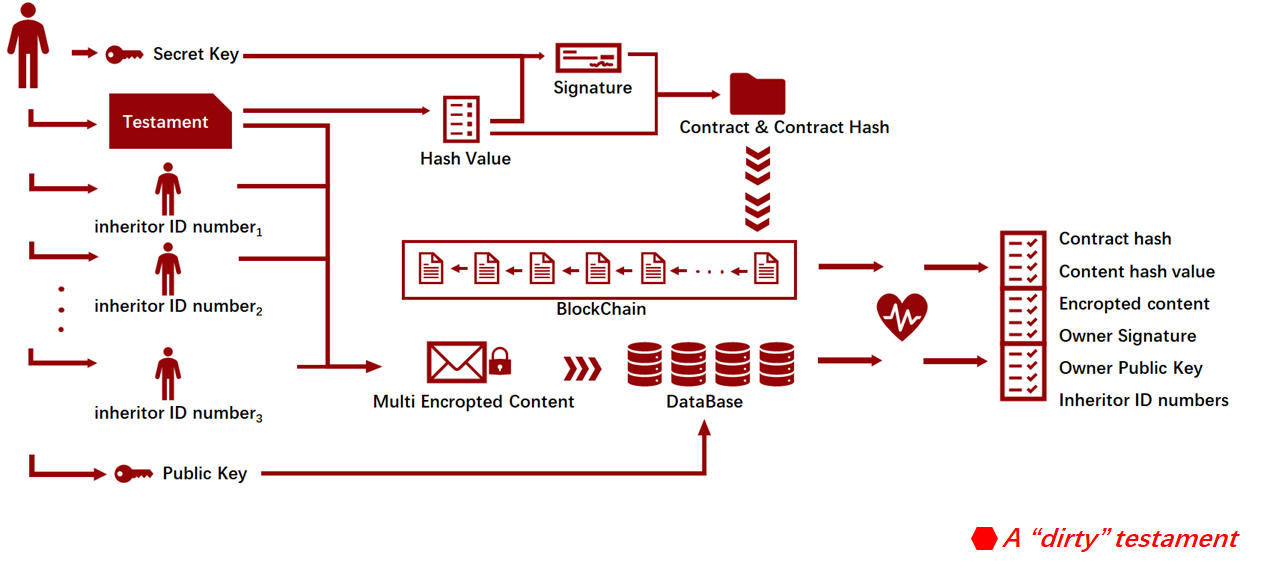
* 1. Pain point 3: Complex Rules in Testaments versus Smart Contract

Blockchain technologies, especially ethereum and the technologies later, can deal with complex logical processes of transactions, which make the applications on blockchain much “smarter”. Developers use a piece of code called smart contract to implement commercial contract, business logic and service in various fields like copyright protection, reserved evidence, and so on. Smart contract is a piece of logical code, which is more abstract and concise than tangible materials and also it works with high reusability, simplifying the process to set a legal testament.

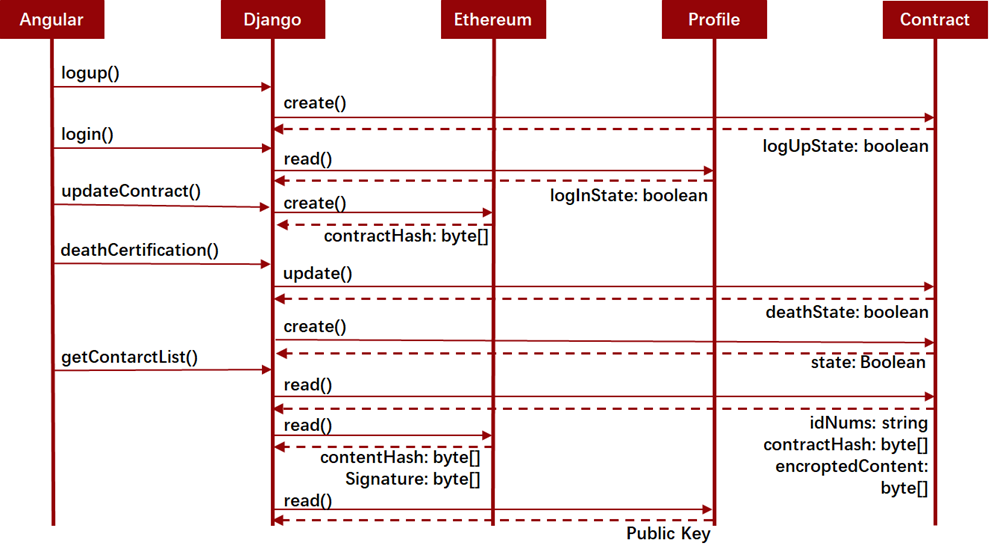
1. Product function
   1. Log up
   2. Update contract
   3. Download contract
   4. Validate
2. System structure



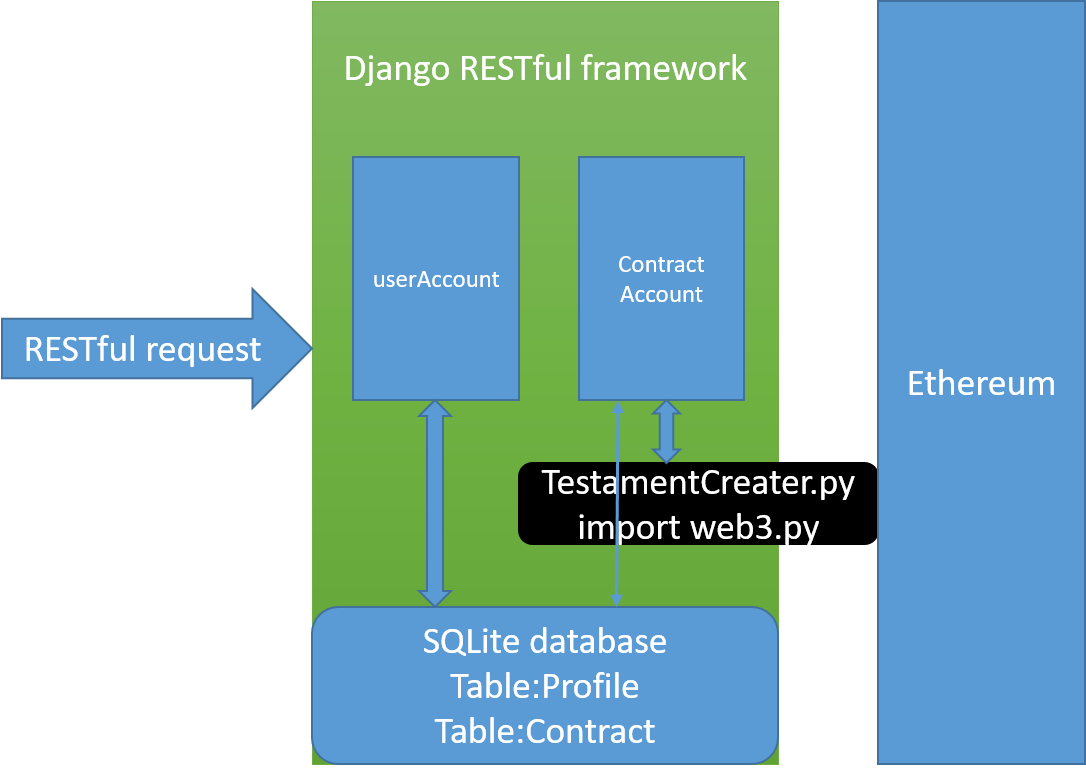
1. System mechanism



1. System sequence diagram



1. Front-end structure
2. Front-end requests
3. Back-end structure



* 1. brief
  2. userAccount
  3. contractAccount

1. api
   1. logup/

{

"idNum":"",

"psw":"",

"publicKey":"",

"email":""

}

{

"code": "1" //1 success 0 fail

}

* 1. login/

{

"idNum":"",

"psw":""

}

{

"code": "1" //1 success 0 fail

}

* 1. updateContract/

{

"content":"",

"idNum":"",

"relativeIdNums":[

"12345",

"1232455"

],

"private\_key":""

}

{

"code":"1", // or 0

"contractHash":""

}

* 1. deathCertification/

{

"idNum":"",

"institution":""

}

{

"code":"1" // 0, if no relative contract

}

* 1. relativeContracts/

{

"idNum":""

}

{

"contracts":[

{

"contractHash":"",

"encroptedContent":"",

"contentHash":"",

"ownerSig":"",

"ownerPublicKey":"",

"idNums":[

{

"idNum":""

},

{

"idNum":""

}

]

},

{

"contractHash":"",

"contentHash":"",

"encroptedContent":"",

"ownerSig":"",

"ownerPublicKey":"",

"idNums":[

{

"idNum":""

},

{

"idNum":""

}

]

}

]

}