

-- KANBAN FLOW ONLINE SIMULATION --



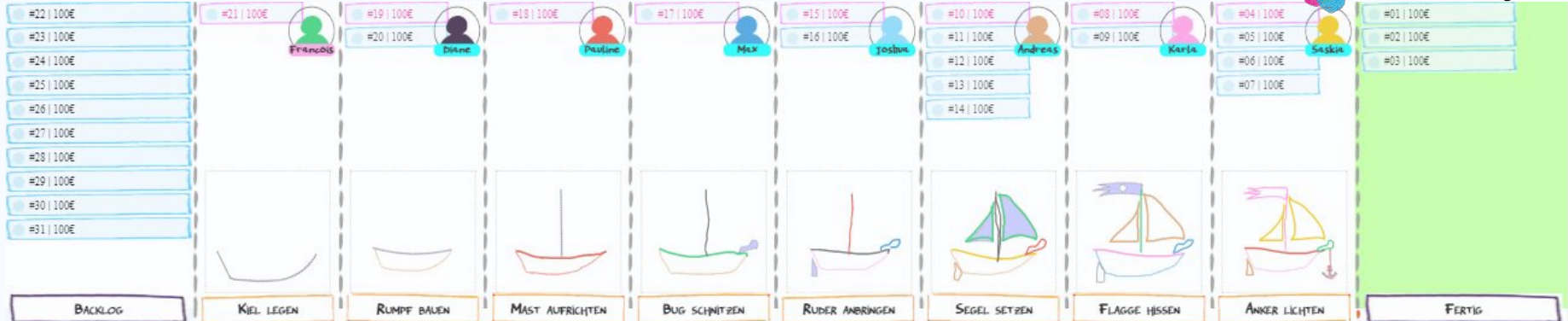
Simuliert wurde eine Produktionsstrasse für gemalte Schiffe.

Zwei Runden wurden durchgeführt, je 4 Minuten. Anschließend wurde jeweils das letzte Schiff (rotes Schiff) in das System eingeführt. Die Runden waren beendet, sobald das rote Schiff fertiggestellt wurde.

Anhand der generierten Daten und der gemachten Erfahrungen wurden folgende Punkte reflektiert:

- Unlimitiertes Push vs. WIP limitiertes Pull
- Bottleneck (Engpass)
- Systemische Betrachtung der Zusammenarbeit in komplexen Umfeldern

[HTTPS://WWW.KOMPLEXITAETER.DE](https://www.komplexitaeter.de)





FRANCOIS



LARS



BRIGITTE



LARA

Nr. 2



MELLI



SONJA



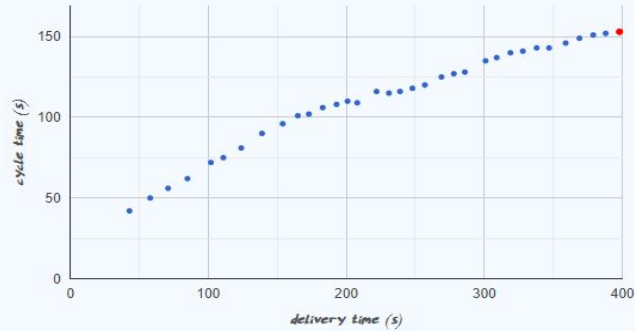
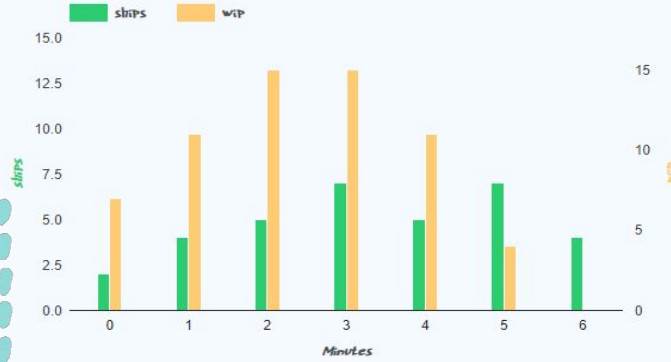
ANDRÉ



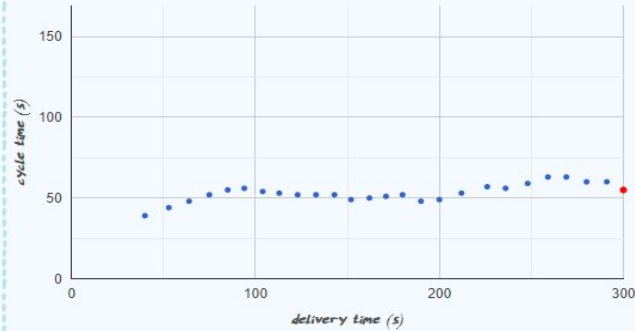
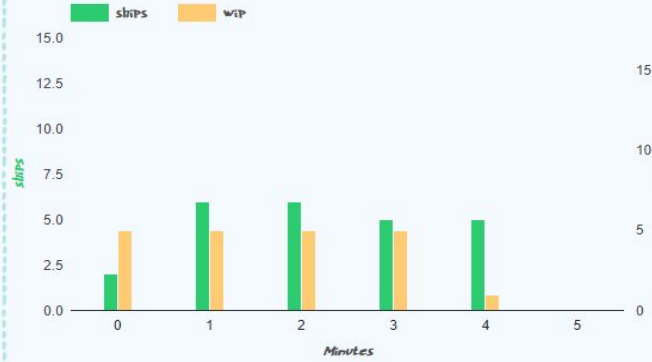
ANDREAS

Nr. 4

1. RUNDE
PUSH
UNLIMITIERT
4 MINUTEN



average Troughput = 5.6



average Troughput = 5.8

2. RUNDE
PULL
WIP LIMIT = 1
4 MINUTEN

BEGRIFF DEFINITION

UND... ?

Backlog Menge der noch nicht gestarteten Arbeit

Im Push wird die Wartezeit aus dem Backlog ins System verlagert. Im Pull nicht.

Engpass Der Arbeitsschritt, der am meisten Zeit benötigt

Der Engpass bestimmt den Durchsatz des gesamten Systems.

Work in Progress (WiP) Menge aller Arbeit aktuell im System

Die Limitierung des WiPs erhöht die Stabilität des Systems (siehe Littles Gesetz).

Durchsatz Menge fertiggestellter Arbeit pro Zeiteinheit (Schiffe/Minute)

Der Durchsatz ist in beiden Systemen (Push- und Pull-Modus) gleich.

Durchlaufzeit Zeit zwischen Start der Arbeit und deren Fertigstellung

Die Durchlaufzeit bestimmt wie lange ein Kunde auf sein Produkt wartet.

Littles Gesetz $\text{Durchlaufzeit} = \text{Work in Progress} / \text{Durchsatz}$

Bei gleichen Kapazitäten erhöht sich die Durchlaufzeit bei jedem zusätzlichen Auftrag im System.

Pull Nach Fertigstellung wartet die Station darauf, dass die nächste Station den Auftrag zu sich zieht

Durch limitiertes Pull wird das System genauso so ausgelastet, wie es gerade kann.

Push Nach Fertigstellung schiebt die Station den Auftrag direkt zu der nächsten Station

Durch unlimitiertes Push wird das System kontinuierlich befüllt, auch über seine Kapazitätsgrenze hinaus.

GEDANKENEXPERIMENTE

Was passiert, wenn man eine Station optimiert, die nicht der Engpass ist, sondern sich vor diesem befindet?

Wie wäre die Durchlaufzeit des roten Schiffs, wenn man es erst nach 15 Min einführt?

Was passiert mit der Durchlaufzeit der anderer Schiffe, wenn wir ein Schiff priorisieren?

Was passiert mit dem Durchsatz, wenn jemand, der nicht Experte für diese Station ist, am Engpass unterstützt?

Angenommen es wollen alle Kunden ab jetzt nur noch grüne Schiffe. Was wären die Konsequenzen in beiden Systemen?



ERGEBNIS



Im unlimitierten Push-Modus wächst die Warteschlange am Bottleneck umso schneller. Die Durchlaufzeit im Gesamtsystem steigt dadurch noch stärker an. Im limitierten Pull-Modus hat es keinen Effekt, außer dass ggf. die schnellere Station noch mehr Zeit hat, um am Engpass zu unterstützen.

Im unlimitierten Push-Modus wächst die Durchlaufzeit stetig. Damit wäre sie nach 15 Minuten bei ca. 10 Minuten (!). Im limitierten Pull-Modus bleibt die Durchlaufzeit stabil. Somit hätte dies keine Auswirkung.

Im unlimitierten Push-Modus wird der priorisierte Auftrag an jeder Station vorgezogen. Somit steigt die Durchlaufzeit aller anderen Aufträge. Im limitierten Pull-Modus wird der priorisierte Auftrag als oberster im Backlog platziert. Somit hat es keinen Effekt auf die bereits angefangene Aufträge. Der Start der Folgeaufträge aus dem Backlog verzögern sich jedoch entsprechend des Durchsatzes des Systems.

Auch wenn der Nicht-Experte nur 10% so schnell wie der Experte arbeitet, erhöht sich der Durchsatz des Gesamtsystem um 10%.

Im limitierten Pull-Modus müssen alle angefangene Aufträge über Bord geworfen (*sunk costs*) oder ohne Absatzchance am Markt fertiggestellt werden. Im unlimitierten Push-Modus gilt das Gleiche, allerdings ist die Menge der angefangenen Arbeit drastisch geringer (bei WiP = 1, sind etwa so viele Aufträge im System, wie es Stationen gibt).