FINDOLOGIC

stop searching - find!

FINDOLOGIC goes Scrum II:
How to plan the unplannable.

Patrick Rappensberger | Lead Infrastructure Developer

How do we develop at FINDOLOGIC?

- Agile Software Development / Scrum.

- Two weekly sprints:

  - PO<->LxD Meeting

    - Planning

  - Planning Poker

  - Retrospective

- Pre-/Review system.

- Homie system for Support/Project Management.

How to merge sprint planning with several sideline jobs?

• Development outside the sprint. (e.g. Plug-in,

Infrastructure)

• Maintaining our infrastructure.

○ Incident handling

○ Deployment

• Other side jobs / bug investigation.

Why is this important?

- To ensure that the sprint can be completed as expected.
    - Enables long time planning / roadmaps.
    - To avoid too much workload for the team.

What do we want before the next sprint?

- An amount of hours than can be invested in it.
- An amount of hours to do the necessary sidetime jobs.

How to determine the necessary time outside the sprint?

• Estimate necessary time.

• Track sideline jobs:

  ○ Incident log

  ○ Disturbance log

  ○ Dev setup log

• Adopt the times after some sprints!

  ○ Use the logs!

Incident log

- When was the incident?

- Which servers were affected?

- Which services were affected?

- Who handled the incident?

- What happened?

- What's the reason for it?

- What was done?

- How much time was spent?

Incident Log (Example)

Begin	End	Servers	Services	Who	What	Why	Actions	Time spent (min)
2018-06-11 10:23 CEST	2018-06-11 10:30 CEST	dummy.server	Hugo	P.	Jetty Thread pool utilization was unusually high.	Unknown	* Checked if any import is running. * Recovered from itself.	5
2018-06-05 06:00 CEST	2018-06-05 09:32 CEST	dummy2.server	Importer	S.	The nightly import wasn't successful on dummy2.server.	We have a bug in the current version of the Importer that caused an fatal error. Therefore the timestamp file wasn't written.	* Investigated automatic-import log and notified Isi. * Downgrade REST exporter.	30
2018-03-12 01:36 CET	2018-03-12 09:56 CET	dummy3.server	Importer	P.	High CPU load.	CPU intensive import of shop: XXX	Killed it, notified Support and re-started it again.	10
2018-02-22 14:50 CET	2018-02-22 15:05 CET	dummy.server dummer2.server	Search Stack	P.	Servers seem to respond slower than usual. It seems that all servers are affected.	Most likely a network issue, but this is just an assumption. I wasn't able to reproduce the issue.	Tried to find the reason. It seems that we have sometimes problems on our search servers, so that some configuration can't be read.	20
2018-02-08 10:21 CET	2018-02-08 10:27 CET	dummy.fallback	Hugo	C.	Core of XXX could not be loaded.	?	Unloaded the core and started new fallback update.	5
2017-12-14 06:00 CET	2017-12-14 08:05 CET	dummy.backend	Importer	P.	Automated importer check was unable to read the output.	A day old import container was still running.	Stopped the old container.	10

What to do yourself and what to put into a sprint?

- Basically Experience is necessary for it.

- How important is it?

- How much effort is it to fix?

- How thoroughly has the solution to be testes?

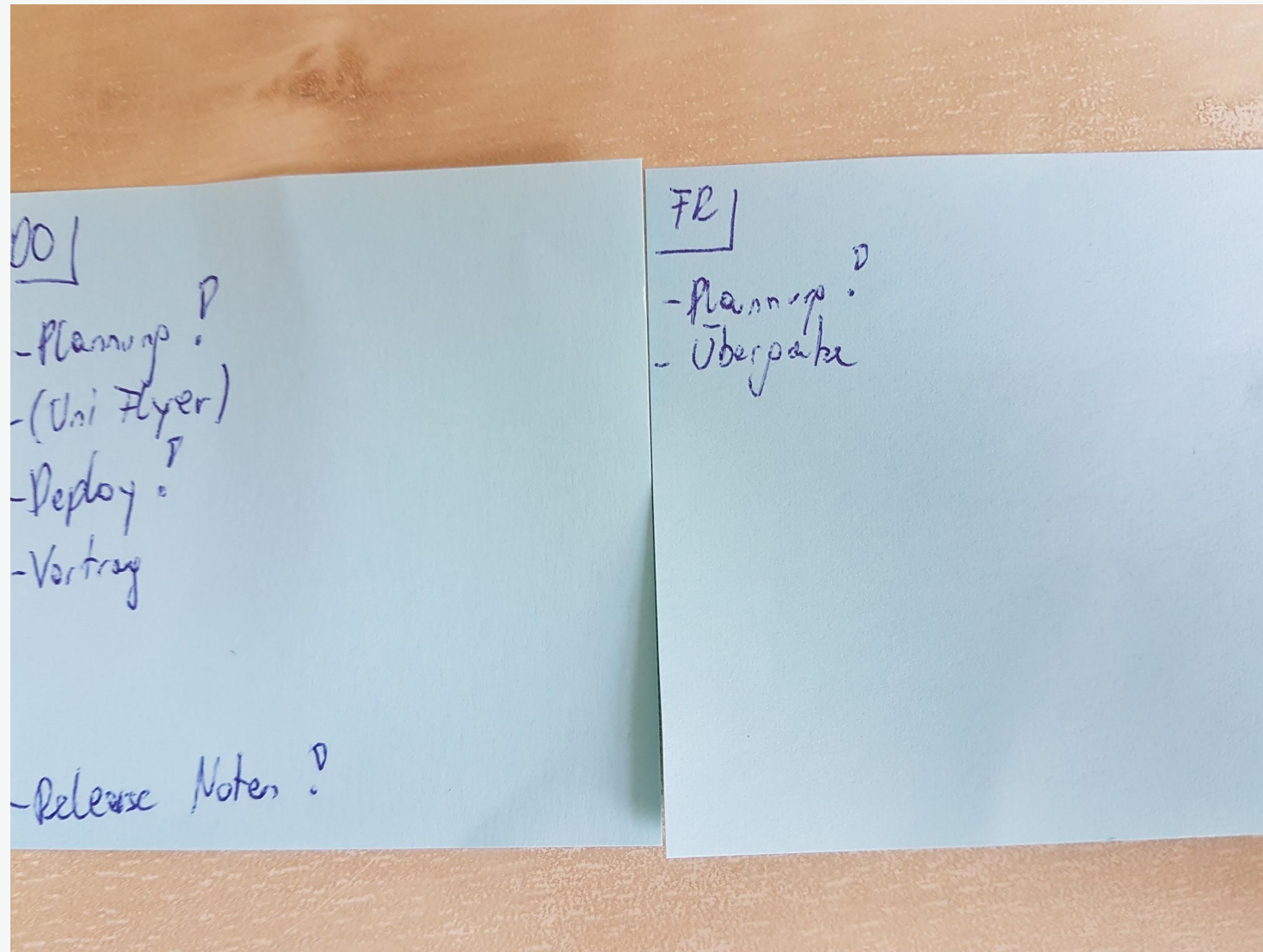- Is it likely that it will be in one of the next sprints?

How to get things done?

- Prioritize everything!

- Get rid of unnecessary tasks.

  - Automize them! (e.g. backup tasks)

- Plan ahead and schedule everything!

  - Reserve time slots in the calendar.

  - PotD

PotD (Post-it of the day)

- Always have a Post-It for Today and one for the next day.
  - Write down what has to be done.
  - Prioritize the tasks and mark optional ones.
  - Write the tasks for tomorrow during the day.
  - Check done tasks.
  - Transfer tasks that weren't done this day to the next day (if necessary).

Post-it of the day.

Limitations of the system

- Some tasks have to done, regardless of the already invested time.
- Development outside the sprint:
    - Quality?
- You have to be stress robust and focused.

Personal experience

- Learn to say "No".

  - Not everything is as important as it seems and has to be fixed immediately.

- Block times in the calendar and schedule everything.

- Adapt time that is necessary outside the sprint.

- PotD works quite well.

- Prepare meetings!

- Hard to prioritize tasks.