

Übungsblatt 2

Software Engineering (für ST) (SoSe 2018)

- Lan Schlufte Cayapa, 3222681, st149553@stud.uni-stuttgart.de
-
- Lion Wagner, 3231355, st148345@stud.uni-stuttgart.de
-
- Jannis Westermann, 3288735, st153444@stud.uni-stuttgart.de
-
- Lukas Pietzsch, 3227178, st151191@stud.uni-stuttgart.de

Aufgabe 1

(a)

1. Begeisterung. Ein Merkmal, dass wenn es vorhanden ist sehr gut ist, dem Kunden aber nicht unbedingt fehlt.
Neues Betriebssystem.
2. Leistung. Ebenfalls ein Merkmal das den Kunden sehr zufrieden stellt und nicht fehlen darf.
Großer interner Speicher
3. Basis. Merkmal, das vom Kunden als selbstverständlich angesehen wird.
Touchscreen/Internet.
4. Unerheblich. Merkmal das der Kunde gerne annimmt, es aber auch nicht vermissen würde.
Filterfunktion in der Kamera.
5. Rückweisung. Merkmal das dem Kunden absolut nicht gefällt, unabhängig von den Vorteilen wird es abgewiesen.
Diskettenlaufwerk einbauen.

(b)

Was würden Sie davon halten, wenn das Smartphone nicht über einen Touchscreen verfügt.

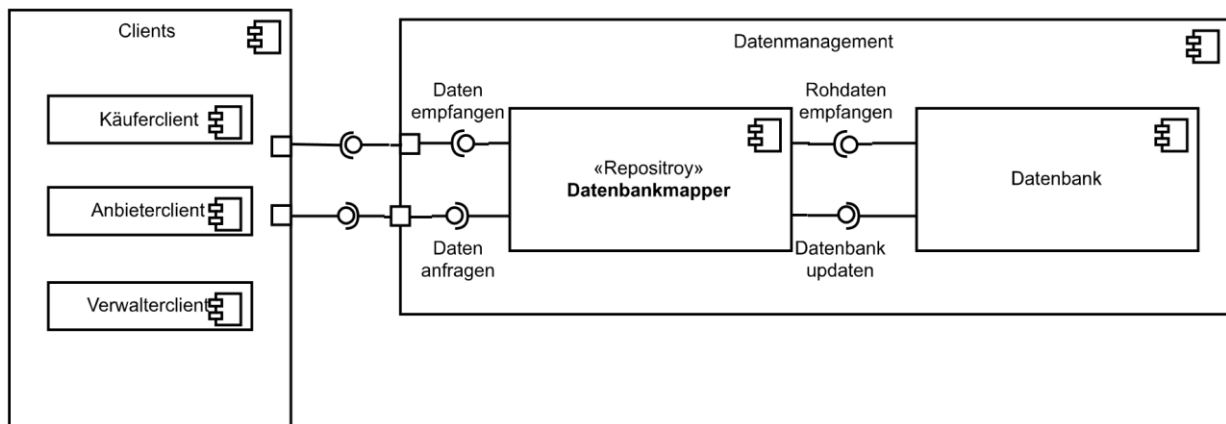
(c)

Was würden Sie davon halten, wenn das Smartphone über einen Touchscreen verfügt.

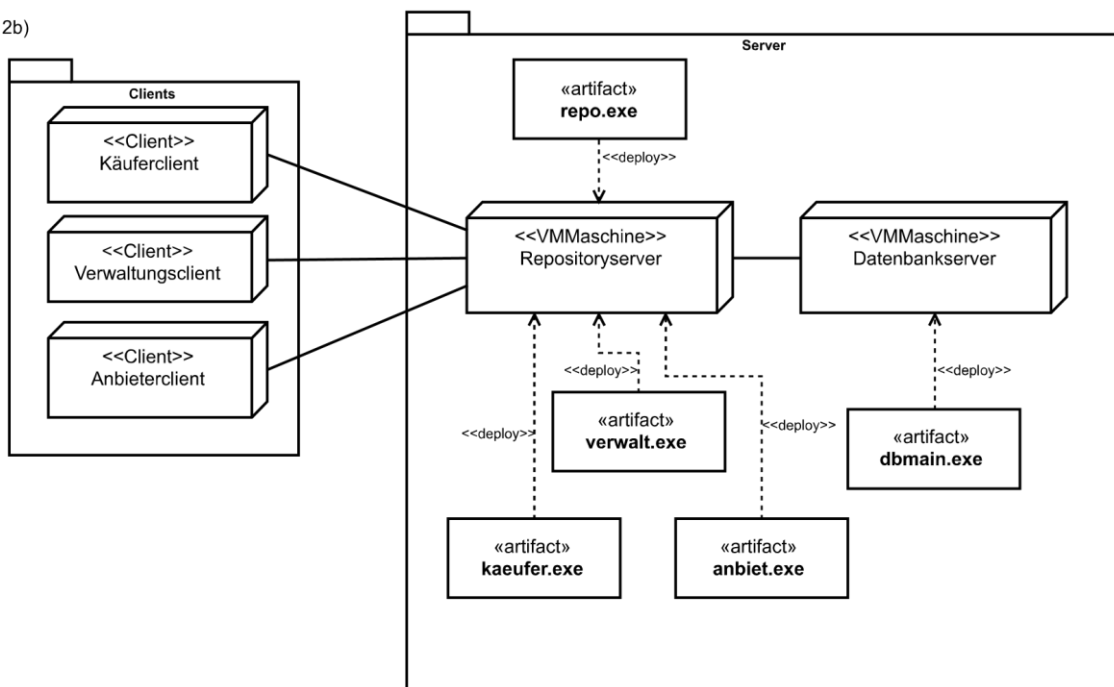
(d)

Begeisterungsmerkmale rutschen nach und nach, aber sicher zu Basismerkmalen ab. Ein Beispiel dafür wäre der oben genannte Touchscreen.

2a)



2b)



2c)

Die Schwachstelle ist vermutlich der Repositoryserver, weil er alle Anfragen von allen Clients explizit und getrennt bearbeiten muss.

2d)

Ein Client-Server-Modell würde sich ebenfalls anbieten. Dabei kann man für jeden Akteurtyp einen eigenen Server zu Verfügung stellen kann, welcher die spez. Anfragen schneller bearbeiten kann.

3)

(a) Programmierer 3 ist der Architekt und vermutlich ein „Entwurfsspezialist“. Er wird von den andern für sein Wissen als guter Entwickler geschätzt und hat eine gute Vorstellung vom gesamten Systementwurf.

(b) Beim Microservices wird das System in verschiedene Module aufgeteilt, welche über fest genormte, programmiersprachenunabhängige Schnittstellen miteinander kommunizieren. Module haben immer eine spezielle Aufgabe und können einfach ausgetauscht werden.

(c)

Positiv	Negativ
+ zunächst gute Wartbarkeit durch Microservices	- Wahrscheinlich Overengineered (Granularität)
	- später schlecht Wartbar, da nur Programmierer 3 sich gut auskennt (Verstehbar/Explizit)
+ Kunden und Verkäufer bekommen relevante Informationen (Stakeholder)	- Neue IT muss installiert werden (Stakeholder)
+ guter Kompromiss aus neuer Architektur und Funktionen (Gut ausbalancierte Tradeoffs)	- ständig wechselnde Sprachen und Ausfall von Programmierer 3 -> Systemverfall (Persistenz)

(d)

- Programmierer 3 geht bald in Rente
- Zu wenig Personalressourcen
- Installateur ist schlecht ins Unternehmen mit einbezogen
- keine Konsistenz in der Programmiersprachen und damit wahrscheinlich auch in anderen Bereichen
- In der Vergangenheit schlechtes Risikomanagement/-analyse, sonst wären früher weniger teuer Fehler passiert

(e)

Das Projekt wird wahrscheinlich abgeschlossen werden, aber in einem schlechten Zustand sein. Das kommt daher, dass anscheinend die Programmierer 1 und 2 schlecht in das Projekt einbezogen wurden. Zusätzlich wird System wahrscheinlich mit dem Renteneintritt von Programmierer 3 sowieso bald verfallen.

(f)

- Besseren Informationsfluss/-übersicht schaffen
- Programmierer 3 sollte jemanden in das neue System einlernen
- Mehr Personal einstellen
- Richtlinien schaffen (Programmiersprache, Projektablaufpläne)
- Risikomanagement etablieren