

Domain Model and Sequence Diagram

Software Engineering & Design
Case Study 2 – Task 2

Web-App für Angehörige von Alzheimererkrankten
Gruppe Grün

Studiengang:

BSc Medizininformatik

Autor:

Simon Adams, Mootas Abu Bakar, Jasmitha Devarasa,
Thevian Sinnappah, Emily Torresan, Gauseagan Uthayathas,
Adrian Zemp

Datum:

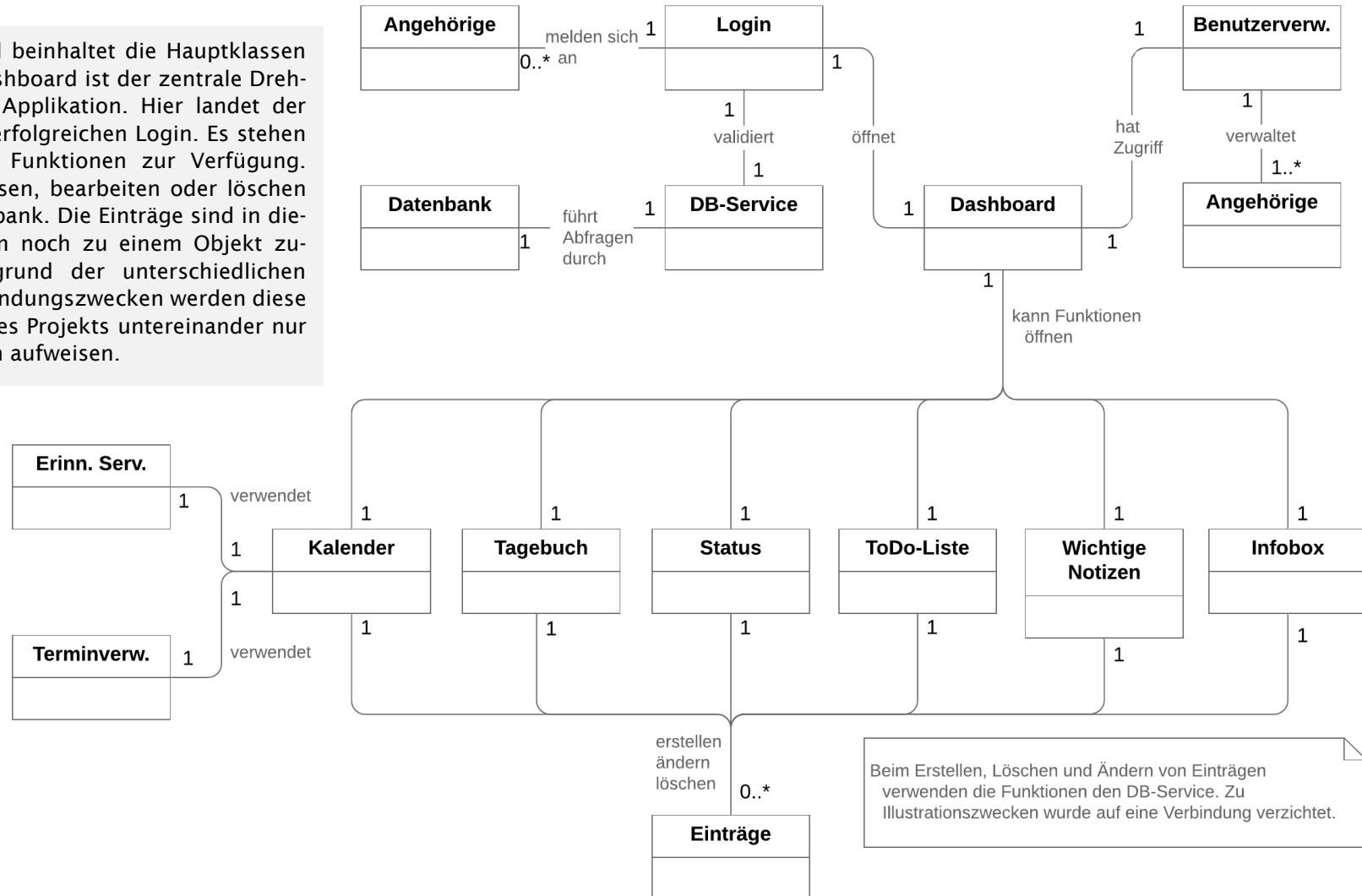
13. November 2019

1 Inhaltsverzeichnis

| | | |
|----|-----------------------------|---|
| 1 | Inhaltsverzeichnis | 2 |
| 2. | Domain Model | 3 |
| 3 | Sequenzdiagramme | 4 |
| 4 | Domain Model (aktualisiert) | 6 |

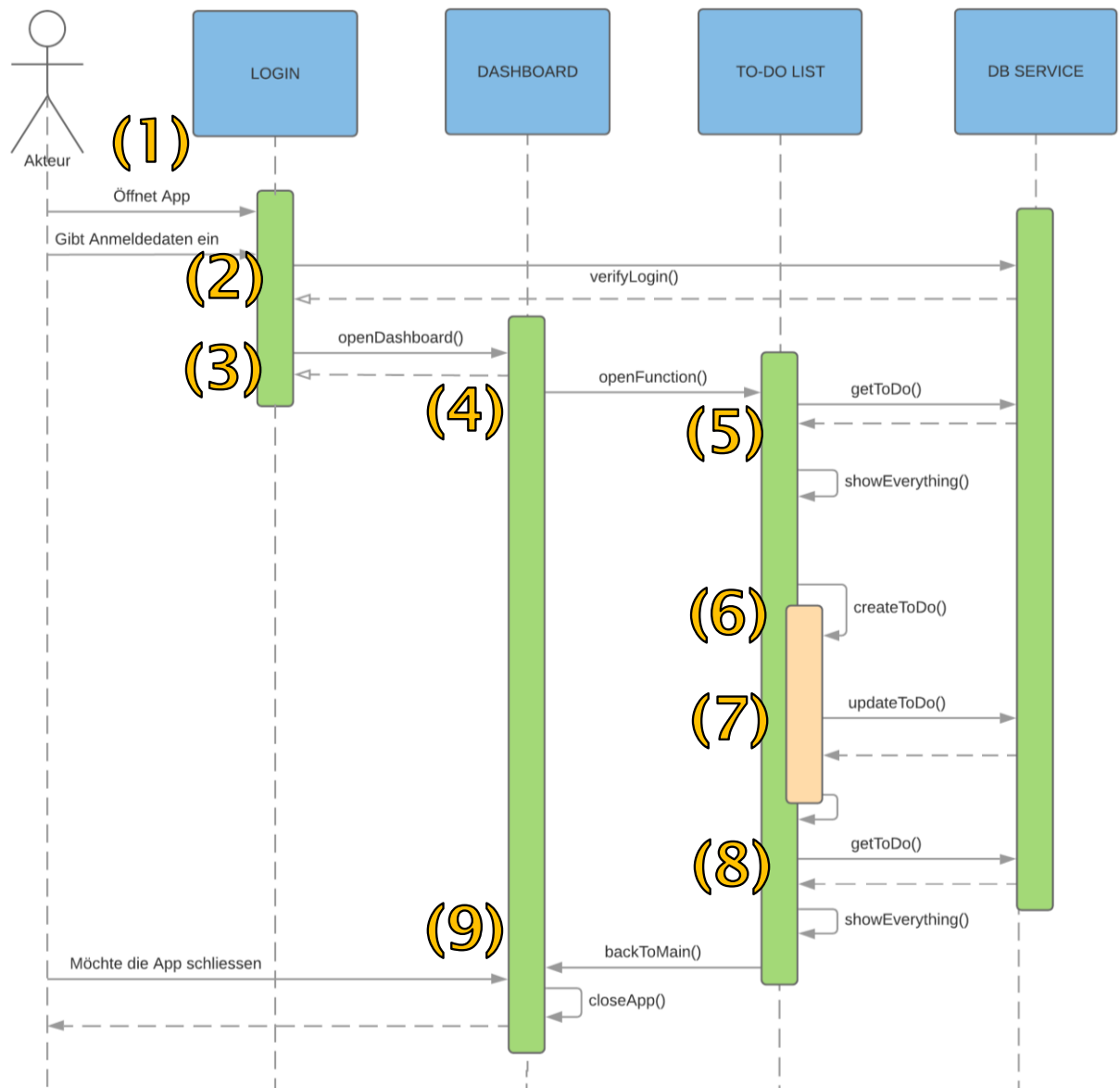
2. Domain Model

Dieses Domain Model beinhaltet die Hauptklassen der Web-App. Das Dashboard ist der zentrale Dreh- und Angelpunkt der Applikation. Hier landet der Benutzer nach einem erfolgreichen Login. Es stehen ihm unterschiedliche Funktionen zur Verfügung. Alle Funktionen erfassen, bearbeiten oder löschen Einträge in der Datenbank. Die Einträge sind in diesem ersten Diagramm noch zu einem Objekt zusammengefasst, aufgrund der unterschiedlichen Ansprüche und Verwendungszwecke werden diese im weiteren Verlauf des Projekts untereinander nur geringe Schnittflächen aufweisen.



3 Sequenzdiagramme

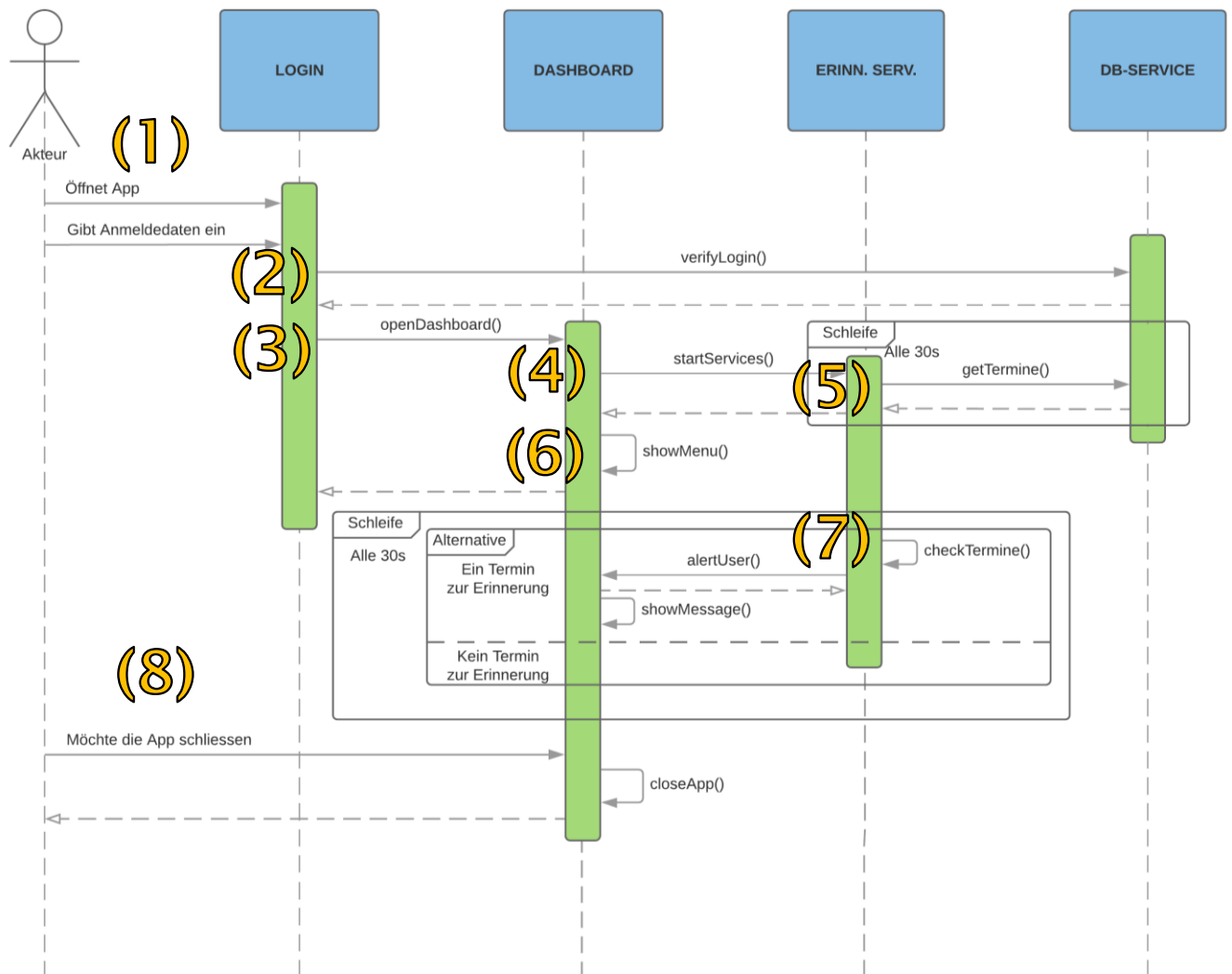
3.1 Eintrag erfassen



In diesem Diagramm wird dargestellt wie der Benutzer einen neuen Eintrag in der To-Do Liste erfasst.

1. Der Benutzer öffnet die App und gibt die die Benutzernummer und das Kennwort ein.
2. Die eingegebenen Daten werden dem DB-Service übergeben. Er vergleicht die Werte mit den in der Datenbank hinterlegten Angaben und prüft ob sie zu einem Konto passen..
3. Nachdem Benutzernummer und Kennwort verifiziert wurden wird das Dashboard gestartet.
4. Der Benutzer klickt auf das Icon der Funktion «To-Do Liste»
5. Die Einträge der «To-Do Liste» werden via DB-Service von der Datenbank abgefragt. Nachdem eine Antwort erhalten wurde werden alle Einträge angezeigt.
6. Der Benutzer drückt auf «Eintrag hinzufügen», es öffnet sich ein Eingabefenster
7. Sobald der Eintrag fertig ist wird er dem DB-Service übergeben um ihn der DB hinzuzufügen. Wenn das geklappt hat wird das Eingabefenster geschlossen.
8. Alle Einträge der «To-Do Liste» werden via DB-Service abgefragt um die Ansicht aktualisieren zu können. Wenn die neue «To-Do Liste» bereitgestellt ist wird sie angezeigt.
9. Der Benutzer hat den Eintrag erfolgreich hinzugefügt und möchte die App schließen.

3.2 Erinnerung bei einem anstehenden Termin



In diesem Diagramm wird dargestellt wie das System den User über anstehende Termine informiert.

1. Der Benutzer öffnet die App und gibt die die Benutzernummer und das Kennwort ein.
2. Die eingegebenen Daten werden dem DB-Service übergeben. Er vergleicht die Werte mit den in der Datenbank hinterlegten Angaben und prüft ob sie zu einem Konto passen..
3. Nachdem Benutzernummer und Kennwort verifiziert wurden wird das Dashboard gestartet.
4. Das Dashboard startet die nötigen Services, unter anderem den *Erinnerungs Service*.
5. Der Service fragt von nun an alle 30 Sekunden die Termine des Patienten ab.
6. Nachdem alle Services erfolgreich gestartet wurden wird das Dashboard angezeigt.
7. In einer weiteren Schleife wird alle 30 Sekunden die aktualisierte Terminliste analysiert um zu schauen ob dem User eine Erinnerung zum anstehenden Termin angezeigt werden muss. Ist dies der Fall wird dem Dashboard eine entsprechende Nachricht übergeben. Das Dashboard zeigt diese dann an. Wenn keine Termine zur Erinnerung vorhanden sind gibt es keine Aktion.
8. Nachdem der Benutzer die Erinnerung gesehen hat schliesst er die App um die Koordination des Termins mit den anderen Parteien zu besprechen.

4 Domain Model (aktualisiert)

Bereits nach zwei erstellten Sequenzdiagrammen konnte das Domain Model um zahlreiche Attribute und Funktionen erweitert werden.

Durch weitere Sequenzdiagramme oder Gruppendiskussionen könnte das Modell weiter verfeinert werden.

