

## 09 - Ideas

---

- Youtube (<https://drive.google.com/file/d/1RiCojSxIw2D0MUfezzvxQITH-2ImwUHg/view?usp=sharing>).
- Github - PDF (<https://github.com/FrankBevr/100-Ideas/blob/main/practice/00/00-Ideas.pdf>).
- Github - Code (<https://github.com/FrankBevr/100-Ideas/blob/main/practice/00/animalo/lib.rs>).

# Overview

---

- Intro
- Think
  - ☒ 100 Words
  - ☒ 10 Sentences
  - ☒ Rate & Select
- Sketch
  - ☒ Idea
  - ☒ Requirements
  - ☒ Stories
  - ☒ Diagrams
- Paint
  - ☒ Write
  - ☒ Build
  - ☒ Test

## 100 Words

- Note: Create Rich Pictures. That's it.

Words				
Earrings	Shirt	Chair	Microphon	Technican
Phala	Computation	Finance	Offchain	Polygon
Mandala	Nightshift	Google Meet	Starbucks	Fanta
Swiss	Berry	Communication	Thailand	TukTuk

## 10 Sentences

Theme	Word	Phrase	Rating
Marketing	Berry	Your animals of exelcene	5
Right Items	Fanta	Right items on the go	6
Computation	Blender	Mint on the fly	5
Call out	Microphon	You cards on the spot	2

## Rate & Select

The winner is `animo.xyz`



# Sketch

## Idea

- Problem: The marketing value of your animal is hard to share
- Idea: Tokenize your public persona animal
- Solution: `animalo.xyz`

## Requirements

### Functional Requirements

- The App must connect to Instagrams
- The App must show a selection of prominent animals
- The Owner must be able to tokenize the rewards

### No Functional Requirements

- The App must take care of the animal automatically

## User Stories

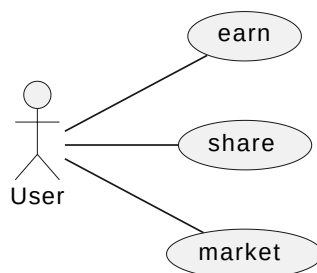
- As a User I want to earn money
- As a User I want to share my animal success
- As a User I want to I want to market my product

Currently creates a animal rewards system.

I bet my ass off that exist in the Web 2 way and uses a centralized entity.

## Diagram

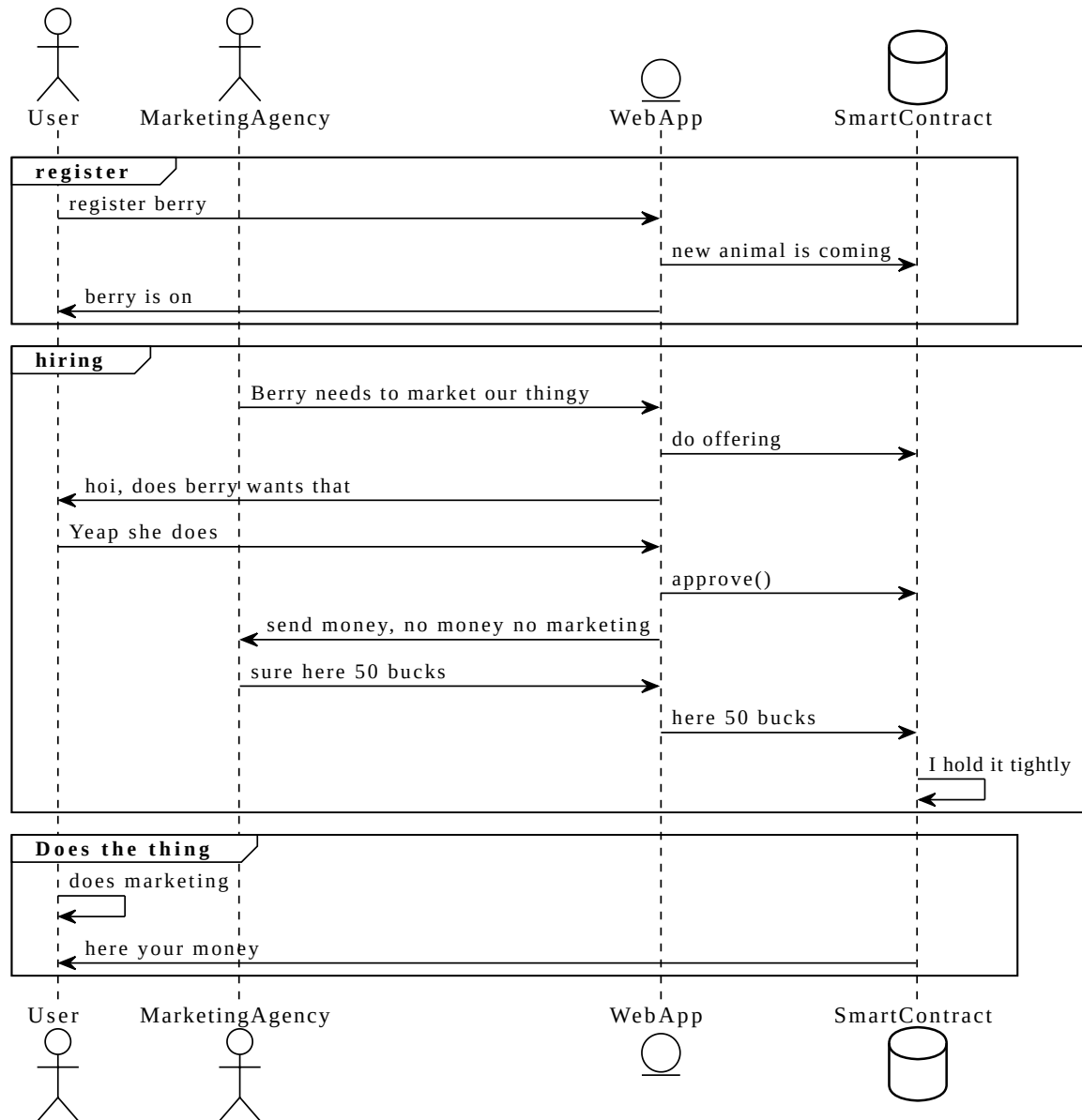
### Use Case Diagram



### Class Diagram

© Animalo
o animal: String o rewarder: AccountsId
o pay(_animal) o get_current_reward(): String

# Sequence Diagram



```
#![cfg_attr(not(feature = "std"), no_std, no_main)]

#[ink::contract]
mod animalo {
    use ink::prelude::string::String;

    #[ink(storage)]
    pub struct Animalo {
        value: bool,
        animal: String,
        rewarder: AccountId,
    }

    impl Animalo {
        #[ink(constructor)]
        pub fn new() -> Self {
            Self {
                value: true,
                animal: "Berry".into(),
                rewarder: AccountId::from([0xff; 32]),
            }
        }

        #[ink(message)]
        pub fn get_current_rewarder(&self) -> AccountId {
            self.rewarder
        }

        #[ink(message)]
        pub fn pay(&self) {}

        #[ink(message)]
        pub fn flip(&mut self) {
            self.value = !self.value;
        }

        #[ink(message)]
        pub fn get(&self) -> bool {
            self.value
        }
    }
}
```