S2BCacademy

Blockchain & Solidity

Whitepaper

Project: Tamacoinchi

Team: Group 7

Member: Enis Tola, Florian Huber, Alexander Parr

● S2BCconsult

● S2BCacademy

● S2BCcoach

MORPHEUS LABS SEED

PROJECT NAME: TAMACOINCHI

What is the problem?

The older ones of us still know them, the Tamagotchi from the 90s. These presented a virtual chick on a small electronic device, which the owners had to take care of by feeding it after it hatched. In the absence of affection, they could even die, and the owner had to start again. This was, among other things, very frustrating for the owners over time and so the Tamagotchi disappeared.

In addition, the world is moving to a blockchain-first society. Blockchain itself is rather difficult to understand and interact though. Children, elderly people and those who struggle with understanding new technologies and therefore usually avoid them can take their first steps with the blockchain as it is hard to learn and understand.

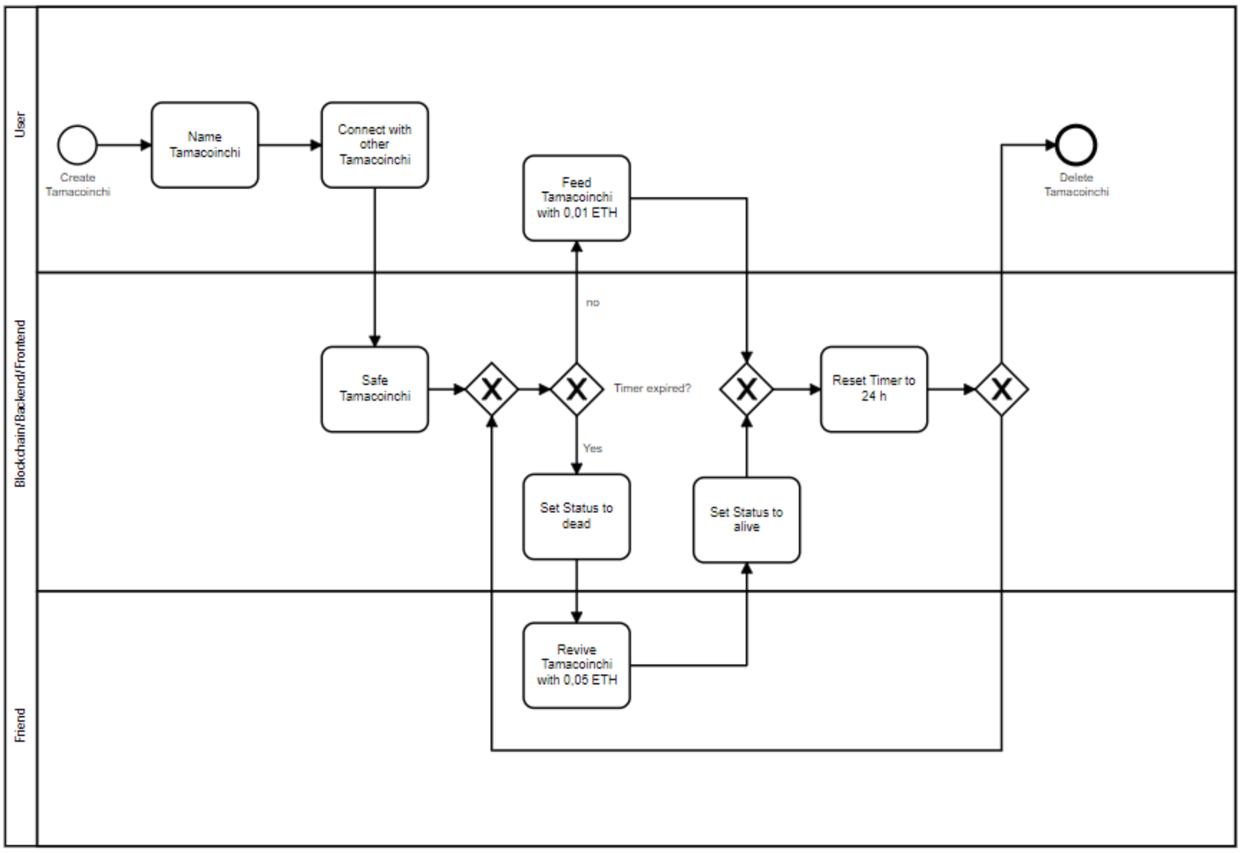
What is the solution?

With the help of the blockchain technologic we connect decentral your Tamacoinchi with those of your friends. Now everyone can view each other's progress. In addition, a cooperative aspect has been added to solve the problem of dying and thus frustration. If a Tamacoinchi dies due to lack of care, it can only be revived by its friends.

Through a specialised, simple game that includes social aspects, people like children, elderly people and those who struggle with understanding new technologies can get their feet wet with the blockchain. Easy usage makes for a good teacher.

How will it work?

Users can create their own Tamacoinchi on the blockchain and give it a name. These can then be linked to other Tamacoinchi. 1 time per day, the Tamacoinchi must be fed with an amount of e.g. 0.01 ETH. If it is not fed, it dies. Now it can only be revived by friendly Tamacoinchi with an amount of for example 0.05 ETH.



ELEMENTS OF THE APP

BackEnd:

Smart Contract: Tamacoinchi.sol

Tamacoinchi is the Smart contract for the pet -> Blueprint for the pet

Variables:

address payable creatorOfThisContract: The creator of the contract and the one to earn money with every transaction happening.

address public owner: Owner of the Tamacoinchi (pet)

string public ownerName: Name of the Owner of the Tamacoinchi (pet)

string public name: Name of the Tamacoinchi (pet)

bool public isMale: Gender of the Tamacoinchi (pet) -> true=male, false=female

uint public lastTimeFed: A timestamp displaying when the Tamacoinchi has been fed the last time. The “life” values in the frontend are derived from this variable. It is one of the most important ones.

mapping(address => bool) public owners: A mapping of owners, used to restrict functions to certain people -> modifiers

modifier ownerOnly / nonOwnerOnly: modifiers to restrict functions to certain people

Functions:

function feed(uint currentTime) public payable ownerOnly: A function to feed the Tamacoinchi. Restricted to the owner. The currentTime parameter is used to compare to the lastTimeFed variable, calculate how much time has passed and how much 0.01ETH will increase the “life” parameter. 0.01ETH increase life by 20% by increasing lastTimeFed by a uint value. Only owners are allowed to feed their pets.

function revive(uint currentTime) public payable nonOwnerOnly: Comparable to feed, except its restriction to non-owners. Revives your pet and sets lastTimeFed to the currentTime. It costs 0.2ETH to revive the Tamacoinchi.

function getSummaryOfYourPet() returns address, address, string memory, string memory, bool, uint: A function to retrieve your Tamacoinchis data.

Smart Contract: PetCreator.sol

PetCreator is the factory class to create Tamacoinchis.

Variables:

address[] public pets : The array of created pets.

mapping(address => bool) public owners: A mapping of owners, used to restrict owners from abandoning their Tamacoinchi by creating a new one. Every Owner can have 1 Tamacoinchi at max.

Functions:

function checkIfOwnerAlreadyHasAPet(address tobeCheckedOwner) public: Check if owner already has a pet. If so, he can't create a new one.

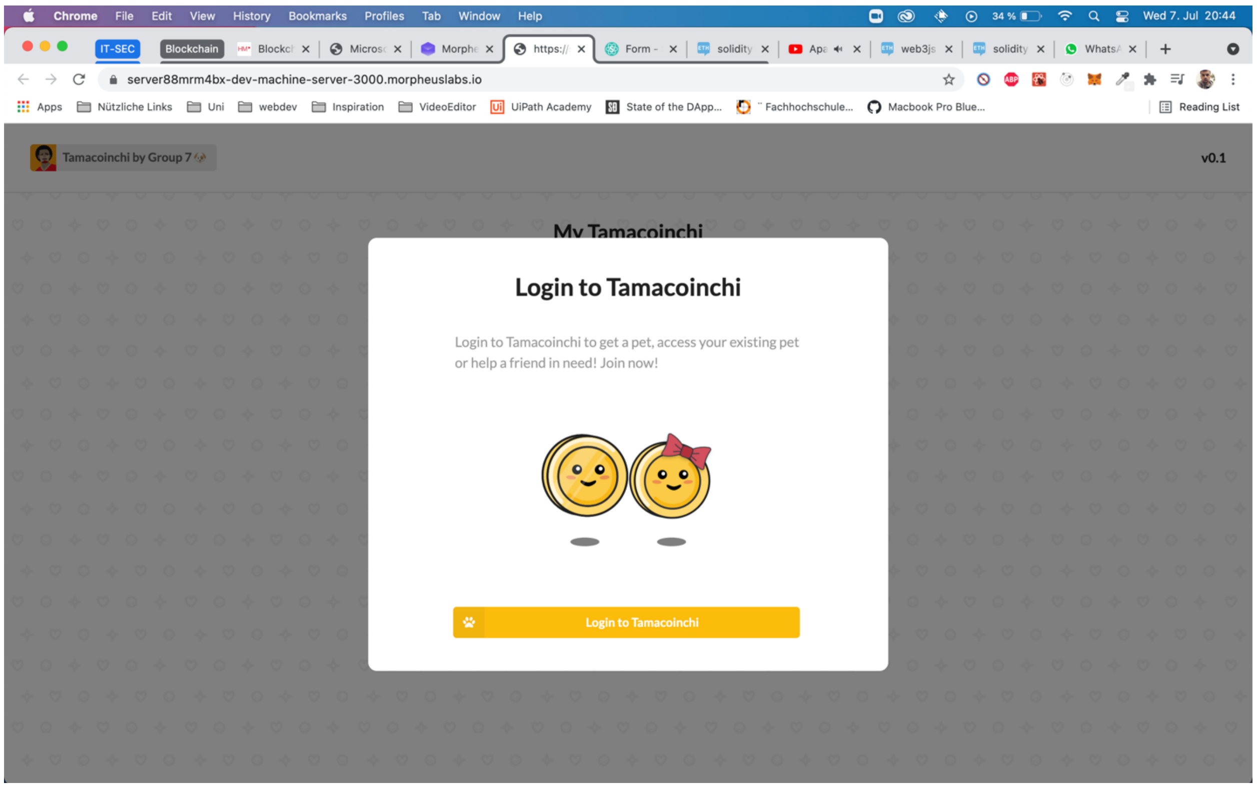
function createPet(string memory _ownerName, string memory _name, bool isMale, uint lastTimeFed) public: Creates the Tamacoinchi.

function getDeployedPets public view returns (address[] memory) public: Returns all deployed pets.

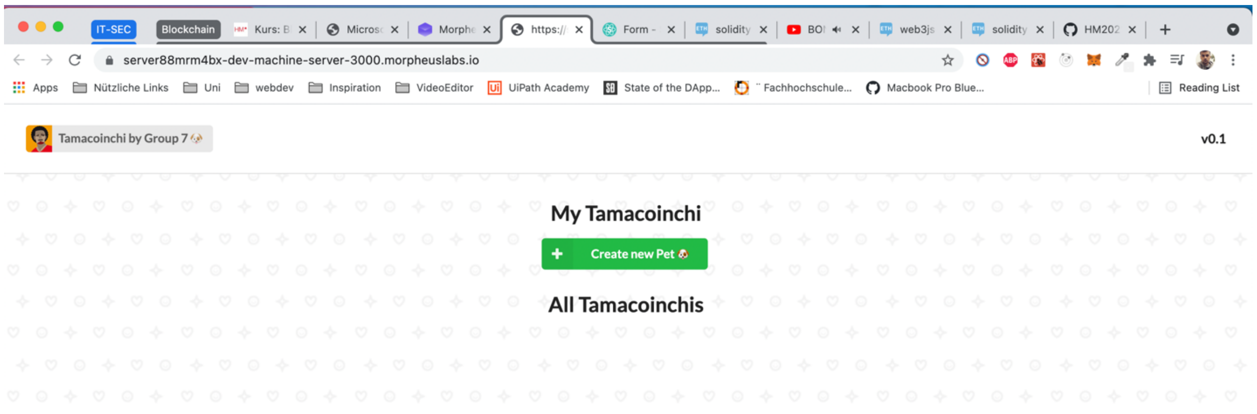
FrontEnd:

Tamacoinchi is a single-page frontend application without additional pages. Everything you want to do is possible on the frontpage.

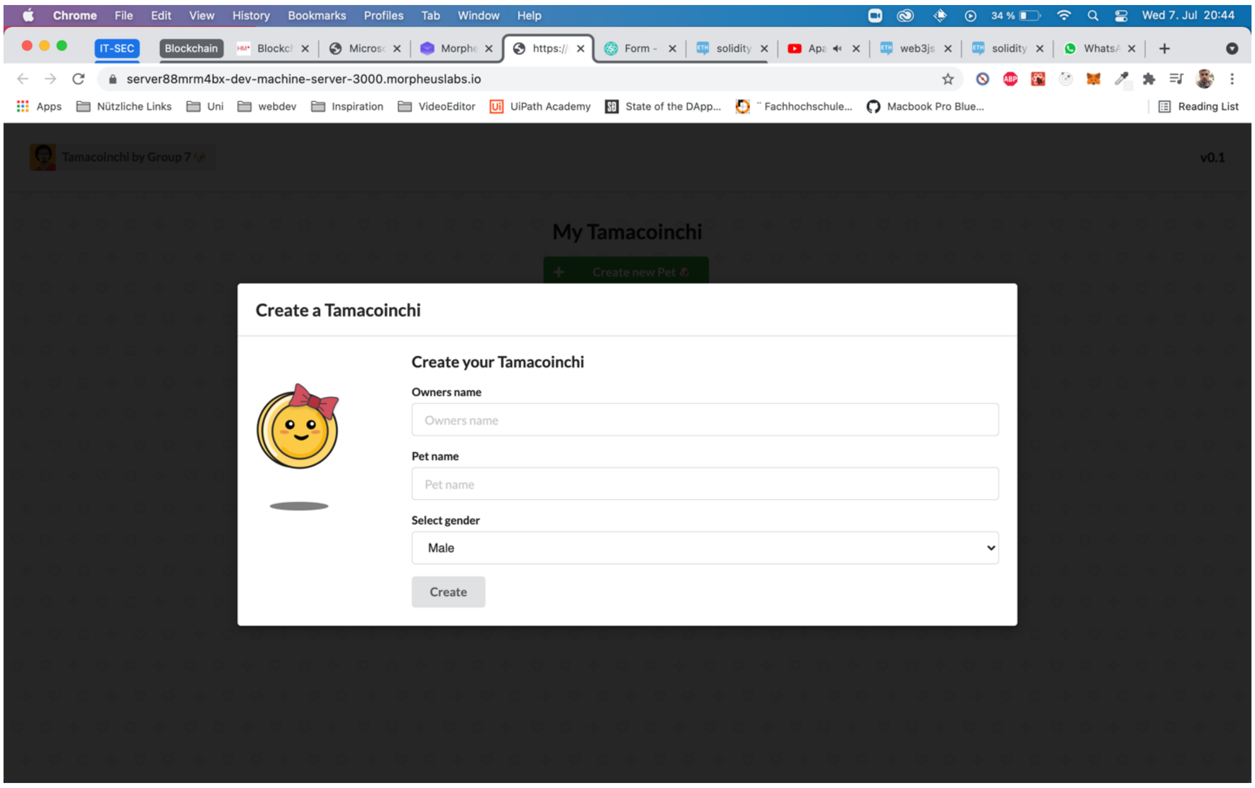
When you first open the site, it prompts you to login with your MetaMask account.


server88mrm4bx-dev-machine-server-3000.morpheuslabs.io
Tamacoinchi by Group 7
v0.1
My Tamacoinchi
Login to Tamacoinchi
Login to Tamacoinchi to get a pet, access your existing pet or help a friend in need! Join now!
Login to Tamacoinchi

After logging in, you are prompted with a window where you can create your own Tamacoinchi.



By clicking on the create-button a prompt will show up, where you are required to enter some values like name, owner-name and gender. Afterwards you can click on create and confirm with your MetaMask.



Chrome File Edit View History Bookmarks Profiles Tab Window Help
server88mrm4bx-dev-machine-server-3000.morpheuslabs.io
Tamacoinchi by Group 7
v0.1
My Tamacoinchi
Create new Pet

Create a Tamacoinchi

Create your Tamacoinchi

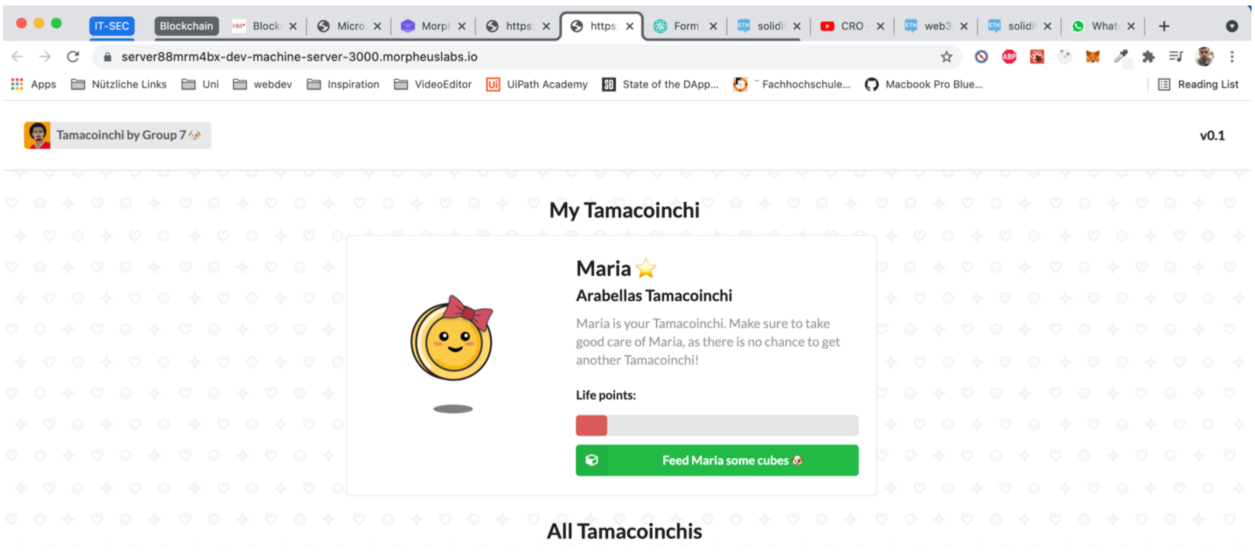
Owners name: Owners name

Pet name: Pet name

Select gender: Male

Create

A Tamacoinchi is created and added to the blockchain. It also will appear on the screen as soon as it is deployed to the blockchain.



My Tamacoinchi

Maria

Arabellas Tamacoinchi

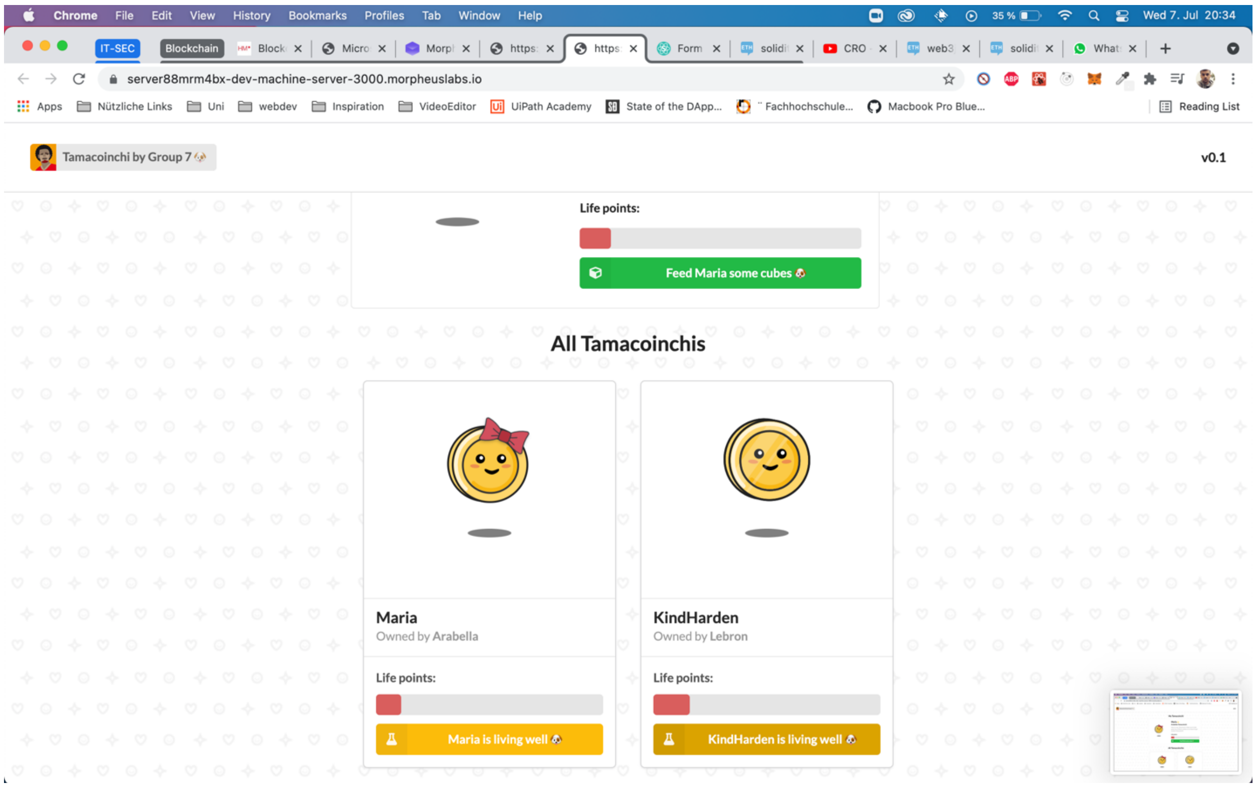
Maria is your Tamacoinchi. Make sure to take good care of Maria, as there is no chance to get another Tamacoinchi!

Life points:

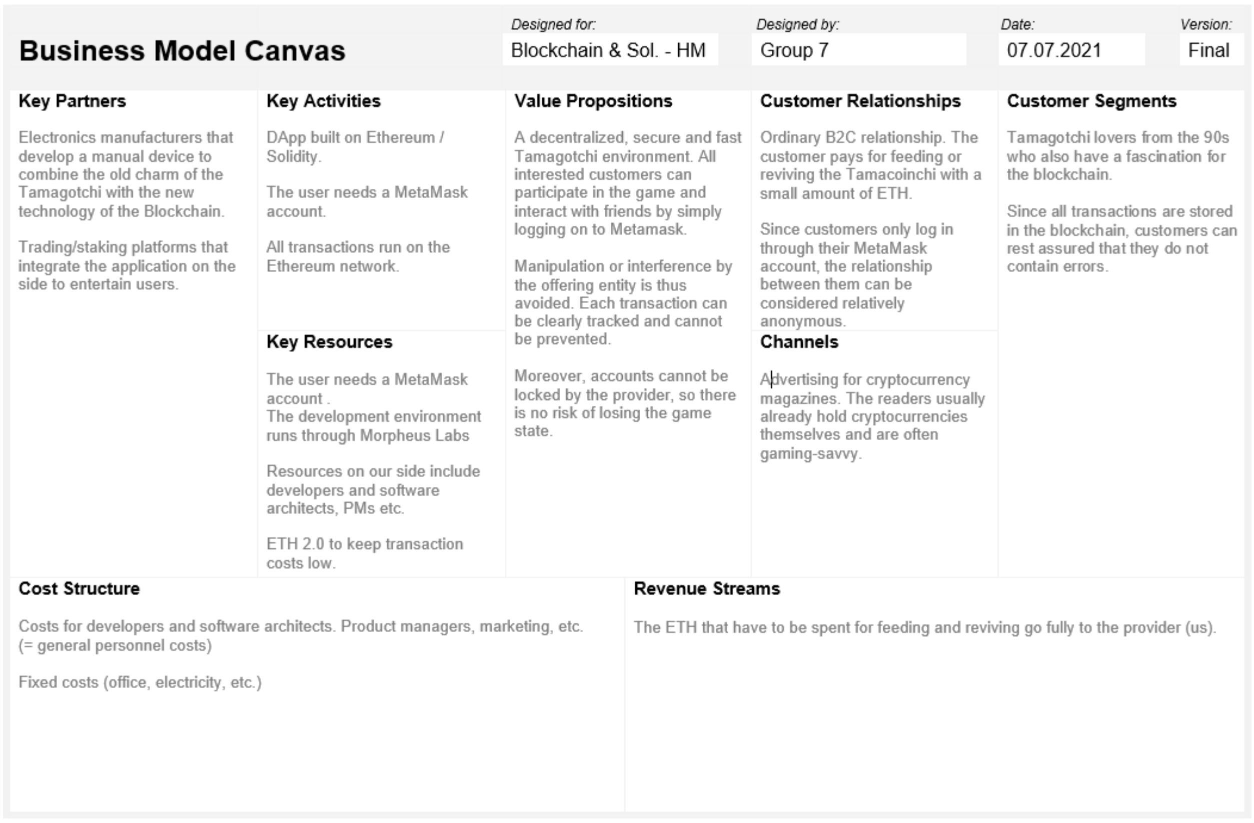
Feed Maria some cubes

All Tamacoinchis

Displayed below are pets of other users. You can either feed your own Tamacoinchi or help others out by reviving theirs, if their Tamacoinchi is dead.



OPPORTUNITY & VALUE PROPOSITION



Business Model Canvas

Designed for:	Designed by:	Date:	Version:
Blockchain & Sol. - HM	Group 7	07.07.2021	Final

Key Partners

Electronics manufacturers that develop a manual device to combine the old charm of the Tamagotchi with the new technology of the Blockchain.

Trading/staking platforms that integrate the application on the side to entertain users.

Key Activities

DApp built on Ethereum / Solidity.

The user needs a MetaMask account.

All transactions run on the Ethereum network.

Key Resources

The user needs a MetaMask account .
The development environment runs through Morpheus Labs

Resources on our side include developers and software architects, PMs etc.

ETH 2.0 to keep transaction costs low.

Value Propositions

A decentralized, secure and fast Tamagotchi environment. All interested customers can participate in the game and interact with friends by simply logging on to Metamask.

Manipulation or interference by the offering entity is thus avoided. Each transaction can be clearly tracked and cannot be prevented.

Moreover, accounts cannot be locked by the provider, so there is no risk of losing the game state.

Customer Relationships

Ordinary B2C relationship. The customer pays for feeding or reviving the Tamacoinchi with a small amount of ETH.

Since customers only log in through their MetaMask account, the relationship between them can be considered relatively anonymous.

Channels

Advertising for cryptocurrency magazines. The readers usually already hold cryptocurrencies themselves and are often gaming-savvy.

Customer Segments

Tamagotchi lovers from the 90s who also have a fascination for the blockchain.

Since all transactions are stored in the blockchain, customers can rest assured that they do not contain errors.

Cost Structure

Costs for developers and software architects. Product managers, marketing, etc. (= general personnel costs)

Fixed costs (office, electricity, etc.)

Revenue Streams

The ETH that have to be spent for feeding and reviving go fully to the provider (us).

THE TEAM

Alexander Parr is in the 6th semester of his bachelor's degree in business informatics at Munich University of Applied Sciences. He became involved with blockchain applications in August 2018, when he understood the advantages of a decentralized database and bought his first time BTC and ETH. In the project, he was responsible for the creation of the whitepaper and BPMN.

Florian Huber is in the 6th semester of the bachelor’s degree in business informatics at the university of Applied Sciences in Munich. The first contact with blockchain was in the year 2020 because of a bachelor thesis of a friend and so the chance to learn more about it was this semester great so he decided to get more information about blockchain in this seminar. In the project, he was responsible for the Business Model Canvas.

Enis Tola is in the 8th semester of the bachelor’s degree in business informatics at the university of Applied Sciences in Munich. The first contact with blockchain was in the year 2020 because of a bachelor thesis of a friend and so the chance to learn more about it was this semester great so he decided to get more information about blockchain in this seminar. In the project, he was responsible for developing the smart contracts, backend, frontend, writing tests, as well as explaining the elements of the app in the whitepaper.

ANNEX

Github: https://github.com/HM2021-BC/BC-Group7