

Solana Developer Workstation by Code Creator

User Guide and Launch Instructions
AWS Marketplace AMI | Ubuntu 24.04

This guide is written for AWS Marketplace customers who want a ready to use Solana development server on Ubuntu 24.04. The AMI is designed for local development, testing, Anchor based program work, and quick command line onboarding.

Publisher	Code Creator
Primary use	Solana local development and program testing
Preinstalled tools	Solana CLI, Rust, Anchor CLI, AVM, Surfpool, Node.js, Yarn
Starter workspace	/opt/codecreator-solana/workspace/my-solana-app
Default local RPC	http://127.0.0.1:8899
Public inbound port	TCP 22 for SSH

Quick Start

The simplest way to use this AMI is to launch the instance, connect by SSH as ubuntu, read the first login note, start the local validator, and build the included starter project.

Step	What you do	What success looks like
1	Launch the AMI and allow SSH	You can connect with your Amazon private key and the ubuntu user
2	Read the first login note	You see the workspace path and helper commands
3	Start the local validator	RPC is available on 127.0.0.1:8899
4	Build the starter project	Anchor completes a local build successfully
5	Start your own project or template	You are ready to build Solana applications

Before You Begin

- Launch a supported EC2 instance with a public IP address.
- Open inbound TCP port 22 in the security group for SSH access.
- Connect to your instance using your Amazon private key and the ubuntu user. OS user name: `'ubuntu'`
- This product is a developer workstation. It does not require a public web port for normal use.
- A practical starting size is t3.large. A stronger default for steadier use is m6i.large.

Launch and First Login

After the instance passes AWS status checks, connect by SSH:

```
ssh -i /path/to/your-key.pem ubuntu@YOUR_PUBLIC_IP
```

After login, view the first boot summary:

```
sudo cat /root/FIRST_LOGIN.txt
```

Check the installed toolchain

```
solana --version
rustc --version
anchor --version
avm --version
surfpool --version
node --version
yarn --version
```

Use the included starter workspace

Move into the preinstalled sample project:

```
cd /opt/codecreator-solana/workspace/my-solana-app
```

Build the sample Anchor project:

```
anchor build      (be patient)
```

Start local development

The default local development path in this AMI uses a helper script that starts a stable local validator and configures the Solana CLI for localhost.

Start the local validator:

```
cc-solana-localnet-start
```

Check status, wallet address, and local balance:

```
cc-solana-status
```

Stop the local validator when you are done:

```
cc-solana-localnet-stop
```

Useful Solana CLI checks

```
solana config get
solana cluster-version
solana airdrop 10
solana balance
```

Create a new Anchor project

Move back to the workspace folder and initialize a new project:

```
cd /opt/codecreator-solana/workspace
anchor init your-project-name
```

Create a broader Solana application template

If you want a fuller dApp template with frontend options, use the official create solana dapp scaffolding tool:

```
cd /opt/codecreator-solana/workspace
npm create solana-dapp@latest
```

Advanced option: Surfpool

Surfpool is also installed for advanced local testing and compatibility workflows. For this AMI, the default recommended validator path is the helper script above because it was the more reliable baked workflow during validation.

```
surfpool start --legacy-anchor-compatibility --no-tui --log-level debug
```

Helpful paths and runtime items

/opt/codecreator-solana	Main product directory
/opt/codecreator-solana/workspace/my-solana-app	Included starter Anchor project
/usr/local/bin/cc-solana-localnet-start	Starts the local validator and configures localhost RPC
/usr/local/bin/cc-solana-localnet-stop	Stops the local validator
/usr/local/bin/cc-solana-status	Shows local validator status and wallet details
/root/FIRST_LOGIN.txt	First boot summary for the instance owner
/home/ubuntu/FIRST_LOGIN.txt	Copy of the first boot summary for the ubuntu user
/etc/systemd/system/codecreator-solana-firstboot.service	First boot initialization service

Troubleshooting

FIRST_LOGIN.txt is missing

Check the first boot service:

```
sudo systemctl status codecreator-solana-firstboot.service --no-pager
```

Solana commands are not on the path in the current shell

Load the profile helper:

```
source /etc/profile.d/codecreator-solana.sh
```

The local validator is not reachable

Restart it and recheck status:

```
cc-solana-localnet-stop  
cc-solana-localnet-start  
cc-solana-status
```

You want a fresh local wallet for testing

Remove the wallet file and start localnet again:

```
rm -f ~/.config/solana/id.json  
cc-solana-localnet-start
```

Recommended EC2 sizes and storage

Root storage	30 GB or more
Small test and evaluation	t3.large
Recommended default	m6i.large
Heavier developer use	m6i.xlarge

Official resources

- Solana documentation: <https://solana.com/docs>
- Solana quick start: <https://solana.com/docs/intro/quick-start>
- Anchor documentation: <https://www.anchor-lang.com/docs>
- Solana Stack Exchange: <https://solana.stackexchange.com>

Important note before publication

Before publishing or updating the AWS Marketplace listing, launch a fresh instance from the final AMI and test it exactly like a customer would. Confirm that the first boot service recreates FIRST_LOGIN.txt, the helper scripts work, and the local validator can be started on a new instance.