Predictsol: A Prediction Tokenizer for the Solana Network

Lukas Saul[1,2], Robert Gludo[2], Jonathan Gludo[2]

1) Hefei University of Technology 合肥工业大学
2)Vermont Secure Computing Consultancy

March 2026

Abstract:

Predictsol demonstrates an architecture for a new generation of decentralized prediction markets which are transparent and modular. For the first time we separate the tokenization and associated redemption tasks of a prediction market into their own smart contract: the prediction tokenizer. A tokenizer contract offers the ability to create "events" such that outcomes are represented by tokens. These tokens are minted by participants in exchange for deposit, and the contract allows any users to redeem these tokens as necessary before and after an event is finalized. This allows the trading and transfer of the event's outcome tokens to occur transparently and without conflict of interest on other platforms, while issuance and redemption remain public, auditable, and permissionless allowing a new level of security and transparency. In this document we describe the motivation and architecture of the Predictsol smart contract.

Introduction:

Prediction markets allow participants to purchase shares which represent the outcomes of an event. Since at least as long ago as Hayek in 1945, many authors have discussed the utility of such a market for accurately predicting events and effectively crowd sourcing intelligence via financial incentives, for a review see [Ivens et al., 2014]. Modern prediction markets have been more deeply analyzed as predictive tools [Cordoba et al., 2025]. With the incentive of a positive return on investment, those who believe themselves to have the knowledge to better predict an event than others can purchase such a share at a discount. When the outcome matches the share or contract, the purchaser can then redeem the share making a profit.

The prediction marketplace is both a place where people can act on their analysis and prediction skills, and also a place to announce their opinion, as the market in total indicates the presence of belief in any such knowledge. In short, these markets are methods of communication and monetization of information, and participation might be likened to polling or voting. However, as with many other financial products, prediction markets can also be seen as wagers or games, and as such should be treated with caution as they allow participants to incur losses, both from improper use and from the markets themselves losing integrity.

Traditional prediction markets are operated as centralized houses, in which one party performs the job of the oracle (determining the final outcome of an event after the fact), the market (holding the assets and matching buyers and sellers), as well as managing the books. While this can work, it also presents attack surfaces and potential weaknesses. Centralized control of funds always creates a risk of theft or loss of the pot, while centralized oracles present a target for blackmail or oracle attack. Further, a centralized system also presents security risks for those involved who possess the power to influence the payouts or decisions, as they could be targeted with the goal of influencing those decisions. The beneficial tasks of a prediction market require our confidence in the fairness and openness of the market itself, which can be difficult to demonstrate for a centralized business.

Fortunately, public blockchain networks have enabled a new level of transparency in which some aspects of the price discovery and trading volume are verifiable. For example, in [Wang, 2020] an outline of an architecture for a decentralized prediction market is described, including the motivations. The technology has evolved quickly and [Rahmen et al., 2025] provide other architectures and a review.

For these reasons we pursue development towards a modular and decentralized prediction marketplace, in which a smart contract is the transparent administrator of all funds, enabling public audit and verifiable operation. In particular, we take the tasks of market matching, market making, price discovery and associated logistics and separate them completely to simplify our contract as much as possible, only controlling the creation and redemption of relevant tokens.

Prediction Tokenizer:

As the first prediction tokenizer, the smart contract is simplified in its direction towards one specific task: to create tokens for an event, and allow them to be minted and redeemed. The functions which the contract offers are as follows:

1) Create Event

When a participant creates an event, they pay for the initialization of the data structure and oracle and also prepare to receive 1/3 of a percent of all incoming funds to that event on the contract. The event itself consists of a natural language string which describes the predicted outcome, in the form of a statement which will be either true or false after finalization. A date and time are chosen at which point the contract is closed for all token issuance and redemption. Two other dates must be chosen which are relevant to the oracle: the first is the *commit* deadline, by which oracle participants must commit their decision, and the second is the *reveal* deadline for oracle participants to reveal their decision. These stages are necessary because the votes of oracle participants cannot be public until after all votes are in (otherwise copycat voting would be a profitable automation).

2) Mint tokens for an existing open event

Tokens are minted for both outcomes of an event (true and false) together, to ensure that there is exactly the right amount of collateral held in the contract to pay for redemption regardless of the outcome. Liquidity providers and participants who mint tokens can then provide them to decentralized exchanges as LP or sell them directly. Once an event has been created, the token IDs are fixed and the minting is always in proportion to the deposited asset (in this case, solana). For example, any wallet can interact with any event on the contract depositing N solana and receiving N “true” tokens for the event and N “false” tokens for the event (N can be less than one down to a single lamport). These tokens can now be sold or staked at decentralized marketplaces such as Raydium or Orca or Uniswap.

3) Redeem tokens (before event is finalized)

Before an event is finalized, event tokens can only be redeemed together; that is, any user with both N “true” and N “false” tokens for the event can redeem these for N solana from the contract vault. This means that if the markets ever show the sum of the prices of the tokens to be less than 1, that there is an arbitrage opportunity available. It can be seen that this ensures there will always be exactly the deposit required to pay for future redemption, while allowing maximum flexibility to participants.

4) Finalize Event

An event is finalized first by stopping all redemption and minting at a given closing date. Then the oracle begins to collect responses from its participants. We use the Truth.IT decentralized binary oracle. The oracle effectively returns one of three results, True, False, or Undetermined. The resulting state explicitly determines who can redeem which tokens. For the Predictsol contract, we demand an 80% agreement among oracle participants to resolve an event, otherwise the event is considered unresolved.

5) Redeem tokens (after event is finalized)

Once the result is determined, anyone with the relevant tokens in their wallet can redeem the winning token for full value of solana. If the user has 0.032 of the winning token, the user will receive 0.032 in solana. If the event was undetermined, then each token is worth half the face value for redemption.

6) Cleanup event funcitons and fees

There are four kinds of fees which affect users of the predictsol contract. The first is that every transaction on the solana network has some very small transaction fee associated with it. The second is that the event creator gets 1/3 of a percent of every incoming solana to the mint. A further fee is an additional 1/3 of a percent which goes to the Truth.IT oracle network to provide incentive for those participants, and 1/3 of a percent which goes to the developer fund of the PredictSol contract software maintainers. Every participant has one month after the end of an event to redeem their tokens, after which they will be swept to the contract developer fund.

Conclusion

Predictsol enables for the first time an entirely transparent prediction market, which when coupled with an effective external contract decentralized oracle and an external contract decentralized exchange enables a new level of precision and verifiability in prediction markets. This eliminated some problems with existing prediction markets including some which were already at least partially decentralized [Rohanifar et al, 2025]. Users of PredictSol are recommended to run the front end UI themselves, which enables them to interact with the solana contract on their own machines without relying on any external server or domain name resolution, but for now they can also use the same front end set up on the domain provided by the developer fund,.

Please invest, predict, and transact responsibly!

References:

Rohanifar, Y., & Ahmed, S. I. (2025). Prediction Laundering: The Illusion of Neutrality, Transparency, and Governance in Polymarket . *Arxiv*.

Cordoba Otalora, F., & Themistocleous, M. (2025). Beyond the Polls: Quantifying Early Signals in Decentralized Prediction Markets with Cross-Correlation and Dynamic Time Warping. *Future Internet*, *17*(11). https://doi.org/10.3390/fi17110487

Ivens, C. F., Ohneberg, B. S., & Brem, M. (2014). Prediction Markets: A literature review 2014. The Journal of Prediction Markets. In *The Journal of Prediction Markets* (Vol. 8, Issue 2). http://ubplj.org/index.php/jpm/article/view/889

Rahman, N., Al-Chami, J., & Clark, J. (2026). *SoK: Market Microstructure for Decentralized Prediction Markets (DePMs)*. http://arxiv.org/abs/2510.15612

Wang, Z. (2020). A decentralized prediction market platform based on blockchain and masternode technologies. *China Communications*, (9), 25–33.