Roadmap: TruCol

A decentralised collaboration protocol for test-driven development

September 8, 2022

1 Introduction

This roadmap describes how TruCol got started, where we currently are, which milestones take us to a significant market share, and what capital is being raised now, and down the road.

1.1 backstory

While developing our own software company besides our studies, we wanted to increase the rate of development by setting out bounties. After evaluating bounty platforms, we noticed most take a cut as middle-person, allow for ambiguity, or cater to niche markets. Since we wanted to give the developer the full reward, we assembled a student team from Radboud University in Nijmegen, and Delft University, both in the Netherlands. We combined Aerospace Engineering students, with Artificial Intelligence students and Computer Science students and competed at the ETHDenver to develop the protocol.

The TruCol protocol presents an improvement of market efficiency and developer autonomy by decentralisation and automation of test-driven development. The protocol promotes inclusive, fair and accessible work, by enabling developers to participate in the market regardless of their circumstances. Employers publish a smart contract with a bounty for deterministically verifiable development tasks which are fit for solving by external parties. Developers from all over the world are able to complete these tasks and get rewarded automatically when the requirement of the smart contract is fulfilled. The protocol thus removes the middleman and costly fees, and stimulates an open and fair development market.

This work was awarded multiple prizes, amongst others, a prize of $3000,- for contributing to UN sustainability goal nr 8:"Providing fair and equal work to all". We continued the development and generated a POC in Solidity. This POC is presented during the Ethereum Conference 4 in Paris in the summer of 2021.

2 Now

We are at the end of our studies, have had amazing experiences and feedback on our protocol, and would like to move from a POC to a practically usable implementation.

3 Milestones

Six major milestones are identified within the TruCol project. To get to an operational break-even position, the first two are relevant. The latter two are relevant for future seeding rounds and exponential growth.

3.1 Operational Break Even

1. *Complete CI deployment* - To improve development collaboration, our self-hosted GitLab-CI should run in a Docker container or Virtual Machine, instead of on our own devices. This is to prevent interference, and to help filter code contributions that do not adhere to a minimum quality. Given the adversarial nature of this protocol, that code quality is essential.

2. *Support all languages* - To allow any company to use the protocol, we should extend the protocol from a Solidity-Solidity implementation only, to oracles that query the (decentralised) CI build status of the bounty hunter solutions.

3. *First Customer Usage* - We have a first customer, Viggo Service Enablers, an airport logistics company, that is eager to use the protocol once available.

3.2 Growth

1. *Wide-spread Adoption* - Over 100.000 bounties must have been deployed, with a net value of at least $30.000.000 in bounties being allocated. This provides us with a minimal amount of data and revenue potential to start working on an in-house arbitrage AI engine.

2. *Exit/Next Seeding Round* - To build an arbitrage AI, we need to raise funds. It is expected this requires North of 100 million, given the complexity of the task.

3. *Self-sustainable AI* - If we succeed to build a self-sustaining arbitrage AI, we can improve software development efficiencies worldwide, the software development landscape has changed into requirement specifications.

The remainder of this roadmap focusses on the roadmap to operational break even.

4 Planning

This section visually presents the planning in the form of a Gantt chart in fig. 1. The description of this Gantt chart is included in section 4.1.

4.1 Gantt Chart Description

The description of the activities in fig. 1 can be given as:

4.2 Decentralised Technology Development

- **Develop protocol** - The programming work and documentation work that is required to render the TruCol protocol to a mature and robust state.
- **On-chain: Solidity+VRF** - Finalisation of the Solidity to Solidity implementation of the TruCol protocol implementation that leverages Chainlinks Verifable Random function (VRF).
- **Git integration: Tellor** - Providing a lower-cost option to the users whilst allowing the user to apply the TruCol protocol in practically any programming language using Tellor oracles that query Git repository content and (build) status.
- **Git integration: Chainlink** - Same as the Tellor option, except using Chainlinks oracles.
- **Alternative Chains** - Implementing the TruCol protocol in alternative chains to facilitate easy use for the users whilst possibly lowering costs and/or modulating the desired levels of decentralisation
- **(Decentralised) Continuous Integration** - Realizing a mature implementation in which the oracles can verify the build status (whether the tests in the smart contract actually passed or not) in a robust fashion. Ideally implementing support for decentralised CIs.
- **Security & Robustness** - A security audit of the TruCol protocol implementations.

4.3 Platform Development

- **Platform & Ecosystem** - Development of the online platform that provides a convenient place for users to use-, discuss- and learn about the TruCol protocol and its various implementations.
- **Website** - Completion of the company website.
- **API** - Application programming interface that allows users to submit contracts using the command line interface (CLI).
- **GUI** - Graphical user interface, that makes it easy and intuitive for new users to start using the TruCol protocol for their applications.
- **Forum** - Environment a-la stack-overflow that cultivates a knowledge base around the use of the TruCol protocol.
- **Marketing platform** - Development of the approach to realise wide-spread adoption of the TruCol ecosystem.
- **Subsidize bounties** - Subsidisation of bounties to attract new users to the platform.
- **Platform Planning Buffer** - A buffer accounting for unknown unknowns/unexpected delays.

4.4 Business Development

- **Launch company** - The administrative and non-technical aspects of growing the TruCol company.
- **Qualitative partner research** - An analysis to identify relevant partners in the growth of our company.
- **Establish Organisation** - organisational aspects of growing the company, with "Auditing, Hiring, Administration, Legal & Financial tasks as its respective subset".
- **Marketing** - Development of the approach to realise wide-spread adoption of the TruCol protocol whilst realising a steady stream of new customers.
- **Organisation Planning Buffer** - A buffer accounting for unknown unknowns/unexpected delays.

4.5 Gantt

Figure 1 contains the Gantt chart that is generated to plan the development of the TruCol company. One can observe that several of the development-activities can be performed in parallel, these are accordingly stacked vertically. Dependencies of outputs of activities imply a "stairway" pattern in the Gantt chart.

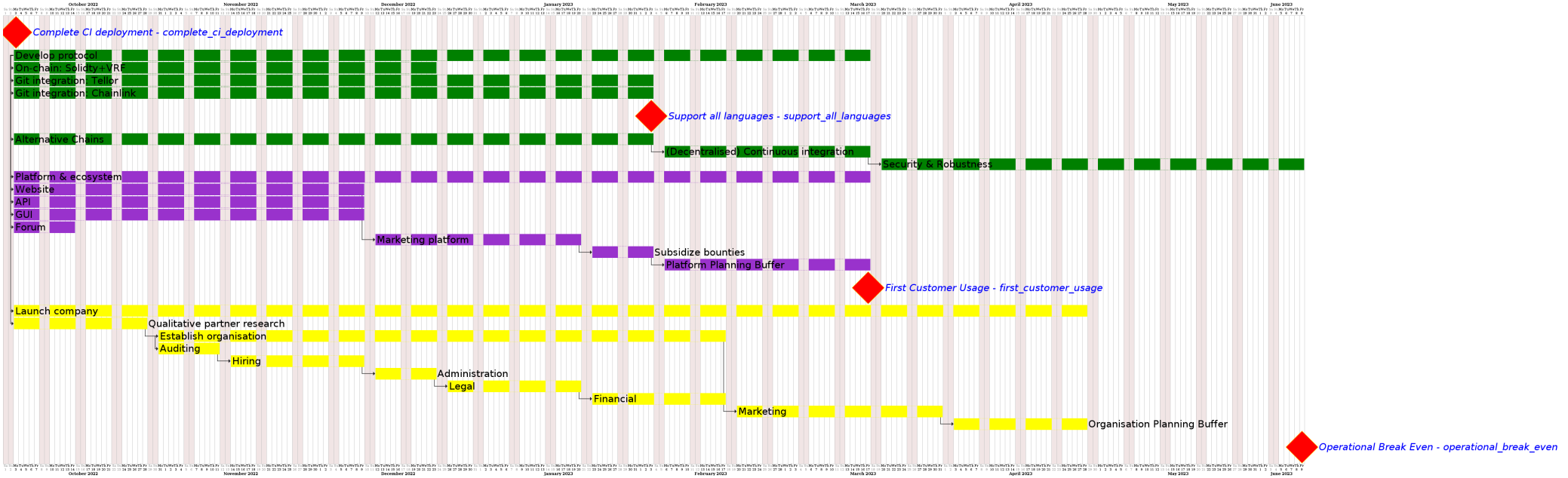

Figure 1: Gantt chart that is generated to plan the development of the TruCol company. (Source code in appendices, and on github.com/trucol/Roadmap).

5 Cost Model

This section describes the expected costs to bring the TruCol company to an operational break-even. These costs are described in the form of a cost model, that is described in section 5.1 and the model parameters are given in section 5.2. The mathematical model specification and assumptions are located in the appendices at section A and section B respectively.

5.1 Description

The total costs are computed based on the following 4 factors:

- The cumulative amount of human labour hours that are planned to be executed, multiplied with their respective hourly wage costs.
- Bounty subsidisation to attract new users.
- Buffer costs to generate wide-spread adoption of the TruCol protocol.
- Daily operational costs, e.g. electricity, company phone usage, travel etc.

For the mathematical specification of this model, see section A. For the Python code of this model, one can check: `https://github.com/TruCol/Roadmap/tree/main/src`.

5.2 Model Parameters

Table 1: Cost Model Parameters in €(/hr or absolute, unlessspecified otherwise)

Parameter	Value
blockchain dev	76
front end dev	41
human resources	36
bounty subsidising	100000
buffer	100000
daily operational costs	200
days to operational break-even	270
labour costs	479600
total cost	733600
non labour costs	254000

For a motivation for these values, see section B

6 Total Cost Estimate

After multiplying the amount of labour hours with their respective hourly labour costs, a total estimated labour costs of €$733600,-$ is generated.

Table 2: Total Expected Investment Costs

Description	Cost [€]
Develop protocol	255360
Platform & ecosystem	114800
Launch company	109440
Bounty Subsidising	100000
Buffer	100000
Operational Costs	54000

7 Conclusion

This document presents the planning to develop the healthy TruCol company within the timeframe of roughly 9 months, and documents the assumptions and methods used to generate this planning. The main tasks in this planning are composed of decentralised technology development, TruCol platform development and business

development. A total labour cost of roughly €479600 is estimated, and another €254000 is estimated for bounty subsidisation and as a buffer.

At the end of this planning, the TruCol company is expected to sustainably operate, generating ROI. Our aim is to gradually increase the cultivation of the diversification potential of the TruCol protocol at this point, whilst starting to develop a strategy to develop our own in-house automation/AI-engine based on the dataset that we continuously grow.

A Cost Model Specification

The costs untill TruCol is expected to be operationally break-even (OBE) are estimated using eq. (1):

$$costs_{OBE} = labour + bounty\,subsidisation + buffer \tag{1}$$

The 3 right hand terms are specified in section A.1 to section A.3.

A.1 Labour Cost Specification

The labour costs costs are mathematically described in Equation (2) and computed automatically in the Gantt chart creation using Python. They are specified as the sum of the labour costs of all workers i. The labour costs of a worker i is defined as the amount of hours worked by worker i, multiplied with the labour costs of the worker i:

$$labour = \sum_{day=1}^{day=OBE} \sum_{worker=1}^{worker=n(day)} hrs(worker, day) \cdot wage(worker) \tag{2}$$

With:

- $labour[€]$: The total expected labour cost to reach operational break-even (OBE).
- $worker[-]$: The index representing a TruCol employee.
- $n(day)[workers]$: The number of workers at TruCol on day day.
- $OBE[day]$: The index of the day on which TruCol reaches operational break-even (OBE).
- $hrs(worker, day)[hours]$: The number of hours that TruCol employee $worker$ has worked on day day.
- $wage(worker)[€]$: The hourly labour costs of a specific TruCol employee $worker$. For simplicity, the hourly labour costs of a TruCol employee are taken as the average hourly cost of that worker, over a time of 1 year.

A.2 Bounty Subsidisation Cost Specification

The bounty subsitisation costs up to OBE, are estimated as:

$$bounty\,subsidisation = €100.000 \tag{3}$$

This cost is estimated with the aim of subsidising 2 to 20 companies. This range has 1 order of magnitude as range to accomodate different strategies. Either one can focus on 1 or 2 leading companies and motivate them to try the TruCol protocol. After these companies have used the protocol, work can be performed to ensure the rest of the market follows, based on the competitive advantages experiences by these 2 leading companies. Otherwise, multiple smaller companies may be motivated to use the TruCol protocol to generate a larger degree of interaction and engagement with the protocol.

A.3 Buffer Cost Specification

The buffer costs up to OBE, are estimated as:

$$buffer = €100.000 \tag{4}$$

This is to overcome known unknowns and possibly unknown unknown occurrances.

B Assumptions

B.1 Decentralisation Developer Wages

The hourly wage of the developers working on decentralised technology is based on a mixture of ± 3 junior developers working at €100.000,- per year, and 2 senior developers working at €200.000,- per year. This yields an average developer cost of

$$\frac{3 \cdot 50 + 2 \cdot 100}{5} = \frac{350}{5} = €70,- \tag{5}$$

The datapoints used to come to this estimate are the promoted starting wages for Junior Developers/Engineers at Optiver in Amsterdam, Think-cell in Berlin, and a third Zurich company, which all ranged between 80 to 120k at the time of inspection (Around March 2021). No proper datapoint is used to estimate the salary of the senior developers. Previous experience in co-working with senior developers led to an estimate that their hourly contributions are at least twice as valuable as that of a junior developer. Another indicator for the doubling in wage between junior and senior dev may be the hear-say high demand in solidity/decentralisation developers.

The 70,- hourly wage is interpreted as $€76,-$ per hour to be on the conservative side of estimates.

B.2 Website/Platform Wages

The website+API+GUI development is estimated at €41 per hour. This estimate is based on a reduced hourly wage of the junior decentralised technology developers (from €50,− to €40,−). Some of the development costs for these activities may be performed at a lower hourly cost price. However, the platform development work also contains UX design. And excellent UX design is quite costly, hence the average hourly wage for this estimate is kept at €41,−.

B.3 Business wages

The hourly wages for the business development side of our company is estimated at €36,− per hour. This estimate is based on a reduced hourly wage of the junior platform developers (from €40,− to €35,−).

B.4 Activity durations

The estimates for the durations of the activities for both decentralised technology development as well as ecosystem development are extrapolations of our experience in developing in these disciplines. The business development activities durations are based roughly on estimating what those activities entail and how long it would take to complete them.