

Here are some bugs / tweaks When trying to test, I kept running into an issue where I would have to add a twitter link (but kept getting an error as I was using x.com/profile, but Turbo wanted twitter.com/profile

- I think what we can do is make the social links optional (people can always add them in later). The only required input would be,

It just needs to start with https:// — the rest of the URL (like x.com or twitter.com) doesn't matter, as there's no strict domain validation. That said, I'll double-check it in case you encountered the issue in an earlier version. Also, you can leave the social links empty — they're optional fields.

Token name Symbol Image (would be great if users could just select an image from their computer / phone, instead of getting the image link)

It's not quite that simple, because the image used in token metadata needs to be publicly hosted on a global service in order to be recognized by DEXes and wallets. These platforms don't store or upload images themselves — they only accept public HTTPS image URLs, ideally immutable (like IPFS links).

This kind of image hosting requires a separate backend service, which is outside the scope of our MVP. Maintaining such a service would also introduce ongoing infrastructure and storage costs, so it doesn't make sense to implement it at this stage — especially since there are free alternatives available, like imgbb.com

Pro Mode:

- Custom token amounts: I think we have a few options for people to select from plus a custom amount where they can type in the amount of total supply if they want (like the image listed),

Ok, I'll update the total supply input to include a few predefined options (e.g. 1M, 10M, 100M, 1B), with 1B as the default, and also allow users to enter a custom amount manually if they prefer (which, in fact, is possible even now). We'll get that change implemented.

For tokens on the main home screen if there is a way to show a progress bar (like how far the token is away from graduating to the dex).

It's actually already displayed — you can see the current raised amount vs. the graduation threshold in the screenshots (e.g. "0.000137 / 12 ETH").

That said, I'll add a visual progress bar or gauge to make the progress toward graduation more clear and intuitive at a glance.

Token creation time and then the market cap in dollars

Got it — I'll implement it.

Displaying the token creation time is straightforward.

As for the market cap in USD, that wasn't part of the original MVP scope, but I can add it as well.

When trying to buy tokens on Turbo its currently set to have you input how many tokens you want to buy, but I think it would be a bit easier if we had it set to the user just inputing the amount of eth they want to spend on buying the token

I understand the idea and agree it could make buying feel more intuitive in some cases.

That said, the current approach (asking users to input how many tokens they want to buy) is the standard in most token sale platforms and bonding curve implementations. This is because the price increases with every token sold, so it's straightforward to calculate the ETH cost for a given token amount — but not the other way around.

Allowing users to input ETH instead (e.g. "I want to spend 0.1 ETH") would require inverting the bonding curve formula, which is much more complex and would require modifying the smart contract to support that calculation. It's not something supported in the current contract and would require a full contract upgrade and retesting.

We can definitely explore that as a future enhancement, but it's outside the scope of the current MVP.

On the bonding curve, I was trying to buy and sell tokens, but there was no way for me to sell tokens on the bonding curve. So I think we need a buy/sell box similar to the listed image, as I could not figure out how to sell the token after I bought it on the bonding curve

Thanks for the note. In the agreed MVP scope, selling tokens back to the bonding curve was not included, and most importantly, the entire flow was intentionally designed without it.

The idea was:

- Users buy tokens via the bonding curve (with a 1% fee)
- The creator benefits as ETH accumulates from those buys
- Once the token graduates, users can sell or trade it, **but on a DEX (GTE or Bronto)**
- Only the creator can withdraw the ETH raised (after graduation)

Supporting on-curve selling would require adding a full sell() function to the smart contract, which we intentionally excluded to keep the MVP simple and low-risk. That said, it's definitely something we can explore as a post-MVP enhancement.

I was also looking for a token after it graduated and could not find it on GTE to trade (<https://testnet.gte.xyz/>)

*Just to clarify — in the MVP, choosing GTE or Bronto as the target DEX is **for display only**, not for automatic deployment.*

As you originally mentioned:

“Select the DEX they want the token to graduate to... used for display only in the MVP — we won’t deploy live liquidity pools on external DEXes yet.”

So after graduation, the token doesn’t automatically show up on GTE. Full DEX deployment would require adding on-chain liquidity and integrating with the DEX’s contracts — which we left out of the MVP to keep it lean and simple.

*That said, I **did build in some support already**:*

- You get a ready-made **DEX metadata JSON**
- There’s a field to **add the DEX listing URL**
- And a **“Mark as Deployed to DEX”** button that updates the token status in the UI

So while it’s not fully automated yet, the foundation is there — and of course we’ll add proper DEX deployment in the production version. It’s on the roadmap for next steps.