

# School of Solana

## LECTURE 5

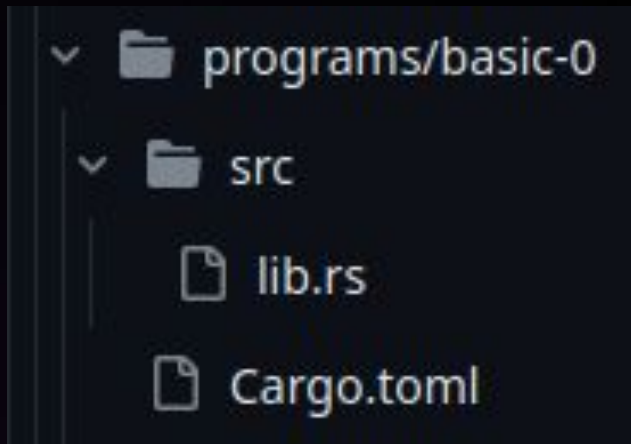
Best development practices and debugging



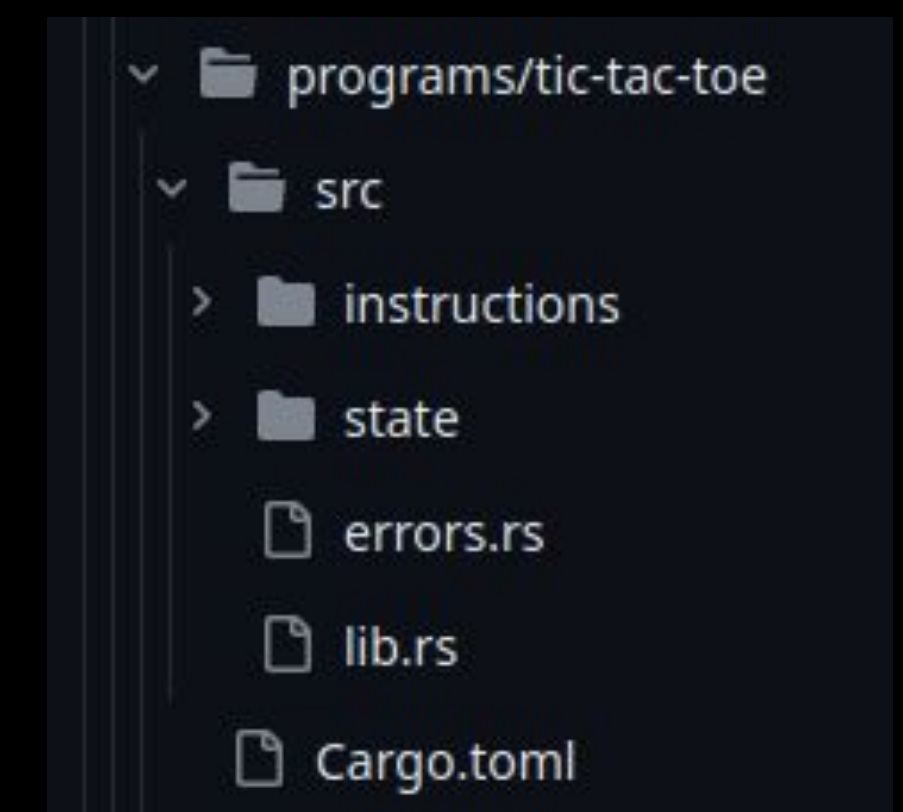
# Modularize Your Code

... and avoid spaghetti code in one file.

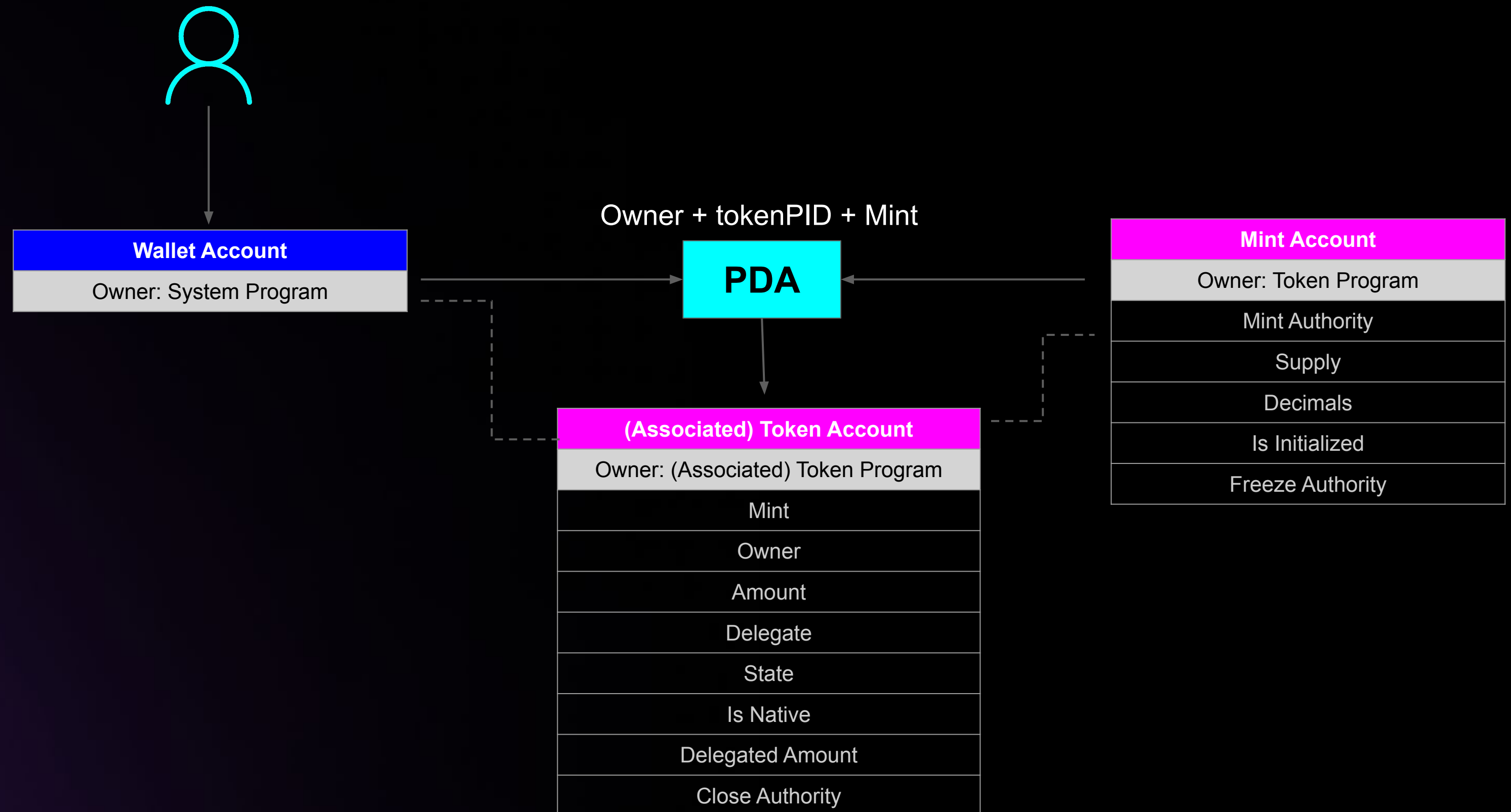
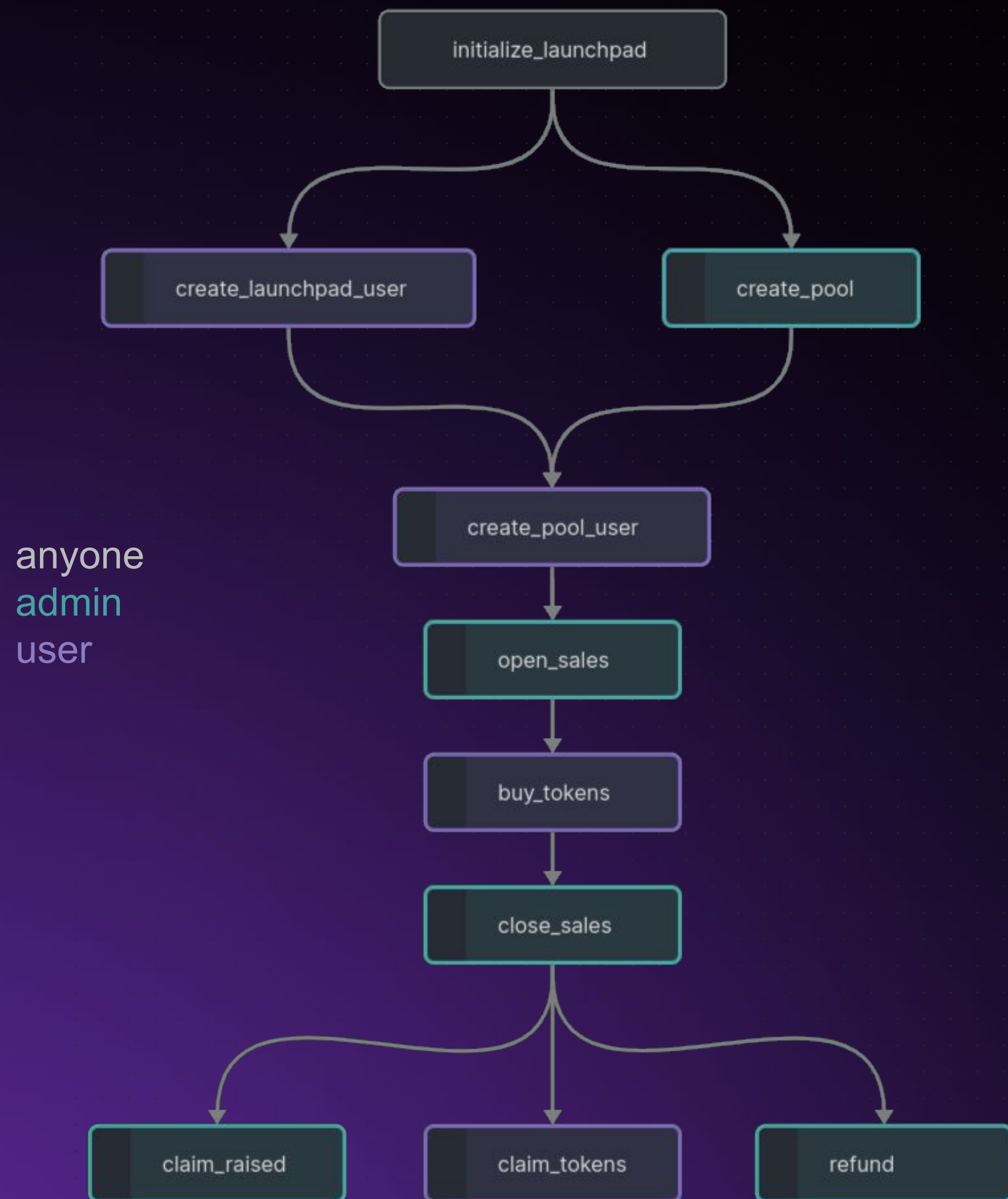
20 instructions in one file with 1200 LoC



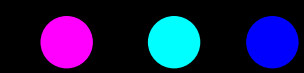
VS



# Visualize Your Program Flow and Data



# Setup GitHub Workflows & Peer Reviews



```
jobs:
  run-anchor-test:
    runs-on: ubuntu-latest
    steps:
      - uses: actions/checkout@v3
      - uses: ../github/actions/setup-solana/
        with:
          solana-cli-version: ${vars.SOLANA_CLI_VERSION}
      - uses: ../github/actions/setup-anchor/
        with:
          anchor-version: ${vars.ANCHOR_CLI_VERSION}
      - name: Run tests
        run: anchor test
```



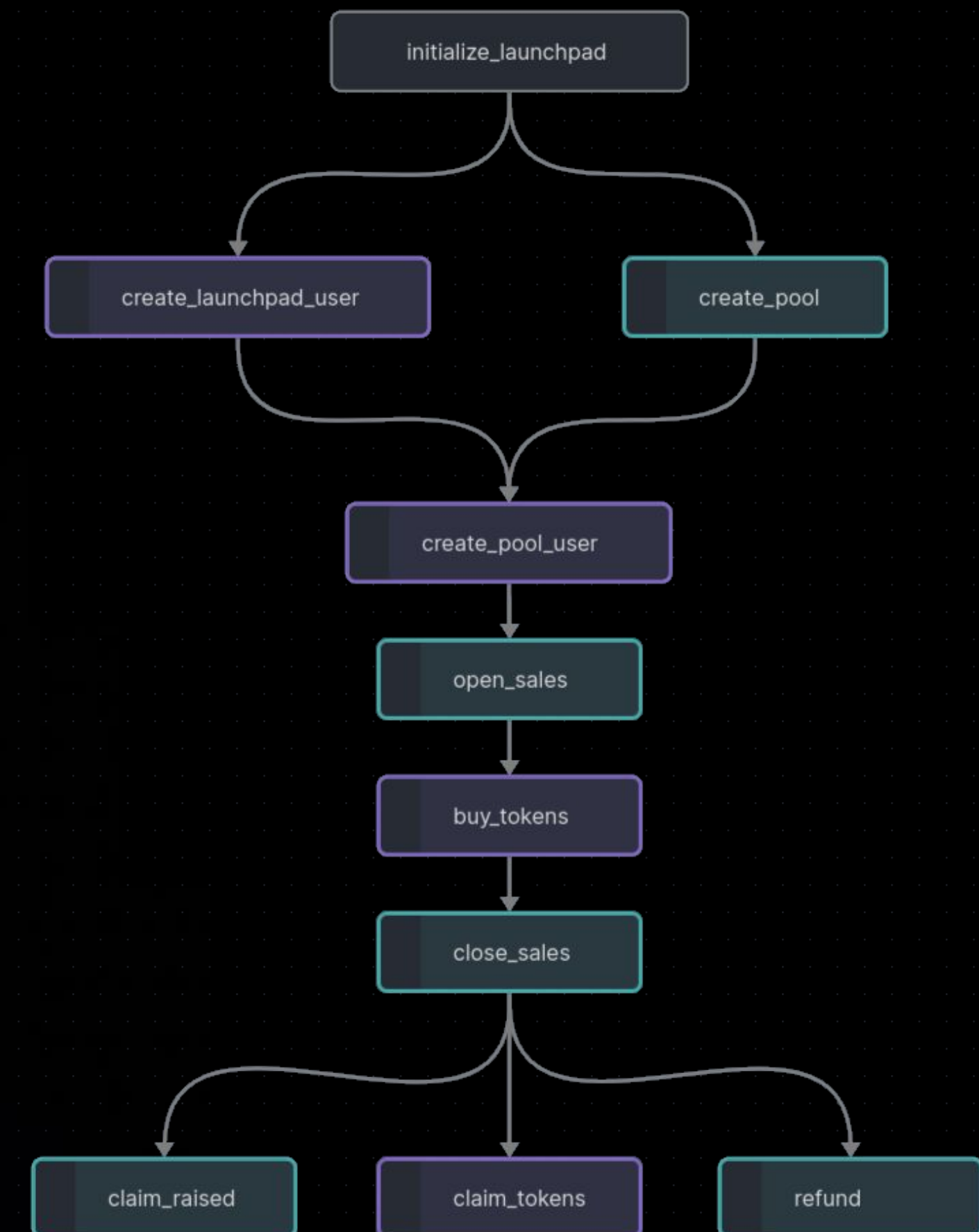
# Do Not Rely on Back-End

*“We do not check <SOMETHING> in our Solana program because it is checked by our back-end.”*

 **you have no control over the transactions that will be sent to your program** 

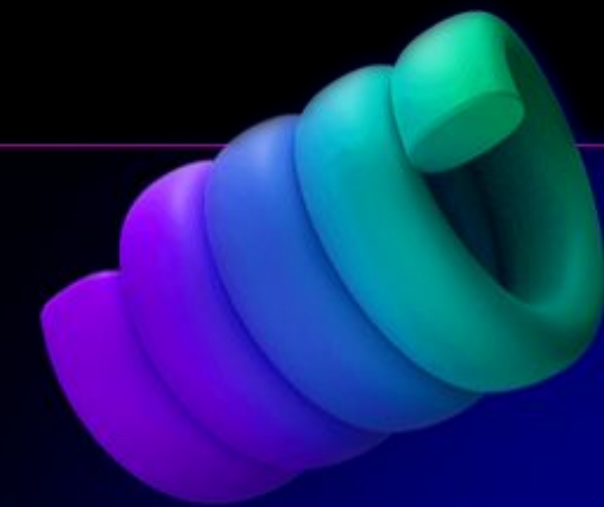
# Test! Test! Test!

- Aim to high test coverage.
- Do not test only happy path scenarios.
- Think about unexpected program flows.
- Shuffle instruction accounts.



# Testing Options

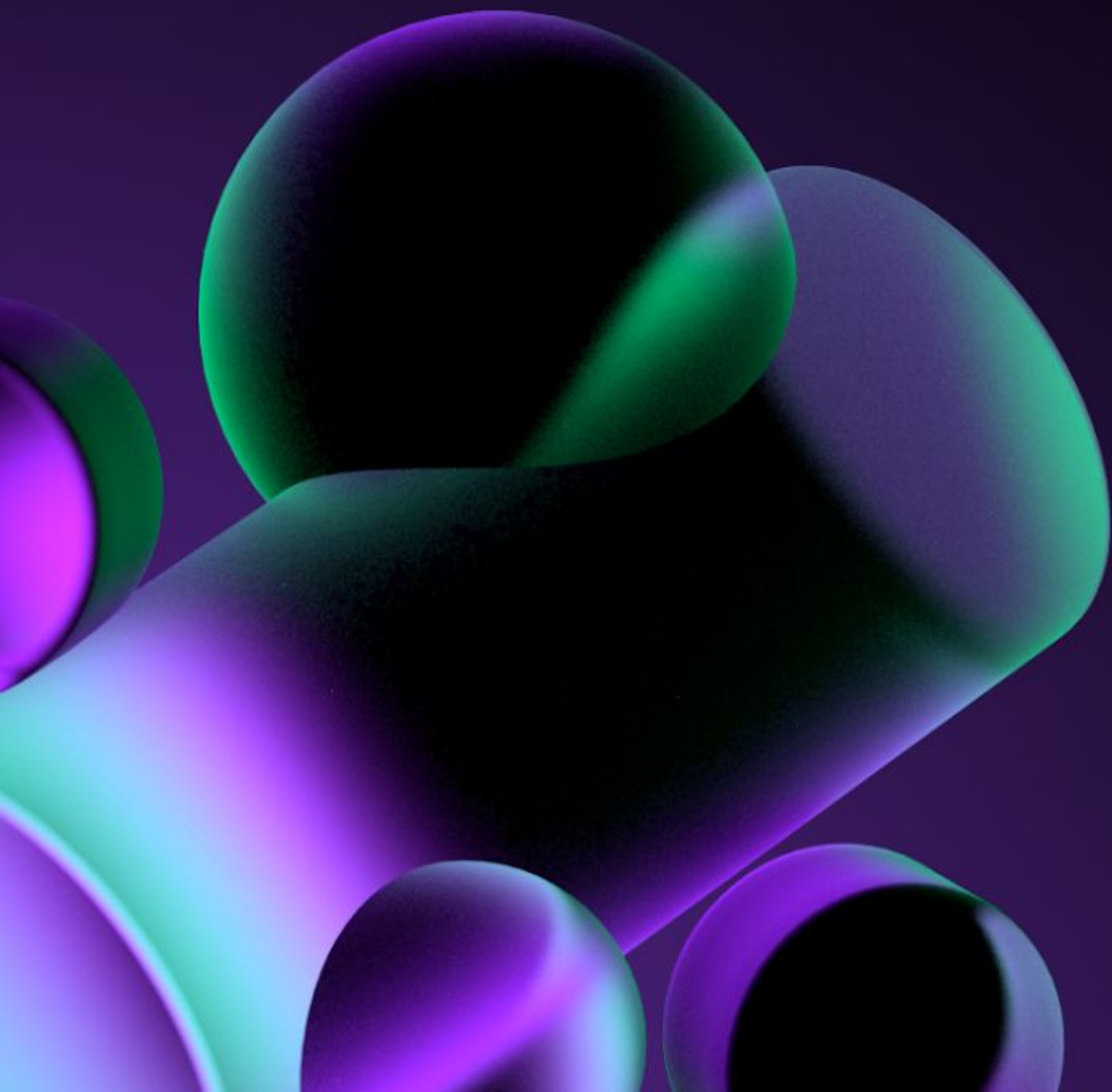
- Anchor integration tests using local test validator.
- Rust unit tests.
- Rust runtime/local validator integration tests.
- Fuzz tests.



## Trdelník

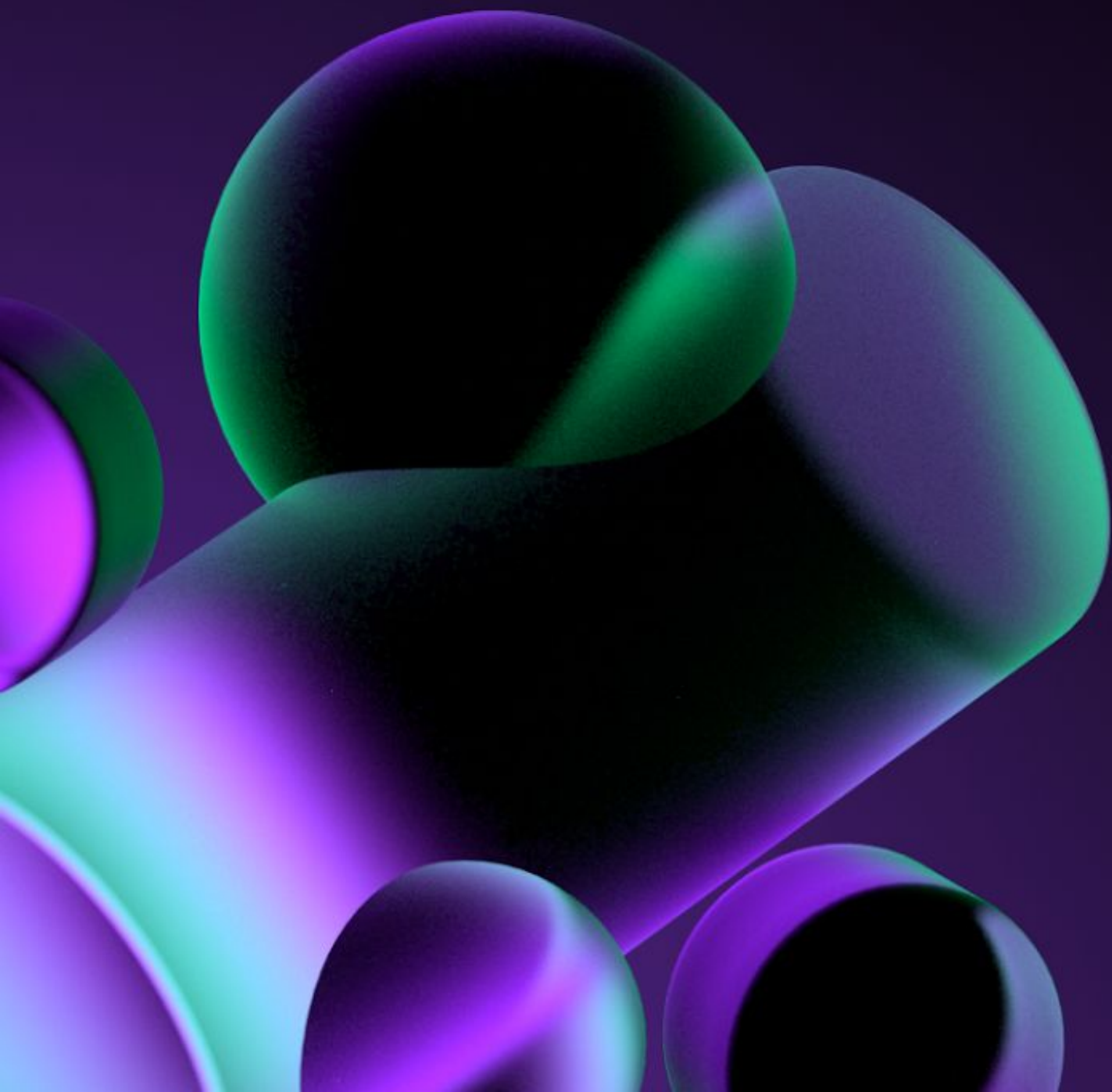
Trdelník is a Rust-based testing framework for Solana programs written in Anchor.

# Hands-on





# Task



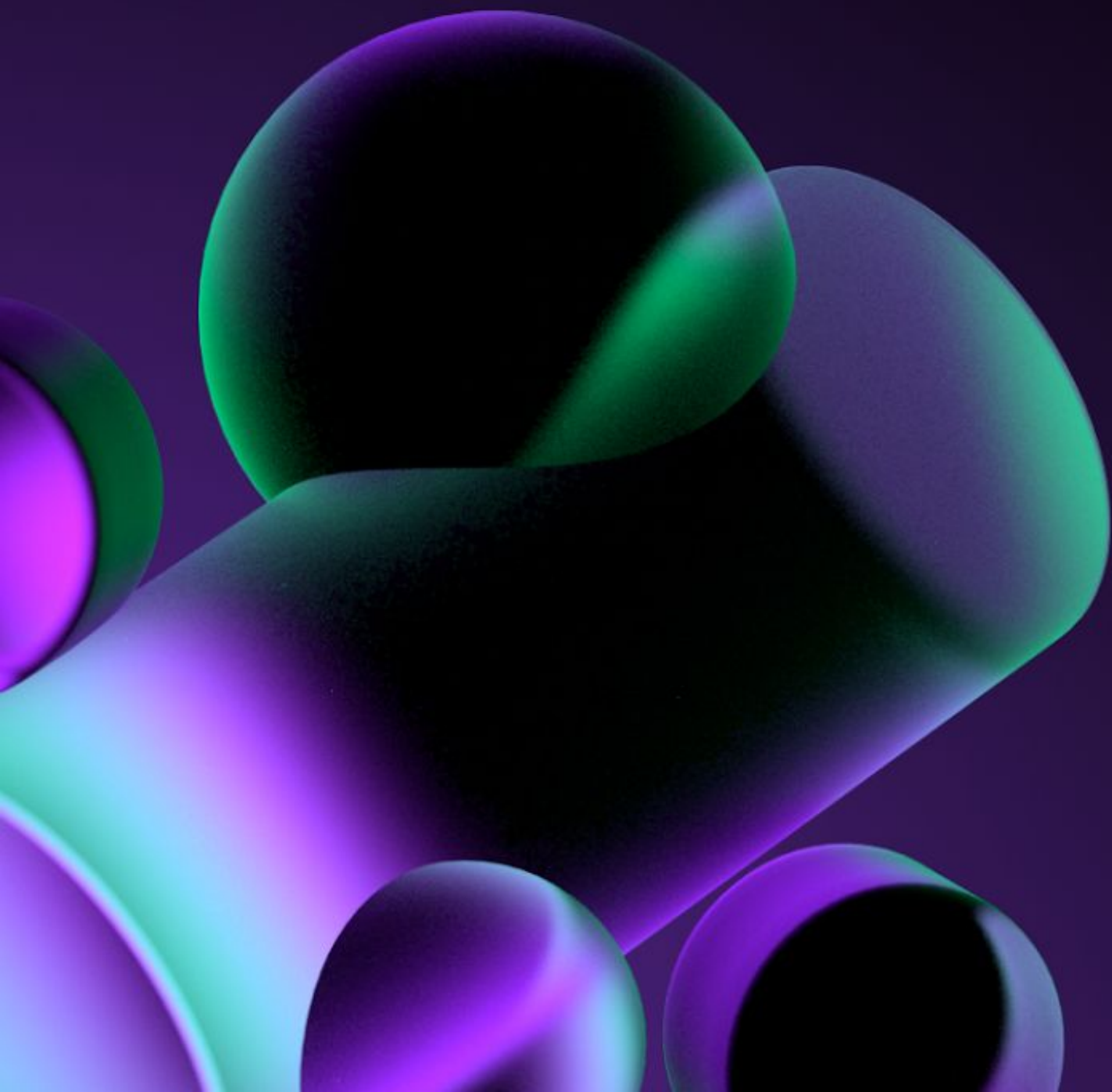


# Thank you

See you next time!



# Heading





# Heading

- List
  - list item
  - list item 2