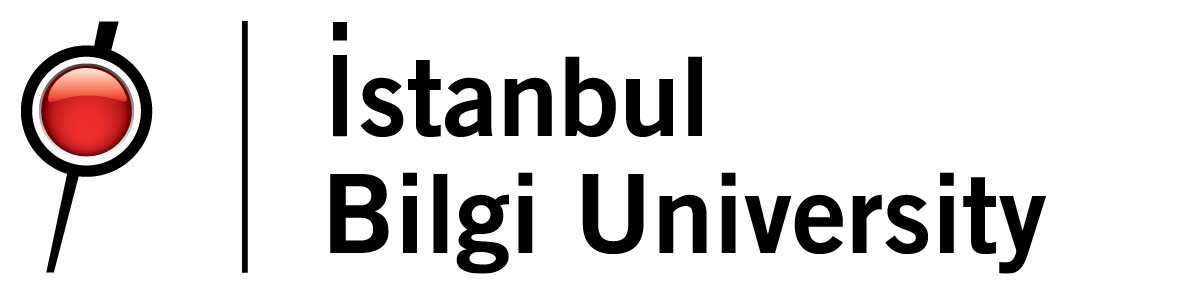
**

*by*

*Barış Tonbul – 117200054*

*CMPE-444*

*Project Report*

Istanbul Bilgi University

*Fall 2020*

**Aim and Importance of the Project**

The aim of this Project to benefit from smart contracts, which is used today and completes its purpose in the best way, to affect positively from contract and avoid any negative situation for both sides. In this project smart contracts used to make hotel reservation in the way of a smart contract that creates safe environment between two persons since nowadays, when a person wants to get any service, they want to involve third person in the contract due to lack of trust and it is not the best and reliable way to make a contract. According to recent conditions, many secure problems and lack of confidence occur. Looking for an environment of trust even in the simplest services will be receive. As a result, it is hard to give money without even getting service at the hotel without any guarantees but at the same time, the hotel does not serve anything without getting any payment and it wants to block credit cards or gets low prepayment which is the best way to ensure the service provision in case of facing any negative situation and best way to away this problem is smart contracts. Under favor of smart contracts when people pay for the service, they get it contentedly without paying extra money, deposit etc. Also, with this contract people going to don’t have to share their personal information with any kind of hotel employee. This project also provides some bringing for example lack of cost for middleman’s, protection of privacy, decrease of hotel prices since some of the employees won’t be necessary anymore and the last one is the disappearance of insecurity for both sides. When people come to the hotel, they will select the room they want to use, pass the service payment to the recipient through the smart contract, and get the password of the room that they will use at the hotel and the number of days they have paid for room will be available on the password when the last day completed the password won’t going to work anymore.

**Smart Contract**

A smart contract is a computer program or a transaction protocol which is intended to automatically execute, control or document legally relevant event and actions according to the term of a contract or an agreement. They are self-executing contracts by writing the agreement between the buyer and seller directly into the lines of code. In other words, it is a tool that formalizes and secures computer networks by combining user interfaces and protocols. To use Ethereum it implements a Turing-complete language on its blockchain, a prominent smart contract framework. Usually, they operate as digital agreements that have to comply with certain rules. These rules are pre-determined with codes on the computer. It is running on blockchain.

**Requirement List**

1.A virtual wallet on blockchain to use the Ethereum to pay for service on the contract.

2.Smart contract that coded appropriately for its aim and target.

3.When both of the sides fulfill the convention grounds their public key shared with each other.

4.The properties and prices of the rooms will be on the screen that client use to pay for room

5. Making a choice and paying for the cost and synchronously if the conditions completed the password of the room that clients choose will be on the screen that client use.

6.Clients pay for the day(s) they want to stay, when the day(s) completed the password that has given to them get changed and occurred contract code and 2 open keys. These keys be determined as one for client and one for seller.

**How It Works**

When a client wants to book for a hotel room, they reach hotel’s smart contract (to book a room) screen. In this screen there are room’s IDs, prices, available dates and information. Client needs to choose one of the rooms by clicking their price (they can only choose one room in one step). After they choose the room, they need to click to pay button with this button the room will be reserved for the client and client’s money will transferred to deployed contract's address. After that when client goes to the hotel and makes sure it exists and the room looks like the photos that on the hotel website the client can confirm the contract by clicking check-in button then the money that on the contract’s address will be transferred to the hotel owner’s Ethereum Address. Also, the client gets a 6 numbers password that sends from hotel to client’s phone when client click the check-in button, but the password can’t be used after the check-out time since it gets changed by the hotel to make ready to room for the next client. If the room does not look like the photos on the hotel website or it does not even exist, the client cancel the process by clicking the cancel button so the money go back to the client’s bank account.

1- With Ganache program, created virtual Ethereum network on a particular way. There are three or four accounts, and each account has a hundred ether which identifies to MetaMask.

2-There is a web page as rooms.html and in that page, pay regard to design and settlement, there are some buttons about doing some events and determined what will do these buttons and these coded on script.js.

3-Main control of webpage with flask library and using app.py that is a python application; with using library flask, completed to connection of hotel database and through this all data can assigned into html codes.

4-Remix.ide that is website, deploy contract which is written in solidity through hotel.sol. In other words, contract take places present block in consisted artificial Ethereum network with specialized addresses. (Contract has own addresses such as users.)

5-Consisted virtual accounts from Ganache, identifies in MetaMask through chrome. When provide same port, a room connected to virtual network.

6-When entered to website and then, with selected button and it actuates the own function.

**Used Technologies**

**Ganache**: For virtual Ethereum network (instead of real payment on Ethereum network, tried in virtual Ethereum network)

**MetaMask**: An application that operates and manages Ethereum accounts and also imports accounts on virtual Ethereum network which created on Ganache.

**Python**: Provides organization of CSS, JavaScript files and html with flask library. And it provides connection to MySQL.

**remix.ethereum.org**: For deployment of contract.

**XAMPP**: Created the project’s MySQL database.

**VsCode**: For arrangements.

**Web3.JS**: It provides us to make transactions and connect HTML pages via JavaScript code on Ethereum network.

**References**

<https://docs.metamask.io/guide/getting-started.html>

<https://flask.palletsprojects.com/en/1.1.x/>

<https://ethereumdev.io/getting-started-with-solidity-smart-contract/>

<https://stackoverflow.com/>

<https://www.trufflesuite.com/docs/ganache/quickstart>