Minimal Slices

Always Deliver Your Work Using Minimal Valuable Slices

- Split your work item into small chunks that are contributed in incremental commits.

- Contribute your chunks frequently. Follow an iterative approach by regularly providing updates and changes to the team. This allows for instant feedback and early issue discovery and ensures you are developing in the right direction, both technically and functionally.

- Do NOT work independently on your task without providing any updates to your team.

Example

Imagine you are working on adding UWP (Universal Windows Platform) application building functionality for existing continuous integration service which already has Android/iOS support.

Bad Approach

After six weeks of work you created PR with all required functionality, including portal UI (build settings), backend REST API (UWP build functionality), telemetry, unit and integration tests, etc.

Good Approach

You divided your feature into smaller user stories (which in turn were divided into multiple tasks) and started working on them one by one:

- As a user I can successfully build UWP apps using current service

- As a user I can see telemetry when building the apps

- As a user I have the ability to select build configuration (debug, release)

- As a user I have the ability to select target platform (arm, x86, x64)

- ...

You also divided your stories into smaller tasks and sent PRs based on those tasks. E.g. you have the following tasks for the first user story above:

- Enable UWP platform on backend

- Add `build` button to the UI (build first solution file found)

- Add `select solution file` dropdown to the UI

- Implement unit tests

- Implement integration tests to verify build succeeded

- Update documentation

- ...

Resources

- [Minimalism Rules](#)