

29/05/2022:

Was sind Smart Contracts, Recherche:

Tiny computer programs stored on the blockchain, they are immutable (which means that once they are deployed they are not changable) and distributed (that means, that validated by everyone on the network, also means decentralizel, no thirdparty has control over it)

Smartcontracts eliminate the middleman

Decentralized = hosted on thousands of private computers

Ziel:

Alle Spieler in einer Lobby, jeder verbindet seine wallet mit dem spiel

Funds werden eingesammelt

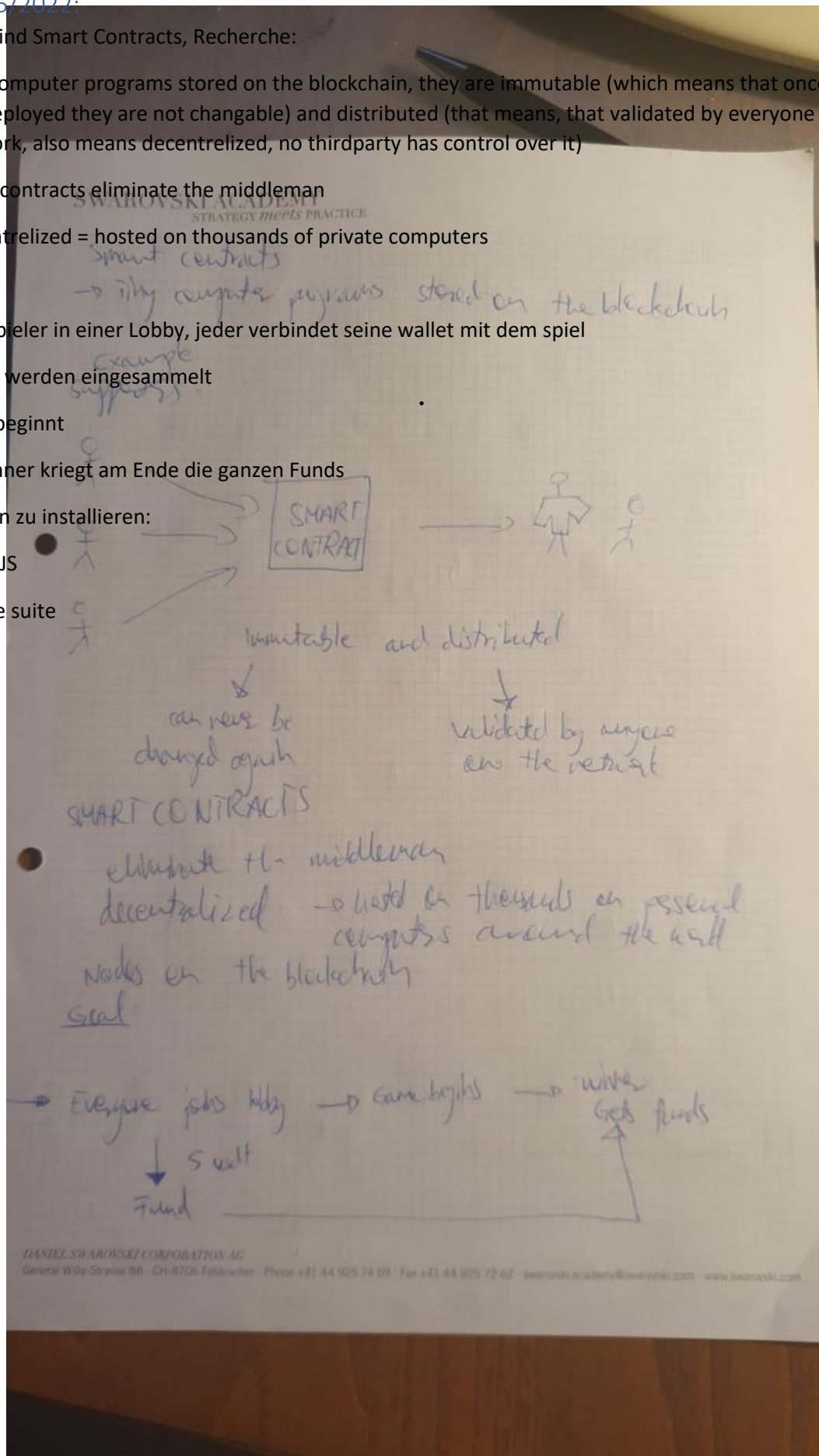
Spiel beginnt

Gewinner kriegt am Ende die ganzen Funds

Sachen zu installieren:

Node JS

Truffle suite



30/05/2022:

Auf welchen Plattformen soll Computerspiel laufen  
Allgemein definieren: ziemlich als Web-basiert machen  
Beifluttext, Eigenanteil präsentieren, annehmen dass  
Grundverständnis da ist, warum es cool ist ~~haben~~  
was wir wie machen

Paper based prototype bauen, ganzen HLAUT von der  
Seite von Spieler definieren.

→ Spielzeit  
im Detail skizzieren →

↳ Grobkonzept perfektionieren, so werden wir es  
wirklich haben!

Recherche anhand Kriterien:

Welche Blockchain werden ich benutzen → erst danach  
Environment  
Community suchen! installieren

Behörde fragen  
Regulationen  
Kooperationen

↳ Fragen andere  
Developer

Andere Leute fragen, tolles  
Netzwerk suchen polieren

Teilen, Socializen.

Gefährliches  
Kriterium → Dfrees



Zeitplan schreiben! → separat in abhängigkeit

Tommas → zuerst Prototyp funktionalisieren → Umkehrung der Teilaufgaben  
Minimal Viable Product

Spiebbuch ins Grobkonzept einfügen

→ Tommas Zeitplan auf Filip's abstimmen

→ Filip: Ab 18 Jahren

Soziale Verantwortung

Next To-do's:

Zeitplan → ins Grobkonzept → Sonntag

Grobkonzept mit Jugendschutz und Spiel sucht updates

Environment installieren

Filip

↳ Titel überarbeiten

Motivation anpassen  
Nicht natürl. formulieren

"Must have"

"Nice to have"

trennen

3. Sitzung individuelles Beraten

→ Probleme aufschreiben und Fragen vorbereiten

Workcase Szenario, was könnte jede Einzelne erschaffen.



To-do's

→ Copy Paste Standardinformation gemäß Vorlage  
Vorname + Nachname

→ wenn möglich Browse-basiert

↳ Metamask wallet externe  
Geldwäsche und Jugendschutz ~~ausserhalb~~ auslagern  
Recherche betreiben → dürfen wir das sammeln!

↳ User account exchange

Spieldurchlauf revidieren, welche Vorgaben??!

ethik, eigene Entscheidung abhandeln.

↳ Marketingvorteil

↳ Gedanken werden und fortsetzen, wieso haben wir was so

Kriterien: Testungsbungen, Backcheck anhand  
entschieden  
Kriterien wählen

↳ begründete Wahl, Begründung sichtbar machen  
was sind Kriterien etc.

Sieger und Verlierer abhandeln

wo werden die Fees abgezogen.

Klare Spielprozess definieren im klaren Detail wo wer

↳ Tomas baut Rahmen der Smart contracts angeschlossen  
UI

→ offizielle Zwitterbesprechung nach dem Sommerferien  
Eigenschaft nach updates am schluss

16/06/2022:

Welche Blockchain benutzen?

Erste Idee: Ethereum Blockchain, jedoch problematisch sind die hohen Transaction Fees 8dollars +, welche wir einfach nicht verantworten können

## BSC Objectives

- Faster throughput than ETH
- Lower Fees than ETH
- Easy to onboard new Developers

BSC is compatible with [EVM](#) (Ethereum Virtual Machine). This is great for BSC, since it allows us to develop BSC dApps with the same tools as for Ethereum dApps.

Das heisst ich kann auf der binance smart chain smart-contracts deployen und trotzdem die gleichen sachen benutzen wie für eth chain development

BSC ensures compatibility with the current Ethereum mainnet in order to facilitate access to its community and collections of relatively mature applications. As a result, Binance Smart Chain easily opens up the scope for using majority of [dApps](#), tooling, and ecosystem components based on Ethereum.

<https://101blockchains.com/binance-smart-chain/>

18/06/2022:

Probe von Entwicklung erster Smart Contracts mit Remix online Ide, mit der solidity language

<https://remix.ethereum.org>

```
pragma solidity 0.5.1;

contract Counter{
    uint public count = 0;

    event Increment(uint value);
    event Decrement(uint value);

    function getCount() view public returns(uint) {
        return count;
    }

    function increment() public{
        count += 1;
        emit Increment(count);
    }

    function decrement() public{
        count -= 1;
        emit Decrement(count);
    }
}
```

19/06/2022:

Versuch von Verbindung von Remix ide mit lokalem ide, für speicherung von code, jedoch sind einige Fehler aufgetaucht, bei der Benützung von Visual Studio Code, ich konnte keine Lösung für diese Problem finden

Ich konnte jedoch Node.js installieren und habe Visual Studios probiert für solidity code editing zu setupen, hat aber nicht funktioniert

Nach einiger recherche wegen frustration:

Start Development today with BSC Studio

BSC Studio is a powerful yet user-friendly graphic IDE platform, specially designed for BSC. It supports running locally on majority OS, including macOS, Windows and Linux and also on web browsers. Comparing with traditional Ethereum development toolkit like Remix and Ganache, BSC Studio streamlines environment preparation procedure and integrates essential tools for a complete development lifecycle, including a graphic code editor, project manager, keypair manager, block explorer, contract inspector and network manager.

<https://github.com/ObsidianLabs/BSC-Studio/releases/tag/v0.10.5>

<https://github.com/ObsidianLabs/bsc-dapp-tutorial/blob/master/README.EN.md#About-Binance-Smart-Chain>

➔ Könnte mir mein Leben massiv einfacher machen, deshalb fällt die wahl wohl auf binance smart chain

Gründe für Entscheid für Binance Smart Chain:

- Tiefe Transaktionskosten (5 cent), wobei bei Ethereum 8dollars +
- Eventuell extreme vereinfachung von installationen der Entwicklungsumgebungen mit bsc studio

20/06/2022:

In kontakt mit anderen developers treten, schauen welche blockchains andere projects benützen

Präsentation anfangen zu entwickeln, schon Frustrationen und Emotionen notizen, vielleicht schon powerpoint machen und entscheidungen festhalten

Zeitplan in ein separates document reinkopieren und anpassen, gedankenspiele wie sieht presentation aus

SEHR WICHTIG:

- Festhalten von entscheidungen und vom progress, das ist wirklich sehr wichtig

Nächste Todo's:

Privaten zeitplan erstellen, den ursprünglichen kopieren

Vorstellung vorbereiten für frau germann

Ev github, fragen, nach hilfe suchen

Realisation:

Der Grossteil der Arbeit wird das installieren der environments, und das finden der ansätze sein, das programmieren an sich wird eher ein kleinerer teil der arbeit sein, auch das verbinden zu dem spiel wird ein grössere teil sein

29/07/2022(nachträglich eingetragen):

Ich habe endlich probiert mit BSC Studio zu arbeiten, jedoch sind natürlich viele Probleme aufgetaucht, so musste ich zuerst Docker herunterladen, das hat jedoch nicht sehr gut geklappt, da man aus dem Studio heraus docker starten sollte, es gibt danach immer error, konnte nicht starten, bitte nochmals probieren. Nur ein sehr simples tutorial zum bsc studio setupen, zu wenig materialien um wirklich alle probleme zu beheben. Habe mich entschieden ganz einfach anzufangen und erst einige smart contracts in remix ide (Online Ide zu starten). Fange an einem langen Smart contract tutorial zu folgen, sollte gut gehen. Ist jedoch auf Ethereum blockchain, mal schauen wie es weiter geht. Eventuell werden wir wohl oder übel müssen auf Eth umschalten, da die tutorials und vor allem hilfe zu problemen viel besser sind.

Eigene Gedanken: Als ziemlicher Anfänger in der Blockchain welt anzufangen ist extrem schwierig, man weiss einfach nicht wirklich wo richtig anfangen. Weil zB was Tomas macht ist in diesem Punkt sehr viel klarer. Er kennt sein entwicklungsumgebung und muss darin probleme lösen, ich habe fast kein anschluss gehabt, fange nun an einen zu finden. Man ist so am herumtreiben von Information zu Information aber jeder macht es anders.

31/07/2022:

<https://youtu.be/M576WGiDBdQ> tutorial

Fange an erste Smart contracts zu schreiben, mein plan zuerst Solidity zu lernen, smart contracts deployen und nachher konkret meine entwickeln, wenns nicht mir Binance Smart chain geht, dann halt mit eth

```
contract SimpleStorage{
    uint256 favouriteNumber = 5;
    bool favouriteBool = false;
    string favouriteString = "Juice";
    int256 favouriteInt = -5;
    address favouriteAddress = 0xe7Aa43a73Cb5B4884fF0074D089a67bD86a0C127;
    bytes32 favouriteBytes = "cat";
}
```

Arbeit des Tages:

```
pragma solidity ^0.6.6;

contract SimpleStorage{
    uint256 favouriteNumber;

    struct People {
```



```

        uint256 favouriteNumber;
        string name;
    }

    People[] public people;

    function store(uint256 _favouriteNumber) public{
        favouriteNumber = _favouriteNumber;
    }

    function retrieve() public view returns(uint256){
        return favouriteNumber;
    }

    function addPerson(string memory _name, uint256 _favouriteNumber)public {
        people.push(People(_favouriteNumber, _name));
    }
}

```

<https://youtu.be/UoGzV094jhE> eventuell kanns helfen später

03/08/2022:

Ich schreibe an meinem Smart contract weiter, und folge einem Tutorial, ich habe das rinkeby test network benutzt um smart contracts zu deployen

- ⇒ Frage? Wie werden wir es machen dass man sich mit einer wallet zu unserer software connected, vielleicht werden wir es doch browser based machen müssen um uns mithilfe von metamask zu verbinden, später werden wir natürlich weggehen können von webbased mithilfe von trustwallet und qr code connection aber am anfang?

```

⇒ pragma solidity ^0.6.6;
⇒
⇒ import "./SimpleStorage.sol";
⇒
⇒ contract StorageFactory is SimpleStorage{
⇒
⇒     SimpleStorage[] public simpleStorageArray;
⇒
⇒     function createSimpleStorageContract() public{
⇒         SimpleStorage simpleStorage = new SimpleStorage();
⇒         simpleStorageArray.push(simpleStorage);
⇒     }
⇒
⇒     function sfStore(uint256 _simpleStorageIndex, uint256
        _simpleStorageNumber) public{

```

```

⇒         SimpleStorage(address(simpleStorageArray[_simpleStorageIndex]))
           .store(_simpleStorageNumber);
⇒
⇒     }
⇒
⇒     function sfGet(uint256 _simpleStorageIndex) public view returns
    (uint256){
⇒         return
    SimpleStorage(address(simpleStorageArray[_simpleStorageIndex])).retriev
    e();
⇒
⇒     }
⇒
⇒ }
⇒
⇒

```

04/08/2022:

Ich habe Smart Contracts in Smart contracts benutzt und chainlink oracle benutzt um Ethereums aktuelle preis zu finden, langsam habe ich Gedanken und Vorstellung wie meine endgültigen Smart Contracts aussehen werde,

**Zuerst bei meinen eigenen Fokus auf Walletbalance finden einer wallet,**

**Dann ein smartcontract entwickeln um assets auf der testnet von einer wallet zu der nächsten zu verschicken**

**danach mithilfe von random number zwischen 1 und 2 gewinner bestimmen zwischen zwei wallets**

**If number 1 equal true then send from this wallet to the other (and opposite)**

**Bald** werde ich versuchen mit meinen Smart Contracts im Remix Ide (online environment) etwas anzufangen, mit hilfe von documentation und den drüber stehenden Ideen sollte ich etwas hinkommen

FOKUS in den nächsten Wochen, bei der entwicklung der Smart contracts ist es unglaublich wichtig dass ich einen Anschluss entwickle, eventuell zuerst sogar versuchen die smart contracts in remix zu behalten und mit api oder so connecten zu Tomas!

```

// SPDX-License-Identifier: MIT

pragma solidity ^0.8.7;

import "@chainlink/contracts/src/v0.8/interfaces/AggregatorV3Interface.sol";

contract FundMe{

    mapping(address => uint256) public addressToAmountFunded;

```

```

function fund() public payable{
    uint256 minimumUSD = 5 * 10**18;
    addressToAmountFunded[msg.sender] += msg.value;
}

function getVersion() public view returns (uint256){
    AggregatorV3Interface priceFeed =
AggregatorV3Interface(0x8A753747A1Fa494EC906cE90E9f37563A8AF630e);
    return priceFeed.version();
}

function getPrice() public view returns(uint256){
    AggregatorV3Interface priceFeed =
AggregatorV3Interface(0x8A753747A1Fa494EC906cE90E9f37563A8AF630e);
    (,int price,,) = priceFeed.latestRoundData();
    return uint256(price * 10000000000);
}

function getConversionRate(uint256 ethAmount) public view
returns(uint256){
    uint256 ethPrice = getPrice();
    uint256 ethAmountInUsd = (ethPrice * ethAmount) / 1000000000000000000;
    return ethAmountInUsd;
    //0.0000016151267820800
}
}

```

05/08/2022:

Ich habe probiert einen Smart contract zu programmieren der mir hilft die balance in meiner wallet herauszufinden. Und ja ich habe es geschafft, natürlich mit viel recherche und habe das meiste abgeschaut, aber ich fange langsam an es zu verstehen.

```

contract MyContract
{
    // Private state variable
    address private owner;

    // Defining a constructor
    constructor() public{
        owner=msg.sender;
    }

    // Function to get
    // address of owner
    function getOwner(
    ) public view returns (address) {
        return owner;
    }
}

```

```
// Function to return
// current balance of owner
function getBalance(
) public view returns(uint256){
    uint256 balance = owner.balance;
    return balance;
```

In diesem contract sehen wir wie wir die balance einer wallet finden

06/08/2022:

Ich habe heute weiter meinem Tutorial gefolgt für ungefähr eine Stunde und gelernt wie ich von meiner Wallet funds herausnehme und wieder zurückdrauflade. Das alles in einem Smart Contract. Langsam setzt sich alles etwas zusammen und ich denke ich werde bald anfangen können meine eigenen Smart Contracts anfangen zu schreiben, zuerst in Remix und danach irgendwie aus Remix herausholen. Ev bald mit Tomas diskutieren um Sachen zu besprechen.

Was zu besprechen ist:

- Eine version webbased gestalten, falls es ohne riesigen Arbeitsaufwand geht, die wäre eigentlich auch ansonsten gut zu haben  
denn es ist für Spieler viel einfacher einfach webbased zu spielen und sich mit einer MetaMask wallet zu verbinden als zuerst software zu installieren und nachher sich irgendwie verbinden. Könnte gut sein um Kunden anzuziehen.

Ideen um Kunden anzuziehen: Tournament events organisieren, mit preisen, preise kommen natürlich von den eintritts-fees.

07/08/2022:

Heute habe ich meinem Tutorial weitergefolgt und habe ein Smart contract beendet welches mir erlaubt in das smart contract Ethereum zu funden. Und es danach zu withdrawen. Es können andere wallets funden und ich kann dann mit meiner withdrawen. Eventuell werde ich etwas sehr ähnliches in meinem Smart contract benützen können.

Nun werde ich weiter dem tutorial folgen und mit web3.py package in python(!!) weiter programmieren. Es ist ein package welches erlaubt irgendwie mit python Ethereum zu coden, sehr gut!! Denn ich kann die python basics.

08/08/2022:

Heute habe ich weiter meinem Tutorial gefolgt und habe visual studio code installiert und dort die packages und extensions installiert. Bei einem package hatte ich grosse Mühe es zu installieren den irgendwie wollte es nicht funktionieren. Nämlich beim Black python formatter, ich habe ein Stunde lang versucht es zu reparieren bevor ich schlafen ging, morgen werde ich weiter versuchen

09/08/2022:

Nach etwa 40 Minuten und nachdem ich viele verschiedene Artikel und Videos angeschaut habe, funktioniert es (black python formatter). Ich habe Mühe gehabt, doch es geschafft, sehr gut. Nun kann ich weiter meinem Tutorial folgen und weiterlernen Smart Contracts zu entwickeln.

```
from solcx import compile_standard, install_solc  
  
install_solc("0.6.0")
```

Solc hat nicht funktioniert für etwa 30 min, bis ich install solc eingefügt habe

10/08/2022:

Ich folge nun meinem Tutorial weiter, etwas mit Ethereum gas fees hat nicht funktioniert, habe jedoch error kopiert und im Browser nachgeschaut und nun funktioniert es. Musste nun zum zweiten Mal etwas anders machen als im Tutorial, doch so lerne ich selbstständig Probleme zu lösen.

Arbeit mit Ganache als lokales Testnetz

11/08/2022:

Heute folge ich meinem Tutorial weiter. Gestern habe ich einen Call mit Tomas gemacht und wir haben diskutiert, woran jeder gerade dran ist. Wir haben auch darüber gesprochen, dass wir wahrscheinlich auf ETH hinüberwechseln müssen. Dabei habe ich nachgeschaut, wie hoch die Ethereum fees gerade sind. Da es ja nur das Fee Problem ist. Und offenbar sind Ethereum fees stark gefallen über die letzten paar Monate (da Ethereum selber auch sehr stark gefallen ist.) Die fees sind im Moment 0.8 Dollars. Was sehr gut ist.

Ich habe gerade weiter programmiert und node.js installiert.

12/08/2022:

Ich habe weiter meinem Tutorial gefolgt und Brownie installiert, das erlaubt sehr viel schneller Smart Contracts zu deployen. Ich benutze außerdem Ganache als lokale Blockchain, auch ohne UI inzwischen sondern via Terminal. Im Moment bin ich einfach am Lernen, und bald werde ich mich an meine Smart Contracts setzen.

13/08/2022:

Heute habe ich weiter dem Tutorial gefolgt, ich lerne immer mehr, jedoch denke ich, dass ich noch ziemlich viel Lernen muss, um wirklich selbstständig an meinen Contracts arbeiten zu können. Ich werde am Dienstag probieren, meine eigenen Contracts wenigstens versuchen zu schreiben, und ansonsten sonst einfach dem Tutorial zu folgen.

Eigentlich sind ja meine Contracts gar nicht allzu kompliziert. Ich muss lediglich eine Wallet Connection bauen, also mit Metamask im Browser als HTML-File wahrscheinlich. Und danach in einer Python-File ein Random Winner Generator programmieren. Und danach auf der Testnet von der Verlierer-Wallet in die Sieger-Wallet Funds überweisen von der Verlierer-Wallet.

Progressplan:

- Manuell zwei Wallets verbinden auf der Rinkeby Testnet
- Dann mithilfe einer Random Winner Picker Function (die ich mache) den Gewinner auswählen

- Und sofort fake eth überweisen lassen.

Gedanken: Wir müssen es so machen dass man im Moment in dem man die Runde beitrifft die funds reserviert werden, denn ansonsten könnten am Schluss verlierer die transaction abbrechen. Alles was der spieler bestätigen muss, muss vor dem Spiel geschehen

15/08/2022:

Heute habe ich 30 minuten lang an einem error herumgesucht nur um zu bemerken dass ich ein Doppelpunkt anstatt einem Strich gemacht habe. Ich habe auch andere weiter errors gelöst. Inzwischen kann ich die meisten errors lösen wenn ich genug lange herumprobier und im Internet herumsuche.

16/08/2022:

Ich habe ein error gehabt wo sich brownie beim deployen der contracts nicht automatisch zu einem offenen ganache rpc client verbunden hat, ich musste unter server 7545 zu 8545 ändern und jetzt funktioniert

<https://stackoverflow.com/questions/69818605/brownie-doesnt-automatically-attach-to-local-ganache-when-i-have-ganache-open-i>

brownie networks add Ethereum ganache-local host=http://127.0.0.1:8545 chainid=1337

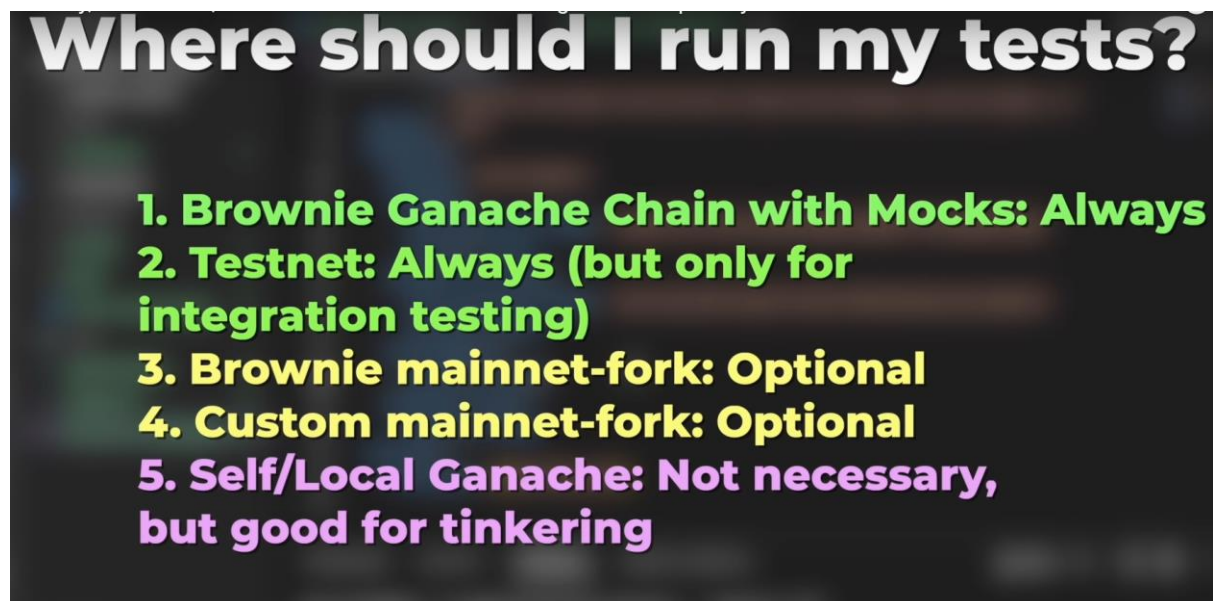
wir können zu brownie jedes evm network hinzufügen wie avalanche, polygon oder auch binance smart chain

18/08/2022:

Heute habe ich meine erste Github repository veröffentlicht mit meinen Projekten die ich im Moment machen bin

[https://github.com/nuevofilipo/brownie\\_fund\\_me](https://github.com/nuevofilipo/brownie_fund_me)

das habe ich bisher gelernt, bald geht es an das entwickeln meiner eigenen smartcontracts





Das sollte man machen zum smart contracts testen bevor man sie auf der mainnet für benutzer ausstellt

20/08/2022:

Heute habe ich im Rahmen meines Tutorials angefangen ein dApp zu programmieren. Ich werde in diesem projekt ein lotto erstellen. Das ist sehr gut denn es ist sehr ähnlich dem was ich machen werde für unser computerspiel!

Für das hinzufügen von networks:

- `brownie networks add development mainnet-fork cmd=ganache-cli host=http://127.0.0.1 fork=https://eth-mainnet.g.alchemy.com/v2/ewQttDgneO9VNfgC0RuhMXoyort0z9Im accounts=10 mnemonic=brownie port=8545`
- Diesen command abrufen, natürlich die fork anpassen und andere sachen anpassen aber in der regel geht es so

23/08/2022:

Heute habe ich einen sehr blöden Fehler behoben nämlich beim installieren von networks zu brownie unter host= <http://127.0.0.1> ich hatte nach dem http noch ein s = https das hat dazu geführt dass beim testen von smart contracts, alles ausgeschaut hat als würde es funktionieren, es hat einfach minutenlang geladen, und ich dachte etwas wäre mit meinem pc falsch. Ich konnte es nur finden indem ich zurück gegangen bin in meinen commands und meinen command mit dem command aus dem tutorial verglich

Commands:

- `brownie networks delete mainnet-fork`
- `brownie test --network mainnet-fork`
- `brownie networks list`
- `mkdir = make new project`

24/08/2022:

Heute habe ich weiter dem tutorial gefolgt und hatte einen dummen fehler

Dann habe ich ein wallet aus brownie löschen müssen und es wieder hinzugefügt. Mein passwort für freecodecamp-account : d\*\*\*\*\*007

Ich denke wenn ich mit diesem nächsten lernprojekt fertig bin, werde ich die meisten sachen gemacht haben die ich auch für meine projekte brauchen werde, also dann legen wir los mit den smart contracts für das game.

25/08/2022:

Heute bin ich in ein paar sehr verwickelte errors gekommen, nachdem ich zuerst etwas code schreiben konnte. Ich muss nämlich einige sachen aus github importen, die brauchen jedoch andere solidity version, aber ich kann nicht überall andere solidity version haben und auch noch andere sachen sind verbugt. Ich bin jedoch ziemlich positiv dass ich die Lösung zu diesem error sehr bald finden werde. Am Wochenende versuche ich das zu lösen.

27/08/2022:

Habe grad geschafft den Error zu beheben. Ich hatte sehr viele Fehler doch am Schluss lag alles daran dass in einer source file die ich importiert habe, nochmals eine andere source file importiert wurde, die ich manuell importieren wollte und so entstand ein DeclarationError.

DeclarationError: Identifier already declared.

Ich musste deshalb einfach ein import statement herauslöschen und schon hat es funktioniert

Hier auf stackoverflow beschrieben, damit ich anderen developers helfen kann:

<https://stackoverflow.com/questions/73510845/declarationerror-identifier-already-declared-import-chainlink-contracts-src>

28/08/2022:

SEHR WICHTIG:

Eth\_usd\_price\_feed: <https://docs.chain.link/docs/ethereum-addresses/>

Und man muss das vom richtigen network nehmen. Das bedeutet dass wenn man auf rinkeby testnet macht, muss man das von der rinkeby testnet nehmen. Hatte einen blöden fehler hier, weil ich das falsche genommen habe.

Ich habe einen error bekommen: ValueError: Gas estimation failed: 'invalid opcode: INVALID'. This transaction will likely revert.

Doch dann hat sich herausgestellt, dass ich in meinem MetaMask wallet einfach keine link testnet tokens hatte. So habe ich es gelöst.

29/08/2022:

Ich habe die smart contract lottery abgeschlossen. Nun mache ich noch ein paar lessons und dann fange ich mit meinen Smart contracts an.

**Brownie bake chainlink-mix**, um alle sehr oft benutzten sachen schon zu haben

Heute habe ich meine eigenen smart contracts angefangen, da es ziemlich schwierig ist bin ich wieder back auf remix gegangen um mich nur um den code an sich kümmern zu müssen, aber bis jetzt siehst gut aus. Ich habe ein contract geschrieben das mir erlaubt von einer wallet auf eine andere eth zu schicken, was eigentlich schon ein grosser teil, ist als nächstes muss ich dass auf der testnet testen und dann können wir weiter funktionen implementieren.

30/08/2022:

Heute habe ich weiter an meinen smart contracts geschrieben. Jedoch bekomme ich immer einen Fehler wenn ich von einer wallet auf die andere schicken möchte, einen fees error.

Ich habe es geschafft, offenbar hatte ich irgendwelche fehler, habe auf einer website eine Anleitung gefunden. Und jetzt funktioniert, sehr gut!

<https://solidity-by-example.org/sending-ether/>

website für contracts zum ethereum sending

Nun habe ich einen Code (noch in remix ide) der erlaubt mir von einer beliebigen wallet den contract zu funden mit einer gewissen menge an eth, und dann zu einer anderen wallet diese geld zu withdrawen.

Da wir in solidity und auf der blockchain keine pseudorandom zahlen erstellen können, werde ich das mit random zahlen in vscode testen, da wir dort random zahlen von python nehmen können

Nächster schritt:

1. Von Vs code heraus von einer wallet einen contract funden und danach sofort zur gleichen wallet withdrawen
2. Zu einer anderen wallet withdrawen

31/08/2022:

Ich habe es geschafft aus vscode heraus, smart contract zu deployen, mit zwei funktionen: sendEther und FundingWithDollars, nun habe ich mit einem etherscan api es dort als ui benutzen und von dort heraus kann ich nun die funds zu verschiedenen wallets schicken.

Ich weiss nicht wie man es macht aber ich will eigentlich dass man contract deployed, und dieser diese leeren variablen quasi schon gefüllt hat mit den values? Keine ahnung wie man das macht. Aber ich recherchier mal.

Ich glaube ich hab schon erste Ideen wie wir es verbinden, und zwar müssen wir die smart contracts in unity hineinbringen und dort in der ui designen, das wird wahrscheinlich der einfachste weg sein, die Frage ist ob wir dann noch einen metamask connect button machen werden können, aber das wird sich noch zeigen

01/09/2022:

```
// SPDX-License-Identifier: MIT
pragma solidity ^0.8.7;

import "@chainlink/contracts/src/v0.8/interfaces/AggregatorV3Interface.sol";

contract ReceiveEther {

    receive() external payable {}

    fallback() external payable {}

    function getBalance() public view returns (uint) {
        return address(this).balance;
    }
}

contract SendEther {
    function sendViaTransfer(address payable _to) public payable {
        _to.transfer(msg.value);
    }
}
```

```

contract FundingWithDollars {
    AggregatorV3Interface internal priceFeed;
    mapping(address => uint256) public addressToAmountFunded;

    constructor() {
        priceFeed =
AggregatorV3Interface(0x8A753747A1Fa494EC906cE90E9f37563A8AF630e);
    }

    function amountInWeiNeeded() public view returns (uint) {
        (,int price, , , ) = priceFeed.latestRoundData();
        int256 amountInEth = 5000000000000/price;
        int256 amountInWei = amountInEth * 100000000000000;
        return uint(amountInWei);
    }

    function fund() public payable {
        uint256 fiveDollars = 3100000000000000 wei;
        require(msg.value >= fiveDollars);

    }

    function withdraw(address payable _to) payable public {
        payable(_to).transfer(address(this).balance);

    }

    function getConversionRate(uint256 ethInputed) public view
returns(uint256){
        ethInputed = amountInWeiNeeded();
        return uint256(ethInputed);
        //0.0000016151267820800
    }
}

```

Bis jetziger code, was ich heute wollte ist, dass nur gefundet werden kann wenn ein gewisser betrag geentert ist, das habe ich geschafft, man muss jedoch wei oder ether oder gwei hinzufügen.

04/09/2022:

Ich habe versucht, die Variabel für die definition von 5 dollar, inbegriffen, welcher eth price im jeweiligen Moment ist. Jedoch weiss ich nicht wieso erlaubt es mir nicht, die variabel einer anderen function die den ethereum price abruft gleichzusetzen. Es geht doch, aber nur auf der echten rinkeby testnet. Macht auch sinn weil wenn es nicht auf rinkeby läuft dann funktioniert die source file nicht. Also ab jetzt vor allem auf rinkeby testen.

Momentane funktionen die ich brauche:

- Funktion die ausrechnet wie viel ethereum 5 dollars sind
- Funktion um diese 5 dollars zu funden von seinem eigenen wallet
- Funktion um wallets die sich verbinden zu speichern
- Funktion um eine random wallet aus diesen die geld gegeben haben herauszusuchen
- Funktion um der ausgesuchten wallet das geld zu transfern

<https://github.com/jspruance/block-explorer-tutorials/blob/main/smart-contracts/solidity/Lottery.sol>

hat mir grade sehr geholfen zu verstehen und konnte einige sachen daraus implementieren.

Okay, jetzt habe ich es geschafft die ersten 3 der oben stehenden Punkte zu machen, sowie funktion welche und der ausgesuchten wallet geld überweist.

Nun habe ich alle Punkte oben geschafft. Als nächsts wird es darum gehen, zu überlegen wie wir es in einer UI darbieten können. Und wie wir es optimieren können. Welche zusätzlichen funktionen brauchen wir?

Ich will es aus vscode heraus deployen können. Das ist ein ziel, und es auch testen können. So lerne ich zu testen.

Todo's:

Sich ernsthaft überlegen welche funktionen wir noch brauchen.

Alles auf vscode transferieren und es im etherscan benutzen können.

05/09/2022:

Ich denke wir werden das verbinden so wie in diesem tutorial machen können.

[https://youtu.be/fSKCF\\_tSKQc](https://youtu.be/fSKCF_tSKQc)

Dann werden wir die contracts direct in die ui des spieles integrieren können.

Ich konnte es transferieren in vscode und es dort benutzen

Die nächsten Todo's:

- Wie kann unser contract usdt oder einen anderen erc-20 token annehmen
- Wie werden wir die adressen speichern können aber nur für ein spiel, um am schluss den richtigen spielern die tokens zu überweisen

<https://consensys.net/blog/developers/solidity-best-practices-for-smart-contract-security/>

06/09/2022:

Ende November erstentwurf von Begleittext schreiben und abgeben.

Ich würde sagen auf jeden Fall mehr Seiten als 2 nicht alles bis in die letzte Woche lassen

Gut überlegen wie wir Präsentation gestalten werden. Müssen wir zusätzliche Sachen und Themen recherchieren.

Jeder schreibt eigenen Begleittext jedoch überlegen ob wir einen gemeinsamen Teil schreiben wollen.

Zielpublikum Klassenkameraden, also müssen wir auch viele Begriffe erklären. Ein Laie soll es lesen können und verstehen was wir gemacht haben.

Von vorne an sauber mit Quellen Arbeiten, sehr wichtig!!! Im Quellenmanager verfassen.

Backups machen!!! Sehr wichtig.

Tomas: Das minimal viable product erstellen, damit sich zwei Spieler in einem Raum treffen können und einen Gewinner definieren können.

Zweiten Spieler und spielbar machen und definieren können wer gewinnt

07/09/2022:

<https://eips.ethereum.org/EIPS/eip-20>

<https://docs.openzeppelin.com/contracts/4.x/erc20>

wichtig für ERC-20 Tokens

Wie man eigentlich ein Smart Contract schreibt (in VS Code):

- Wenn man Sachen importiert braucht man eine `brownie-config.yaml`
- Man braucht ein `.env` um seine private keys zu sichern.
- Man braucht einen helpful script um dort seinen oft benutzten Code hineinzuschreiben

<https://ethereum.org/en/developers/tutorials/transfers-and-approval-of-erc-20-tokens-from-a-solidity-smart-contract/>

09/09/2022:

Heute habe ich einfach ein basic solidity tutorial mir angeschaut damit ich solidity wirklich verstehe.

11/09/2022:

Versuche zu verstehen wie ich mit ERC-20 Tokens arbeiten kann und diese verschicken kann

<https://www.geeksforgeeks.org/solidity-fall-back-function/>

12/09/2022:

Ich habe heute mich weiter über ERC-20 Tokens informiert. Denn das Problem das ich momentan habe ist das Verschicken von Coins. Und das wollen wir machen, denn wir möchten ja nicht mit ETH die Runden joinen sondern USDT oder BUSD. Und ich verstehe einfach noch nicht wie ich das machen kann.



Das Ziel diese Woche ist:

- Alles was ich bisher habe also funden und random withdrawen mit einem ERC20 token machen können. Danach geht's darum die smartcontract in unity zu implementieren.

13/09/2022:

Die Frustration wird grösser und grösser, da ich keine Ahnung habe wie ich in meine smart contracts ERC20 tokens hineinfunden kann. Ich habe sehr viele tutorials angeschaut und auch viele website gelesen aber keine davon hilft mir das zu machen was ich machen möchte.

Ich glaube es geht darum wie in diesem video gezeigt, dass man mit dem offiziellen coin approve muss.

<https://youtu.be/FQe91txqP6k>

Also könnte ich mir theoretisch ein coin deployen der genau diese funktionen hat. Und diesen dann benutzen um die contract zu schreiben. Den auf der mainnet hat USDT die approve funktionen

!!!! Ich habe geschafft was ich machen wollte, nun verstehe ich wie ich es machen kann. Das ist sehr gut. Ich muss nämlich bei dem ERC20 den ich in den smart contract schicken möchte, denn muss ich auf der blockchain im contract des coins approve. So geht es, ich habe den coin den ich gestern deployed habe nun in mein smart contract gefunden, das ist sehr nice!! Erfolgreiches Frustieren.

14/09/2022:

Ich habe heute probiert aus meinem contract heraus ein anderen contract zu callen. Um dort zu approve damit ich ERC20 tokens in den contract schicken könnte. Jedoch funktioniert es leider noch nicht. Ich habe aber viel herumprobiert und bin zuversichtlich dass es gehen muss.

17/09/2022:

(how to accept ERC20 tokens stackexchange, n.d.)

Heute versuche ich ERC20 tokens aus meinem contract heraus zu approve. Keine Ahnung wie das geht, ich kann es nicht finden.

Ich glaub ich habe es, es geht gar nicht dass ich aus einem smart contract ein approval auf einem anderen smart contract mache. Man muss es einfach auch in das front end einbauen. Sofort nachdem man auf enter with five dollars drückt kommt ein anderer button, der approve USDT heisst und nachher kommt sofort der button transfer 5 dollars. Ich muss einfach den smart contract des ERC20 tokens in das front end einbauen.

Also jetzt geht's darum zu integrieren. Es wäre am einfachsten fürs verbinden wenn ich es ins unity integrieren könnte.

18/09/2022:

<https://forum.openzeppelin.com/t/approve-erc20-token-on-another-contract/4018/2>

Hier dieser link zeigt, dass nur der holder der tokens diese approven kann. Es kann nicht ein smart contract es für den holder machen.

```
// SPDX-License-Identifier: MIT

pragma solidity ^0.8.7;

import "https://github.com/OpenZeppelin/openzeppelin-
contracts/blob/master/contracts/token/ERC20/ERC20.sol";

interface USDC {

    function balanceOf(address account) external view returns (uint256);
    function allowance(address owner, address spender) external view returns
(uint256);

    function transfer(address recipient, uint256 amount) external returns
(bool);
    function approve(address spender, uint256 amount) external returns (bool);
    function transferFrom(address sender, address recipient, uint256 amount)
external returns (bool);
}

contract transferUSDC {
    USDC public USDC;
    address owner;
    mapping(address => uint) public stakingBalance;
    address payable[] public players;
    uint256 public counter;

    constructor() {
        USDC = USDC(0x281b6DDD31b86D4F564fff336f69fA1B004Ef694);
        owner = msg.sender;
    }
    function depositTokens(uint $USDC) public {

        // amount should be > 0

        // transfer USDC to this contract
        USDC.transferFrom(msg.sender, address(this), $USDC * 10 ** 18);

        // update staking balance
        players.push(payable(msg.sender));
        stakingBalance[msg.sender] = stakingBalance[msg.sender] + $USDC * 10
** 6;
        counter++;
    }
}
```

```

    function transferTokensToOtherWallet(address sender, address recipient,
uint256 amount) public{
        USDC.transferFrom(sender, recipient, amount);
    }

    // Unstaking Tokens (Withdraw)
    function withdrawTokens() public {
        uint balance = stakingBalance[msg.sender];
        uint index = getRandomNumber() % players.length;
        // balance should be > 0
        require (balance > 0, "staking balance cannot be 0");

        // Transfer USDC tokens to the users wallet
        USDC.transfer(players[index], balance);

        // reset balance to 0
        stakingBalance[msg.sender] = 0;
        players = new address payable[](0);
    }

    function getRandomNumber() public view returns (uint) {
        return uint(keccak256(abi.encodePacked(owner, block.timestamp)));
    }

    modifier onlyOwner() {
        require(msg.sender == owner);
        _;
    }

    function approveTheTokens() public pure returns(string memory){
        string memory messageOut = "You have to call the approve function in
the smart contract of the Erc20 Token!";
        return(messageOut);
    }
}

```

Das benutzte ich um in unity meine contracts zu haben <https://github.com/MoralisWeb3/unity-web3-game-kit/releases/tag/v1.2.10>

Ich habe es leider noch nicht geschafft es ins unity zu integrieren, bin aber schon in unity dran es zu machen.

Nächste ToDo's:

- In Unity das UI bauen für meine contracts, die buttons, scenes etc bauen
- Mein smart contracts im vscode deployen um das abi von etherscan zu kriegen

- Danach mit moralis und dem tutorial probieren es zu verbinden

21/09/2022:

Heute habe ich weiter daran gearbeitet meine contracts in unity zu integrieren. Ich habe das chainsafe sdk heruntergeladen, das ist ein package welches erlaubt contracts zu integrieren. Ich versuche momentan ohne moralis alles hinubekommen, weil ich will das game nicht noch von einer weiteren thirdparty abhängig machen.

Das einzige was etwas nervt beim testen ist, dass ich immer ein webgl build generieren muss, und das dauert ab und zu 2 min andersmal nur 16 sekunden.

Ich habe es grad geschafft aus der unity webgl einen erc20 token an eine andere adresse zu schicken

Erklärung: Ich habe aus der unity quasi auf der blockchain etwas verändert... das ist schon mal sehr gut.

Nächste todos, es dazu bringen dass wir in den contract hineinschicken

Ich weiss was ich machen muss, ich muss ein contract machen, welcher sich aufschreibt wenn man ihm funds schickt, der denn sender sich notiert. Extrem wichtig, dass wir das haben, ansonstent wird spiel nicht laufen

22/09/2022:

Ich habe geschafft was ich ständig machen wollte. Jetzt kann ich aus unity heraus approve auf der blockchain und was ich jetzt noch machen muss ist, deployen der endgültigen smartcontracts und die integrierung von denen in unity, und dann haben wir das minimal viable product.

Danach zu tomas schicken und der integriert es in sein part und dann haben wir eine Betaversion des spiels!

23/09/2022:

Heute versuche ich die integrierung fertigzustellen. Ich versuche die scenes so vorzubereiten, sodass ich nachher Tomas alles schicken kann und wir es verbinden können.

Was noch ein Problem darstellt, ist dass wir es so kreieren wollen, dass mehrere Runden gleichzeitig laufen können. Das Problem ist, dass alle in den gleichen contract hineindeployen. Und dann wird alles durchmischt. Die einfachste Lösung welche mir in Sinn gekommen ist, ist genau das durchmischen zu erlauben. Alle funden in den gleiche contract hinein. Und dann haben wir functions, welche gecallt werden wenn man ein sieger, oder zweiter platz und so weiter ist. Das wichtige ist, dass wir irgendwie müssen wir ein server benutzen der die contracts funktionen callt, asnonsten sind wir nicht protected gegen diebstahl, wenn es nicht als onlyOwner definiert wäre.

24/09/2022:

Ich habe ein wenig nachgedacht und realisiert, dass wir es irgendwie machen müssen, dass nur jemand der coins hineindeployed hat, auch die funktionen zum herausnehmen, callen darf. Und die Personen müssen auch winners sein, das heisst es braucht eine funktion welche eine gewisse wallet

als winner definiert. Und als 2place definiert und so weiter. Wir brauchen also diese funktionen noch:

Funktion für jeden platz, 1sten 2ten usw. Momentan machen wir nur 2 plätze, fürs minimal viable product

Funktion welche von einem onlyOwner gecallt wird, und welche bestätigt dass die Person welche withdrawen möchte wirklich ein sieger oder ein 2tplatz ist

Und wir brauchen noch eine funktion welche hasApproved besagt, damit wir nur dann die szenen wechseln.

⇒ Das hasApproved geht nicht sehr einfach, weil wir die funktion approve nicht in unserem contract callen

Ich habe nun alle funktionen welche ich noch brauchte integriert ausser die hasApproved, aber diese ist auch nicht so wichtig für die sicherheit des spiels

Morgen muss ich diesen neuen smart contract ins unity bringen.

25/09/2022:

Heute habe ich alles was ich gestern neu erstellt habe hineingebracht ins UI das einzige was ich jetzt noch bleibt zu machen, ist es zu automatisieren, dass sich jeder, nicht nur der owner des contracts verbinden kann, und je nachdem ob er gewinnt, die funds zugeschickt bekommen.

Ansonsten funktioniert alles sehr gut und ist eigentlich ready um verbunden zu werden mit Tomas' spiel. Morgen rede ich mit Tomas wann und wie wir es genau machen möchten. Die eine Frage wegen des automatisierens bleibt noch, und danach können wir versuchen es aufs BUSd network zu bringen, um es mit echtem geld zu machen.

Was noch zu machen bleibt, sind die funktionen für jede position, gewinner 2t platzierter usw.

Um es ein profitables spiel zu machen, müssten wir natürlich noch eine kleine provision zu unserer eigenen wallet nehmen.

Aber das alles sind noch extras und das minimal viable product ist eigentlich fast fertig, bis Ende Monat haben wir es!

26/09/2022:

Minimal viable product mit zwei Spielern erstellen, zum alles durchtesten, skalieren können wir später immer noch.

Nächster wichtiger Todo:

Automatisation der bestätigung des winners durch ein thirdparty program.

Unentschieden? Verhindern, oder abwägen, kommerzieller apsekt, verliert man fee.

Der Fall unentschieden muss abgewägt werden. Was für zusätzliche Probleme könnten entstehen.

Spätestens Mitte November integration und fertig mit MVP

Beim Schreiben, was habe ich selbst gelernt, mit abläufen darstellen. Den Kern vom Konzept schreiben, warum habe ich das so gemacht wie ich das gemacht habe.

Verweisen,

Schon den schriftlichen Teil anfangen zu schreiben. Fokus Maturarbeit.

Präsentation inhalt visuell, wie präsentier ich das?

Auf Referenten zu kommen wenn wir ready sind mit einer ersten Rohfassung der Arbeit und eventuell erste Ideen für Präsentation. Von Anfang an gut mit Quellen arbeiten.

Konzentrieren auf was wir gelernt haben, den Lernprozess den herausfind-prozess aufzeichnen

27/08/2022:

Heute habe ich versucht, dass unser game die confirmation automatisch macht, damit es gesichert ist von diebstahl auf der blockchain.

Ich muss noch ein wenig hier herumprobieren, und mir den Code irgendwie zusammensetzen

<https://docs.gaming.chainsafe.io/#custom-interactions>

Was ich eigentlich machen muss:

- Private key in code, und so die function im cs script callen, und nicht dass das popup kommt. Das müssen wir vermeiden, wir müssen dem game zugriff auf meine wallet geben, und dann die function von dort aus callen.

07/10/2022:

Ich struggle immer noch mit der externen confirmation einer onlyOwner wallet des smart contracts um sicherheit im system zu schaffen

<https://ethereum.stackexchange.com/questions/121008/web3-best-practice-for-calling-onlyowner-view-from-backend>

Das ist sehr frustrierend, denn es wirklich eigentlich dieses Problem und dann können wir integrieren. Wir müssten quasi die function callen schon mit der wallet und mit der confirmation, nicht dass es danach noch manuell gedrückt werden muss.

Ich habe ein wenig nachgedacht und vielleicht können wir dieses Problem in einem sehr eleganten weg lösen. Wir bräuchten einen eigenen erc20 coin, oder nft. Mal schauen, wahrscheinlich besser mit eigenem erc20 token. Und zwar, wenn bestätigt wird, wird unserer coin geschickt von der



onlyOwner wallet, und nachher hat ein spieler diesen coin, und ihm austausch zu diesem coin kann man dann seine funds withdrawen.

Weiteres nachdenken: Hätten wir irgendwelche dummen situationen? Das jemand diese tokens speichern könnte um später erst zu withdrawen? Aber eigentlich nicht weil es ist eine mathematische bilanz. Solange so viel eingezahlt wie withdrawt wird, wird es funktionieren.

Also zusätzliche funktionalität zum einbauen:

Die withdrawal function so machen dass man unsere coin braucht um withdrawen zu können

08/10/2022:

Nach ziemlich viel herumstruggeln habe ich es geschafft, das einzige leichte problem ist, dass man relativ lange auf confirmations warten muss, aber vielleicht könnte ich wenigstens das approval für unseren ingame coin wegmachen, obwohl, dann haben wir keine security mehr.

Es funktioniert so weit, jetzt ist es das ziel, alles ins unity zu integrieren, zu testen und dann haben wir zwei minimal viable products, welche danach verbunden werden müssen.

Heute habe ich es fast geschafft mein Teil vom MVP fertig zu stellen, jedoch habe ich noch eine schwierigkeit zu beheben, ich weiss wie, mache es aber morgen oder irgendwann sonst, und zwar was zu machen ist

ToDo:

Für die letzte withdrawal winner function, ein argument hinzufügen

Für alle anderen player auch eine withdrawal function schreiben, alles in remix testen, auch dass die funds wieder zurückkehren, zur admin wallet

Alles wieder schön in unity integrieren und fertigstellen. Ev UI etwas schöner machen, aber das ist ein nice to have

Lieber vorher UI mit backend interagieren lassen, damit die szenen erst gewechselt werden wenn doe transaction durch ist

09/10/2022:

Ich muss schnell überlegen was ich machen muss:

1. Die function abändern und argument hinzufügen, MVP für 2 personen.
2. Danach den contract deployen und den neuen contract integrieren

Es ist mir gerade ein neuer seher dummer fehle raugetaucht, es schaut nämlich aus, als ob etwas das ich habe und das immer funktioniert hat, nämlich das verifizieren von contracts im etherscan plötzlich nicht mehr funktioniert

Sehr dummer fehler bin seit 2 stunden an dem fehler dran und kein schritt weiter gekommen ,  
offenbar hat etwas tief im infura oder etherscan aufgehört zu funktionieren, denn ich habe es  
versucht mit alten projects wo es ohne probleme lief, und nun geht es nirgendwo mehr, ich habe  
versucht neuen account für etherscan herzustellen funktioniert aber nicht

Und jetzt nachdem merge kann man nur noch goerli testnet benutzen, anstatt rinkeby, kovan  
ropsten etc. Also ist es einfach ein pain, ich wäre ansonsten heute fertiggeworden ,s o kämpfen wir  
doch noch ein wenig weiter.

10/10/2022:

Ich habe ein wenig nachgedacht wie ich diesen blöden Fehler beheben kann, er ist ganz sicher nicht  
etwas das ich ausgelöst habe. Es ist entweder etwas mit der testnet kaputt, aber ich denke es ist  
eher etwas mit brownie kaputt, brownie ist eine library/tool welches mir erlaubt contracts viel  
einfacher in vscode zu deployen, es hat nämlich unter der list der networks, wenn man auf goerli  
testnet drückt kommt error page not available, während bei anderen testnets(welche leider nach  
dem eth merge deprecated sind) kommt die website auf.

Also wie lösen wir es:

Schauen ob es reagiert wenn wir den infura id wechseln

Brownie löschen und nochmal installiere, vielleicht neuere version

Jetzt will gar nichts mehr funktionieren, nicht einmal mehr verifien und anzeigen dass es nicht  
funktioniert will es. Kein plan wie ich dass fixen soll.

Ich probiere jetzt alles auf der bsc testnet, hoffentlich geht es besser, vielleicht ist es mein wlan  
weshalb es so langsam

⇒ Ich glaub ich weiss wo das problem lag. Erstens heute, dass es mir gar nicht erst angezeigt  
hat, das hat glaube ich mit dem etherscan an sich zu tun, ich glaube ihre server sind einfach  
lahm. Und das andere, dass es mir plötzlich von einem Tag auf den anderen aufgehört zu  
verifien, es liegt am WLAN hier auf Mallorca, weil mein Bruder auf eine website nicht  
kommen konnte, er dachte die website wäre down genommen worden, aber dann hat es  
über mobile daten funktioniert.

Ich probier morgen noch mal über hotspot

11/10/2022:

Offenbar lag es doch an etherscan alleine, denn heute geht es sogar über das Wlan hier, keine  
Ahnung wo das Problem lag, sicher nicht an meinem Code, denn ich hab eigentlich nichts geändert.  
Sehr frustriert, aber zeigt wieder dass die ganze crypto space noch nicht sehr saturated ist und es  
noch sehr viel verbesserungsoptionen gibt.

Ich hab es mein Minimal Viable Product. Jetzt geht es darum es mit Tomas zu verbinden. Ich werde  
ihn Fragen ob er ein github repository für sein part machen kann, damit ich es nachher verbinden  
kann. Ich glaube es macht mehr Sinn wenn ich es zuerst erstmal probiere provisorisch zu verbinden  
und danach erst mit Tomas versuche. Aber ich schaue mal.

Nächste ToDo's:

Tomas repository haben, während ich darauf warte, noch eine zweite function implementieren, und die bilanz aufstellen für zwei spieler jedoch nur.

Danach einfach die verschiedenen scenes in die richtige reihenfolge bringen, und dort wäre es halt schonoch gut wenn wir ein abhören der blockchain ind er frontend hätten

Danach wallet mit den jeweiligen playern verbinden, sollte kein allzu schwieriges mapping sein

Und danach hinten die winnerwithdrawal szenen anhängen und den ingametoken überweisung ins backend kriegen

Danach noch backto first scene buttons kreieren

Denken und recherchieren: Kann man ein erc20 ohne approval machen, damit wir unser ingame token so machen können == der player spart zeit beim approven

15/10/2022:

Ich habe heute versucht Tomas file auf meinem Pc zu öffnen und es geht, ich habe in einem file, etwas kommentiert und zwar createcontoller() function weil es einen error gemacht hat, ich weiss nicht was die konsequenzen davon sind, aber so funktioniert das game wenigstens.

Aber ja das Ziel ist das game klar auf meinem PC zum runnen zu bringen und danach mit meinem part verbinden, ich muss schauen ob die webgl build machbar sind anonsten müssen wir mein part ein wenig umbauen damit man sich mit dem handy verbinden kann. Aber auf dem web ist halt schon schöner.

Der webgl build hat nach 5 minuten funktioniert, aber es hab ein error, der auch beim nicht webgl build gekommen ist.

Es hat ein problem und zwar dass wenn ich das game in unity runnen lasse, bei der game scene ein standbild ist, es hat ein paarmal funktioniert aber jetzt funktioniert es irgendwie nicht. Es ist auch nicht sauber importiert. Wieso können wir es nicht als schöne github repository machen.

Irgendwie funktioniert der game part nicht. Ok es funktioniert, aber ich muss vorher das gameplan anhalten, und dann wieder laufen lassen dann funktioniert es. Aber der client ist nicht connected

Irgendiwe zählt der timer links oben von 9 herunter und sobald er stoppt haben wir denn error weil quasi das game beendet wird aber trotzdem noch das synchen von szenen im hintergrund läuft

<https://forum.unity.com/threads/operation-set-252-not-called-because-client-is-not-connected-and-i-have-one-more-problem.887458/>

Ah ja jetzt sehe ich es im script timer, ist nur 10 sekunden als float amount of time gesetzt, perfekt

Kein Ahnung, auf jeden Fall Zeit auf baldigen Call mit Tomas sobald ich in der schweiz bin am Mittwoch, bis dann kann ich noch etwas herumprobieren, oder mein Teil so ausrichten dass er auf die blockchain hört, oder auch einfach anfangen zu schreiben, einen groben Plan machen

16/10/2022:

Heute habe ich versucht Tomas part auf meinem PC zum laufen zu bringen, doch es will nicht funktionieren, also habe ich entschieden es zu löschen und nochmals zu downloaden, dch dann habe

ich mir selbst irgendwie den zugriff auf den google drive verweigert, egal. Ich habe jetzt ein github repository für mein unity project gemacht also kann eigentlich tomas jetzt mein part auf seinem pc zugreifen.

<https://github.com/nuevofilipo/SmartContracts-Game>

offenbar geht es nicht umgekehrt mit Tomas projekt auf github weil zu grosse file

18/10/2022:

Ich habe gestern angefangen mein Begleittext zu schreiben. Und ich muss aufpassen, dass ich nicht zu viel erzähle sondern lediglich informtaiv beschreibe, was ich gemacht habe. Ich habe einen groben Plan gemacht was ich machen werde, mit grobem inhaltsverzeichnis.

Ich versuche jetzt meine unित्र smart contracts so auszurichten, dass es funktioniert dass wir auf transactionen warten können. Weil ansonsten ist das spiel eigentlich nicht von usern spielbar, denn die scene wechseln sich einfach wenn wir auf die buttons drücken und nicht wenn die transaktion, bzw bestätigung schon durch ist.

Das ist aber eigentlich eher ein nice to have, damit das spiel wirklich von allen spielbar ist aufdem internet, und es ist nochmal sehr viel arbeit das nochmals zu machen.

19/10/2022:

Ich habe weitergeschrieben an meinem Begleittext

20/10/2022:

Nachträglich eingetragen

Wir haben einen call gemacht und darüber diskutiert wie wir verbinden möchten. Ich kann es machen, jedoch war das projekt beim letzten mal verbuggt. Tomas sein game hat nun 10GB grösse, und weil diese grösse vor allem von den Graphics stammt habe ich vorgeschlagen dass er die file kopiert und ledihlich eine macht mit einer einfachen flachen plattform als lanschaft, damit ich integrieren kann.

21/10/2022:

Ich habe meine github repo ja vor ein paar tagen gemacht. Nun werde das Projekt so optimierne, bzw alles herauslöschen was ich nicht benutze. Dann werde ich es Tomas schicken und das integrieren wird sehr einfach sein, da er lediglich die richtigen Scenes an der richtigen Stelle anschliessen muss.

Es macht auch einfach mehr Sinn, dass er mein Teil in seinen integriert da sein Teil extrem gross ist

Todo's:

Smart Contract deployed mit gewinner sowie zweitplatziertem:  
0xd75F5371Da7BD82fC228C5735cDf96d8361c629B

Genau das in unity integrieren

Für gewinner sowie zweiten jeweils eine scene machen

Winner admin transfers 7

Looser admin transfers 3

Both enter with 5 dollars

22/10/2022:

Ich habe alles was ich mir aufgeschrieben habe gestern, gemacht. Nun hat mir Tomas seinen part geschickt und ich werde versuchen meinen und seinen zu verbinden

23/10/2022:

Ich habe meine guthub repository auffgefrischt, ich habe Folders gemacht mit den assets welche ich actually brauche.

Ich glaube es wäre viel einfacher wenn Tomas einfach meine git kopiert und bei sich schnell die scenes integriert. Später call

<https://github.com/nuevofilipo/SmartContracts-Game/tree/main/SmartContract%20withsdk%20-%20Copy/Assets>

24/10/2022:

Ich habe heute weitergeschrieben an meinem Begleittext und ein Schema der Smart Contracts integriert.

Was ich machen muss:

- Den Code clean machen und richtig kommentieren
- Das ins Unity hineinbringen
- Die Scenes für Fünf Spieler sowie die Bilanz aufstellen, und das senden der InGame Tokens ins BackEnd bringen
- Weiterschreiben

26/10/2022:

Heute habe ich versucht meine Smart Contracts auf die events auf der Blockchain hören zu lassen. Doch ich weiss gar nicht wie es gehen könnte. Ich versuch später vielleicht aus diesem artikel etwas herauszulesen:

<https://docs.nethereum.com/en/latest/unity3d-smartcontracts-getting-started/>

Ich hätte vielleicht eine Lösung für diese problem, und zwar, dass ich noch einen erc20 token calle. Und was viele dapps machen ist, extrem hohe approvals zu setzen, dass erlaubvt später zeit zu sparen, also vielleicht on entry into application, do approval, damit wir später im spiel prozess zeit sparen können. Also was ich machen kann ist genau das. Während der spieler deposited, wird vom smart contract heraus ein erc20 token auf seine wallet herüberschickt.

Danach kann ich eine funktion abrufen, welche die balance des coins prüft, und erst das zulassen in die lobby erlaubt sobald diese balance erkannt ist. Am ende werden vom spieler aus halt alle tokens im gleichen moment zurückgeschickt

Nicht so elegant gelöst, aber die infrastructure welche ich benutze hat das viewen von events einfach noch nicht eingebaut

27/10/2022:

Enter Token : 0x59cDC482370C36CC55162926C4270570894b2cAD

Was ich probiert habe zu machen, ist dass ich tracken kann wie oft eine gewisse wallet schon deposited hat

Aber ich habe geschafft mehrere approve function in gleichen button zu bringen. Und nur alle 10 spiele callen zu müssen.

28/10/2022:

Neuste version des contracts: 0xCc626161FED2CB565D306e715CaaC95864801B2D

Ich habe es geschafft eine funktion zu machen, welche ein mapping updated und trackt wie viele male ein spieler gespielt hat

Und habe eine liste gemacht, welche beim ersten mal wenn ein spieler spielt, ihn dort hineinfügt

Ein allPlayers liste

29/10/2022:

Goerli testnet ist wirklich schlecht

Wenn man in einem faucet pro 24 h 0.05 goerli eth bekommt, aber eine transaktion in fees 0.4 eth kostet, dann was ist das überhaupt, das ganze system ist kaputt

Zeit auf binance testnet zu gehen

Ethereum chain ist wirklich schlecht, also eigentlich unmöglich, dass langfristig darauf entwickelt wird, andere chains werden ethereum ablösen müssen. Wenn man auf eth ein smart contract auf der mainnet deployen möchte, dann kostet dass um die 200 dollars, ein deployment!!!!

Addresses:

- testBUSD: 0xbCe98d116cA02A87a2E6c8EDf9597CEd50f3B0a2
- IGTOKEN: 0x5f310227dd9a9e65DaEb9d92282E27DD0eFcA02E
- ENTR Token: 0xBB7DFc1aBbd94d53648e9DF1F7584B898b1D57C2
- gameContract: 0x7d5A6F6313633Ba8DdbbE893f893761C90dab0eA



31/10/2022:

Ok perfekt jetzt habe ich es auf der Binance smart chain integriert

Jetzt muss ich die Funktionalität integrieren, sodass nur, wenn der player vom entrtoken eine gewisse menge hat, er das spiel joinen kann

Weil es eine async funktion ist kann ich den return der wallet balance irgendwie nicht loopen

01/11/2022:

Ich bin in der regel mit meiner Integration fertig, ich könnte noch das warten auf den Token mit dem letzten Button machen, noch eine animation hineinbringen und so weiter, aber ich denke, dass das erst eher ein nice to have wäre nachdem ich auch fertiggeschrieben habe

Morgen versuche ich meinen Teil mit Tomas seinem zu verbinden und dann schaue ich weizer

05/11/2022:

Ich habe verstanden wieso es immer so gelagt hat, nachdem ich meine contract scenes hinzugefügt habe. Der Grund ist, dass dann in der hierarchie, die reihenfolge der scenes anders ist, man müsste also überall im Code wo man die scene angibt, mit loadscene(0, 1, 2, etc) es austauschen, oder halt einfach hinten anhängen die approve und pickwinner scenes, aber im webgl build sind diese dann nicht am richtigen ort

06/11/2022:

Ich habe es geschafft zu verbinden!

Jedoch muss ich mich noch mit Tomas zusammensetzen, damit wir die scenes auch in richtige reihenfolge bringen können für den webgl build, sowie die entscheidung wer ein gewinner ist, und gegebenenfalls in die richtige scene gebracht werden.

To Do's:

- Scenes für erstplatzierten und zweitplatzierten machen
- Aber zuerst bilanz aufstellen
- Tomas Gewinner Tabelle machen und nachher jeweils in die richtige scene leiten
- Scenes in richtige Reihenfolge bringen, sowie überall abändern damit es funktioniert
- Erst wenn alles wirklich funktioniert, webGL build machen

Ich habe probiert auch den withdraw funds warten zu lassen, hat jedoch irgendiwe nicht funktioniert, muss es morgen nochmals probieren, aber eigentlich haben wir es fast verbunden, nur noch diesen einen, nämlich, dass spieler in die richtige scene geleitet wird

10/11/2022:

Ich habe eigentlich schon alles verbunden, das einzige problem ist dass:

- Wir noch keinen gewinner definieren können
- Und dadurch nicht in die entsprechenden scenes gehen können
- Ich weiss ausserdem nicht wie ich einen webgl build den ich schonmal gemacht habe, nochmals runnen kann, da unity einen localhost erstellt, aber ich später nicht weiss wie man einen localhost erstellen kann.

Ein komplett dummes Problem welches heute aufgetaucht ist, ist, dass die binance testnet nicht funktionieren möchte in meiner wallet, ein kompletter scam, was soll ich jetzt machen? Ich habe probiert die testnet nochmal hinzufügen funktioniert aber immer noch nicht. Auf goerli möchte ich nicht zurück wär ausserdem zu viel Arbeit.

Kein Plan wieso, aber es funktioniert nicht. Ich konnte mit einem leicht anderen rpc eine bsc testnet hinzufügen jedoch zeigt es mir dort bei jedem meiner coins die balance 0

Ich schaue morgen , oder am Samstag, weil das nervt mich wirklich

13/11/2022:

Ich habe es geschafft, offenbar funktioniert der server von diesem rpc nicht, ich musste auf einer website einfach einen funktionierenden rpc finden

Die website: <https://chainlist.org/chain/97>

Meine integrierten contracts welche schon perfekt funktioniert haben, haben nun aufgehört zu frunktioniere, die ingame transaction will gar nicht mehr, das abwarten auf die balance eines gewissen tokens dauert 2 min während sie vorher 10 sekunden gedauert hat

Nichts will mehr funktionieren, meine integrierten smart contracts wollen einfach nicht funktionieren.

Kein plan was ich mach soll, ich versuche vielleicht wieder auf goerli hinüberzugehen, oder ich versuche einfach von neu aus die integration der smart contracts, aber irgendwie fällt es momentan in sich zusammen

15/11/2022:

Ok perfekt, alles funktioniert wieder, cryptotechnologie ist komisch ab und zu funktioniert es einfach nicht

Das heisst worauf ich jetzt warte, ist dass Tomas seinen Gewinner Anschluss macht, und dann kann ich mich verbinden

Ok, mein minimal viable product ist fertig

Das einzige problem während der präsentation könnte sein, dass unser testnetwork nicht funktioniert, was recht risky ist

27/12/2022:

Endspurt todo's:

Den code nochmals durchschauen, macht wirklich alles so Sinn

Text fertig schreiben, Begriffe korrekt definieren, quellen richtig angeben, gendern

Kann man wirklich nicht Geld waschen? Wenn ja wie könnten wir es in zukunft lösen, geht es überhaupt mit einem solchen system

Brauche ich wie die Idee entstanden ist?

02/01/2022:

Endspurt #2 :

Ich habe nun ein Vorwort und die Einleitung gefällt mir recht gut

Ich muss Code online sowie ausgedruckt haben, überlegen, muss ich noch irgendwelchen code, zum beispiel, den mit chainsafe sdk, auch abgeben, ja wahrscheinlich schon

Ich würde sagen, ich entferne alles aus diesem file was ich nicht brauche

Ok weiter zum begleittext, den muss ich kürzen, wirklich nur das nötigste und dass sich nichts wiederholt

Fussnoten integrieren

Schluss teil:

Fazit, bin ich zufrieden mit meinem projekt

Was würde ich anders machen, was habe ich gut gemacht, was rate ich anderen developers,

welches war die grösste schwierigkeit, bin ich mit dem physischen endprodukt zufrieden, was würde ich anders machen

bin ich mit dem prozess zufrieden, welches war ein vorteil zu zweit zu arbeiten, welches ein nachteil

03/01/2022:

Feedback tomas: