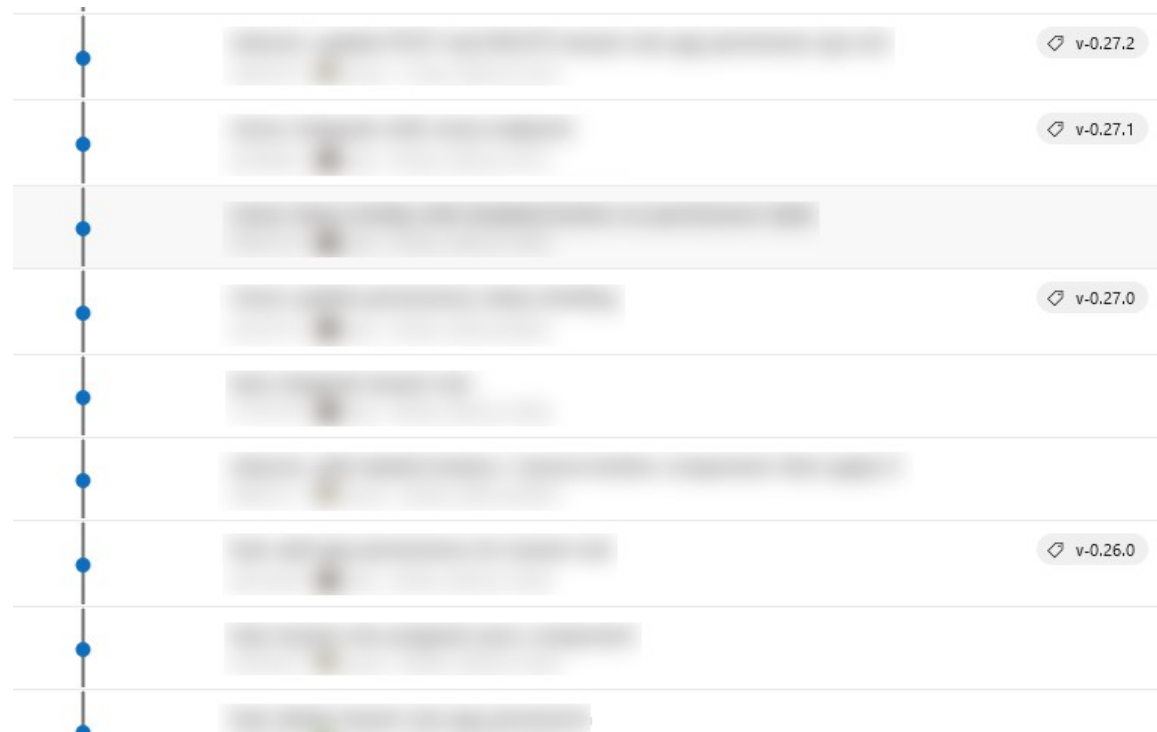# Presentation on automatic versioning release through git tag

What to expect of presentee

- I will go through the 3 versioning flows, after each i will stop and answer some questions regarding those flows.
- There will be a Dicussion session after i have presented the 3 flows.
- No interruption to dicussion problem until Dicussion page.

Table of Contents

- Flow

  - Current flow
  - Proposal 1
  - Proposal 2

- Discussion

Flow

Current flow

1. release triggered
2. devops ask for commit to release
3. devops generate changelog and write release note
4. devops select build from release commit
5. devops deploy build to environments

Flow

Proposal 1

1. release triggered
2. dev generate changelog and commit it in source
3. dev tag the release commit
4. dev notify devops with release tag
5. devops select build based on release tag
6. devops deploy build to environments

notes:

- every commit to master will be built, those with tag will be built into tagged build
- 3. -> commit here mean pull request into master, that tagging happen in the merging branch

```
                                /- transfered from pr → tagged "v-0.1.0" → build into "build #0.1.0"
master (*)----------------(*)----->
         \                  / → pull request
feat:     \---(A)---(B)---(C)
                          \-tagged "v-0.1.0"
```

Flow

Proposal 2

1. release triggered
2. dev generate changelog and commit it in source
3. dev tag the release commit
4. dev notify devops with release tag
5. devops select build based on release tag
6. devops deploy build to environments

notes:

- only commit from master that is tagged will be build

- 3. -> commit here mean commit in master, that tagging happen with the merged master commit

```
                               /-tagged "v-0.1.0" → build into "build #0.1.0"
master (*)----------------(*)----->
        \                   / → pull request
feat:    \---(A)---(B)---(C)
```

Discussion:

- who (or what) will generate the changelog

- changelog format

- communication channel to transfer the changelog to devops

- version numbering: **Semantic Versioning**

- solution to versioning commit and release in source code

- git tag format

- multi project repo: PCS-API and AMS-API

- tool for managing git tag

Changelog format

Proposed:

```
[version number]
  Features:
    PBI 1234 Implement ABC XYZ
  Bug fix:
    Bug 5678 Wrong DEF when access IJK
  Miscelaneous:
    * Add test user
    * Refactor Authorization module
```

Communication channel to transfer the changelog to devops

Proposed:

- In a file called CHANGELOG.MD in the root of source code
- In the commit messages of the PR

Solution to versioning commit and release in source code

Proposed:

Front-end:

- autobump: bump based on feature commit count
- manually bump in package.json

Back-end:

- manually bump in csproj

Git tag format

Proposed:

prefix with "v-"

Multi project repo: PCS-API and AMS-API

Proposed:

commit for PCS-API is prefixed with "v-"
commit for PCS-API is prefixed with "ams-v-"

Tool for managing git tag

Proposed:

Azure Devops/Repos/Tags

QA Time!

The End