

DeFi on Solana · Week 6

Speaker script: every slide with its notes. Facts checked 11 to 14 Oct 2026; re-check anything marked "verify".

1

Week 6 of 6

6

Capstone, demo & careers

Task

Blog post

Ship a capstone in your tier and present it on demo day.

My 6-week journey in Solana DeFi and my next steps into Web3.

DeFi on Solana · Week 6

Week 6 goal: everyone leaves with a finished piece of work they can show a stranger, and a concrete next step.

Timing: this week is less teaching and more coaching. 20 minutes on the capstone and careers, then work time, then demo day (in the same session or the next one, depending on your schedule).

Pick the demo day presenter by the clearest explanation, not the fanciest build.

2

Today

1

Pick a capstone tier

2

Ask a question evidence can answer

3

Where this leads

4

Demo day

→

Ship and present

Today

The capstone pulls together weeks 1 to 5. Every tier asks for the same thing: a clear question, real evidence, an honest list of risks, and a five-minute explanation a non-specialist can follow.

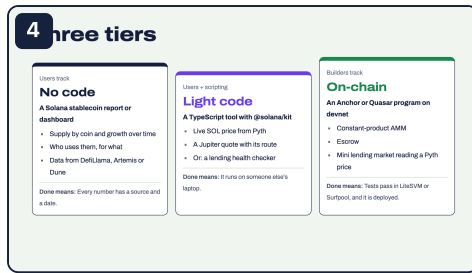
3

The capstone

Same rigour. Pick your tools.

The capstone

No notes. Let the slide speak.



Three tiers

All three are equally valid. The tier is about tools, not about how serious the work is.

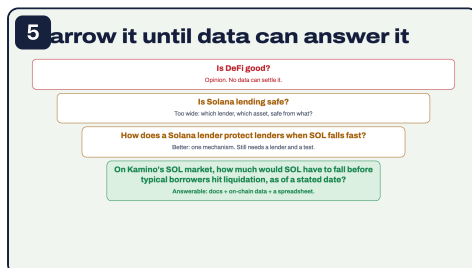
No code: a Solana stablecoin report or dashboard. Supply by coin, growth, who uses them. DefiLlama, Artemis or Dune. The bar: a date and a source on every number. For context, Solana stablecoin supply hit a reported record of \$17.3B on 25 Sep 2026, mostly USDC (handbook figure). They should pull their own number on the day they write.

Light code: a TypeScript tool with @solana/kit. Live SOL price from Pyth (the on-chain price account, as in builders/ts w4-pyth-price.ts), a Jupiter quote, and a "best route" readout. Or a wallet health checker for a lending position (week 4's spreadsheet, as a script).

On-chain: an Anchor (or Quasar) program. Constant-product AMM (week 3), escrow, or a mini lending market that reads a Pyth price (week 4). Tested with LiteSVM or Surfpool and deployed to devnet.

The curriculum's four capstone options (investment and risk analysis, ecosystem research comparing three protocols, a financial model in a spreadsheet, a beginner developer investigation) all fit inside these tiers. A finance graduate who builds an excellent lending model in a spreadsheet has a stronger portfolio piece than a copied AMM.

More on-chain ideas that fit six weeks: a time-locked escrow (maker deposits, taker completes, refund after a deadline), a bounty or contest pool priced in USD through an oracle and paid in SOL, or sponsored transactions so participants need no SOL. Each one reuses a week of this course: escrow is accounts and PDAs (week 2), USD pricing is oracles (week 4), and the security review is week 5.



Narrow it until data can answer it

Most weak capstones fail here, before any research starts.

Walk the funnel:

1. "Is DeFi good?" is an opinion. No dataset settles it.
2. "Is Solana lending safe?" Which lender? Which asset? Safe from what: hacks, liquidations, depegs?
3. "How does a Solana lender protect lenders when SOL falls fast?" One mechanism. Better. Still needs a specific lender and a test.
4. The answerable version names a protocol, an asset, a measurement and a date. It can be answered with the protocol's docs, on-chain data and a spreadsheet.

Other good questions from the curriculum: How does a decentralized exchange generate revenue? How sustainable are the returns of a particular pool? What risks arise when several protocols depend on the same stablecoin?

The example uses Kamino because it is a major Solana lender. Students should pick their own protocol.

6

our lenses on any protocol

Financial

Who plays, who earns, and is it sustainable?

Technical

Which accounts, programs and transactions?

Risk

How could users lose money? What stops it?

Strategic

Who uses it, and what could change that?

Primary

docs, explorer data, audits, the code

Secondary

dashboards, articles, threads

Four lenses on any protocol

Curriculum chapter 6.1, research area C. Every capstone covers all four lenses, even if one dominates.

Evidence: prefer primary sources (the protocol's documentation, the explorer, the audit report, the code itself). Use secondary sources to find leads, then confirm in primary data. Record access dates for anything that changes.

The habit from week 5 carries over: separate what you observed from what you think it means. A report that says "I could not verify X" is stronger than one that hides the gap.

7

uilders: will it still deploy?

Anchor

Pinocchio

Quasar

0.29.0

0.30.1

0.30.2

0.31.1

0.31.2

0.32.2

Will not deploy

Anchor 0.29.0: Old tutorials pin this. It will not deploy once SIMD-0500 is active.

Check the Anchor releases page before you start

Versions as of 6 Oct 2026. Solana moves to sBPFv3 in Q4 2026; after SIMD-0500 activates, older builds cannot deploy.

Builders: will it still deploy?

Two traps found while building builders/anchor (checked 14 Oct 2026). First, `anchor` runs whatever version avm has selected, not the one in Anchor.toml: run `avm install 1.2.1 && avm use 1.2.1` first. Second, a project scaffolded by Anchor 1.2.1 builds sBPFv3 programs, but its generated LiteSVM 0.10 tests cannot load them (InvalidAccountData). Use LiteSVM 0.16 and Rust 1.95, as builders/anchor does, or start from that workspace instead of `anchor init`. On Anchor 0.32.x, plain `anchor build` may still produce old-format programs: use `cargo build-sbf --arch v3`. SIMD-0500 is expected on mainnet with Agave 4.4 in Nov 2026; existing programs keep running, but new deploys and upgrades need sBPFv3.

From the handbook, checked 11 Oct 2026:

- Start on the latest Anchor 1.x: 1.2.1 as of 6 October 2026.
- Solana is moving to sBPFv3 in Q4 2026. Once SIMD-0500 activates, programs not built for sBPFv3 cannot be deployed.
- Supported minimums: Anchor 0.30.2, 0.31.2, 0.32.2 or 1.2.0+; Pinocchio 0.10; Quasar 0.1.0.
- Old tutorials pinned to Anchor 0.29 will break. If a student follows a 2023 tutorial, this is why it fails.

Anchor 1.0 (April 2026) made Surfpool the default test validator.

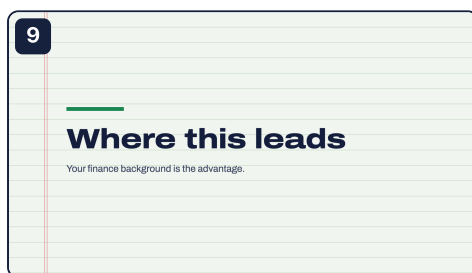
Click a few versions live, starting with 0.29.0, the version most old tutorials use.



The on-chain capstone, end to end

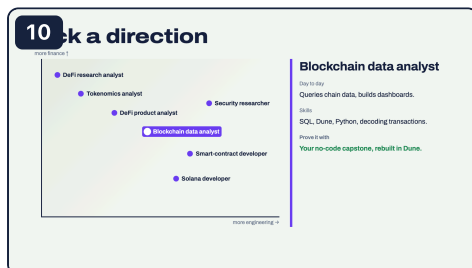
The order matters. Students who deploy first and test later spend demo day debugging.

1. Start from a reference: solana-developers/program-examples (tokens/token-swap is a constant-product AMM in Anchor), Blueshift's challenges (Anchor escrow, Pinocchio AMM), Cyfrin Updraft's Solana course.
2. Write the happy-path test first in LiteSVM: fast, in-process, no validator.
3. Then write at least one attack test from week 5: wrong signer, fake account, stale price. It must fail.
4. If the program reads real mainnet state (a Pyth feed, a Kamino market), use Surfpool: surfpool start forks mainnet accounts lazily as you touch them, with cheatcodes to fund wallets and move time.
5. Deploy to devnet.
6. For the demo, show one real devnet transaction on Orb with ? cluster=devnet. Reading it out loud proves it is real.



Where this leads

No notes. Let the slide speak.



Pick a direction

Click each role. The plot is a rough map, not a ranking: roles on the left need more finance, roles on the right more engineering.

Important for this group: most mentees come from accounting and finance. The left half of this map is where they already have an edge. A DeFi research analyst who understands credit risk, collateral and revenue quality is rare. Developers who can also read a balance sheet are rarer.

The "prove it with" line maps each role to work they have already done in this course. The six blog posts are a portfolio.

Career research task (curriculum chapter 6.2): pick two roles. List the skills they share, the skills that differ, and one project that would show readiness for each.

11 here to go next

Superteam Earn	Bounties, projects, grants. Nigeria-only writing and video bounties run often.	superteam.fun/earn
Turbin3	Free 6-week builders programme. Selective: under 30% get in.	
Blueshift	Free challenges with on-chain certificates.	
Cyfrin Updraft	Free Solana course, including security.	
Colosseum	Runs the main Solana hackathons. Check the schedule.	

Where to go next

Superteam Earn is the most direct path to getting paid. The blog posts from this course are exactly what writing bounties ask for. Superteam Nigeria often runs Nigeria-only bounties.

Turbin3 is the natural next step for builders: free, six weeks, selective (under 30% of applicants make it through).

Blueshift: free, open-source challenges with immediate feedback and on-chain certificates. Good for steady practice.

Cyfrin Updraft: free Solana course with an AMM module and security sections.

Colosseum runs the main Solana hackathons. Hackathons are where many people get hired; check the current schedule.

12

How I got here

Five minutes, your own story. From Turbin3 to working in the ecosystem.

How I got here

This is the most useful career content you can give them. Tell it straight:

- Where you started.
- The first thing you built or wrote that someone noticed.
- What Turbin3 was like and what it led to.
- One thing you would do differently.
- What you look for when someone new asks for help.

Keep it to five minutes and take questions.

13

Demo day

Five minutes. One question. Evidence.

Demo day

No notes. Let the slide speak.

14 Five minutes, five parts

0:30 The question	1:30 How it works	1:30 The evidence	1:00 The risks	0:30 Next
-----------------------------	-----------------------------	-----------------------------	--------------------------	---------------------

Show one real thing: a transaction, a chart with a date, a passing test.

Five minutes, five parts

Bar widths match the time. Most students overrun "how it works" and skip the risks; the structure stops that.

The question: one sentence, the narrowed one from the funnel.

How it works: the mechanism, with one diagram.

The evidence: the data, the model, or the running code.

The risks: what could go wrong and what they could not verify.

Next: one thing they will learn or build next.

Rule: show one real artefact. A devnet transaction on Orb, a chart with a date on it, a test run. Screenshots of other people's dashboards do not count.

15 Score yourself before you present

Quality of research and sources	Primary sources, cited sources, kept apart from assumptions.	20%	01234
Understanding of DeFi mechanics	Can explain how the thing works without slides.	20%	01234
Financial or technical analysis	A model, a calculation or working code, not just description.	25%	01234
Risk evaluation and critical thinking	Names what could go wrong and what you could not verify.	20%	01234
Communication and presentation	A non-specialist follows it in five minutes.	15%	01234

75 / 100

Self-assessment weights, not official grading.

Score yourself before you present

Weights from the curriculum's week 6 assessment: research and sources 20%, DeFi mechanics 20%, financial or technical analysis 25%, risk and critical thinking 20%, communication 15%. These are proposed self-assessment weights, not official WIBA grading criteria. Say so.

Use it two ways: each student scores themselves before presenting, then peers score them after. The gap between the two is the feedback.

16 Which capstone question is best?

A Is Solana better than Ethereum?

B How much would SOL need to fall before typical borrowers on one lender hit liquidation, as of a stated date?

C What is the future of DeFi?

D Is staking good?

Quiz: Which capstone question is best?

The others are opinions or too broad. Ask a student to narrow one of the wrong answers live, using the funnel.

17

Final blog post

"My 6-week journey in Solana DeFi and my next steps into Web3."

Link your five earlier posts and your capstone. Name one role you are aiming for and one next project.

Final blog post

This post is the cover letter for their portfolio. Linking the earlier posts turns six separate pieces into one story. Naming a role and a next project makes it actionable, and gives you something to follow up on.

18 Closing the books

1	Traced a Jupiter route	1 post
2	Sent devnet USDC, read it on Orb	1 post
3	Provided liquidity, built an IL table	1 post
4	Walked a liquidation by hand	1 post
5	Wrote a post-mortem, audited a token	1 post
6	Shipped and presented a capstone	1 post + demo

Balance carried forward

6 posts, 1 capstone, a public portfolio

Closing the books

Click through the six weeks as ledger entries, then the balance carried forward. It is a ledger because that is where this course started: from the bank's private book to a public one anyone can read.

Thank them. Tell them where to find you, and that the decks are open source on GitHub if they want to teach this themselves.

19

Teacher prep

For the mentor: Skip in class.

Teacher prep

No notes. Let the slide speak.

20 You, before class

- Prepare your own career story: five minutes, honest, with one thing you would do differently.
- Check Anchor's releases page and the sBPFv3 / SIMD-0500 status. Update the version slide if it moved.
- Check Superteam Earn for live bounties, especially Superteam Nigeria writing ones. Show real listings.
- Check Turbin3's next cohort dates and Colosseum's hackathon schedule.
- Agree the demo day format: time per student, who scores, how the presenter is picked.

You, before class

The careers half of this week lands much better with live, current links: an open bounty they could apply for tonight, a cohort with a real deadline.

Demo day logistics: with more than 10 students, split into two rooms or two sessions. Five minutes each plus two for questions adds up fast.

21 Verify before class

Toolchain versions move

Anchor 1.2.1 and the sBPFv3 timeline were current on 6 to 11 Oct 2026. Re-check before telling builders which version to use.

Programme details change

Turbot's acceptance rate, Superstorm bounty types and hackathon dates come from handbook research. Confirm them on the day.

Rubric weights are a suggestion

The 2020/25/2015 weights are proposed self-assessment weights, not official programme grading. Use the programme's rules where they differ.

Verify before class

Anything in this deck that names a version, a date or a programme detail should get a fresh look the week you teach it.
