HIERO

PYTHON SDK

AGENDA

01

RELEASE STRATEGY

02

BRANCHING, COMMITTING, MERGING

03

VERSION SUPPORT, README & OTHERS

RELEASE STRATEGY

01 Version tag & increments

MAJOR.MINOR.PATCH (eg. 0.1.0 for first release, as not fully fledged SDK)

- MAJOR: Incompatible API changes
- MINOR: Backward-compatible functionality added
- PATCH: Backward-compatible bug fixes

02 Release

- every 2-4 weeks
- merge into /release branch
- deploy from /release branch
- CHANGELOG update

BRANCHING, COMMITTING & MERGING

BRANCHING

trunk based system

feature/

(hot)fix/

release/

COMMITTING

- **feat:** adds a new feature.
- **fix**: fixes a bug.
- **docs:** only changes or adds documentation.
- **chore:** doesn't affect source code or tests (eg updating build scripts or package dependencies)
- **test:** adds or modifies tests
- **refactor:** refactors code without adding new features or fixing bugs
- **style:** code styling, without changing logic.

MERGING

- **Squash & Merge:** Keep as it is to avoid clutter
- **GPG key setup & signing off commits**:

https://docs.github.com/en/authentication/managing-commit-signature-verification/generating-a-new-gpg-key

-> eg commit with:

git commit -s -S -m "added docstrings"

PYTHON VERSION + README

01

PYTHON VERSION

Do we stick with 3.9.2 until it is no longer supported? Or, because it will take a few months to finish the SDK anyway, do we target 3.10 and worry about 3.11 in 18 months?

02

README GOING FORWARD

take code examples / Transactions out and only focus on project overview, how to run, contributions

FURTHER DISCUSSION POINTS

CONCLUSION, KEY TAKEAWAYS, QUESTIONS?