

Systementwurf

Annalena, Eric, Linn, Marius, Michael, Sönke und Yasmine

19. September 2013



Inhaltsverzeichnis

1	Systemübersicht	4
1.1	Einleitung	4
1.2	Entwicklungsumgebungen	4
1.2.1	Software	4
1.2.2	Hardware	5
1.2.3	Orgware	5
2	Klassendiagramm	6
2.1	Android-App	6
2.2	Webseite	7
3	Sequenzdiagramm	8
3.1	Android-App	8
3.2	Webseite	10
4	Paketdiagramm	11
5	Komponentendiagramm	12
6	Entwicklungsdiagramm	14

1 Systemübersicht

1.1 Einleitung

Das Produkt MediDex soll die Arbeit der Antibiotic-Stewardship-Teams optimieren und als Pharmazeuthisches Assistenzsystem dienen. In diesem Systementwurf wird eine Übersicht über die zu entwickelnde Software gegeben. Dabei wird die Produktübersicht, bzw. die Funktionalität und Abhängigkeit mit verschiedenen Diagrammen dargestellt. Die dazu verwendeten Diagramme sind zwei Klassendiagramme, jeweils eins für die Android-App und eins für die Website, sowie ein Paketdiagramm, ein Komponentendiagramm und ein Entwicklungsdiagramm. Mit einem Sequenzdiagramm für die Webanwendung und einem für die Andorid-App wird jeweils ein möglicher Ablauf des Nutzers dargestellt.

1.2 Entwicklungsumgebungen

Für die Entwicklung und Bearbeitung des Produktes MediDex werden folgende Entwicklungstools und Technologien für jeden Mitarbeiter benötigt:

1.2.1 Software

- Eclipse Kepler 4.3
- L^AT_EX
- L^AT_EXPlug-in für Eclipse
- Grails Plug-in für Eclipse
- Android Development Tools (ADT) Plug-in für Eclipse 22.2
- RapidSVN 0.12.1
- Visual Paradigm for UML 10.1
- aktueller Browser
 - Firefox 24.0 final
 - Google Chrome 29.0 Final 1547.66
 - Opera 16.0 Final 1196.73

Internet Explorer (für XP) 8.0 Final

Internet Explorer 10 Final für Windows 7

1.2.2 Hardware

- ein Rechner auf dem die oben genannte Software läuft

1.2.3 Orgware

- Zugang zu Jira 5.0
- Zugang zu Confluence 4.2
- Internetverbindung

2 Klassendiagramm

2.1 Android-App

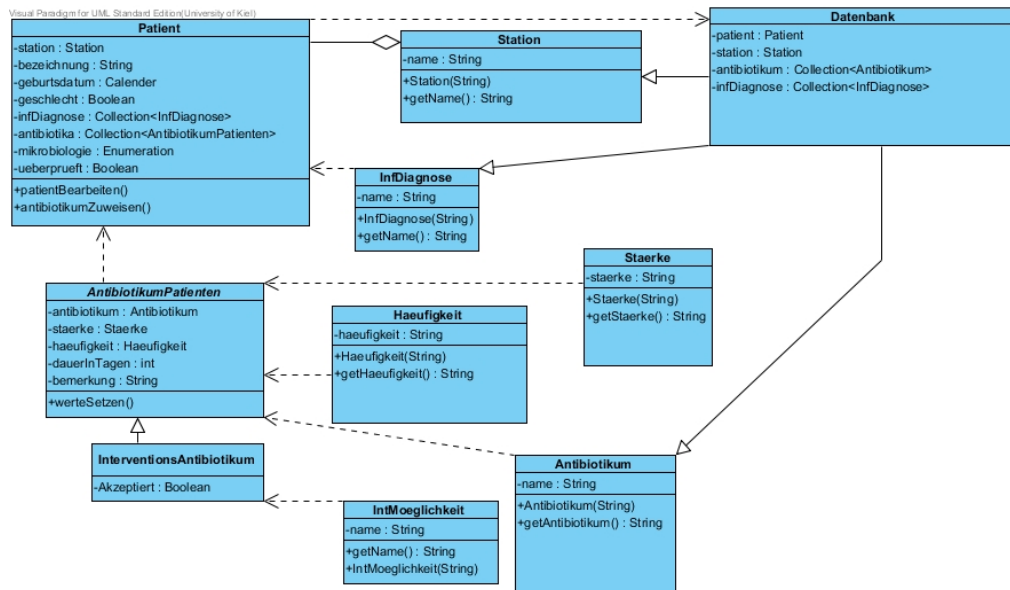


Abbildung 2.1: Klassendiagramm der Android-App

Das Model der App hat einige Klassen, die dazu dienen eine bestimmte Art von Daten zu speichern. So sind alle Häufigkeiten in einer Klasse Haeufigkeiten abgelegt. Diese werden aus der Datenbank erstellt und danach nicht verändert. Somit brauchen die Attribute keine Setter. Genauso wie Häufigkeit verhält sich InfDiagnose, Staerke, Antibiotikum, IntMoeglichkeiten und Stationen. In Patient werden für jeden Patienten seine Daten hinterlegt. Dazu gehören neben allgemeinen Daten auch die Infektionsdiagnose, die Antibiotika, die der Patient bekommt, der Befund der Mikrobiologie, sowie ob dieser Patient schon vom ABS-Team überprüft wurde. Zusätzlich kann man einen Patienten bearbeiten und ihm Antibiotika zuweisen. Dabei erben die InfDiagnose, das Antibiotikum und die Station von der Datenbank, wobei der Patient in die datenbank eingepflegt wird.

AntibiotikumPatienten sind die Antibiotika, die ein Patient verschrieben bekommen hat oder vom ABS-Team empfohlen wurden. Diese enthalten neben ihrem Namen auch die Angaben zur Dosierung und im Falle eines Interventionsantibiotikum auch, ob es vom Arzt akzeptiert wurde oder nicht. Zusätzlich kann man die Werte setzen.

2.2 Webseite

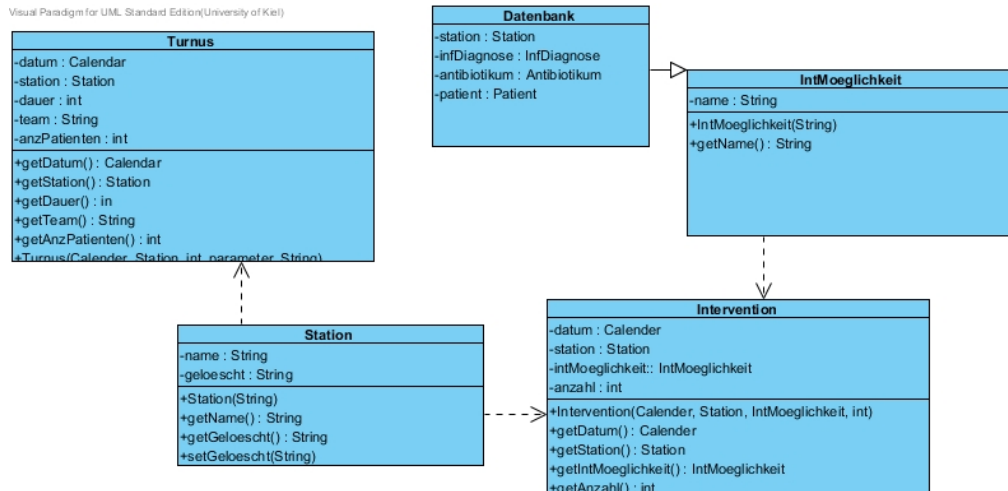


Abbildung 2.2: Klassendiagramm der Webanwendung

Um die vorhandenen Daten abzubilden braucht das Klassendiagramm der Webanwendung vier Klassen. Zum einen eine Klasse in der je eine der vorhandenen Interventionsmöglichkeiten gespeichert werden kann. Diese heißt IntMoeglichkeiten und wird aus den Daten der Datenbank generiert. Danach werden diese Daten auch nicht mehr verändert, wodurch keine Setter von Nöten sind. Dann eine in der die angelegten Stationen abgelegt werden. Diese heißt Station. In der Klasse Turnus werden Daten zu einem Datum und einer Station abgelegt. Dazu gehört die Anzahl der Patienten in der Station, die Dauer der Visite dieser Station und der Name des Teams, das die Visite durchgeführt hat. Die Klasse Interventionen speichert zu jedem Datum in jeder Station zu jeder Interventionsart die Anzahl, wie oft diese Art angewendet wurde.

3 Sequenzdiagramm

3.1 Android-App

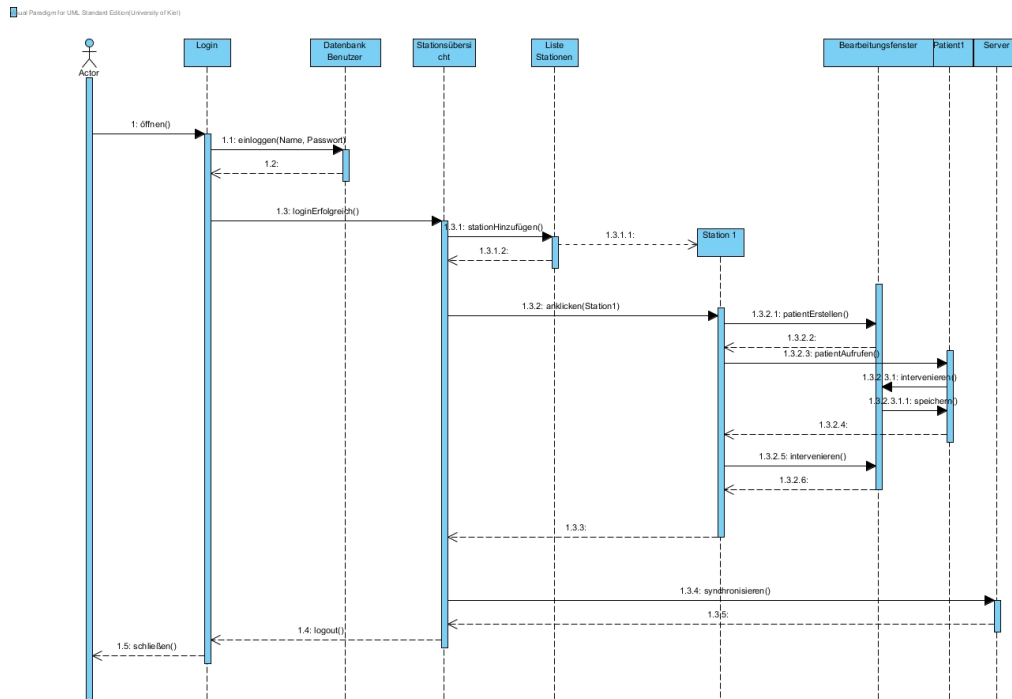


Abbildung 3.1: Sequenzdiagramm der Android-App

Um die MediDex Applikation zu nutzen, muss der Benutzer (Actor) die Applikation durch Klicken auf das Icon öffnen. Jede Funktion, die der Benutzer tätigen kann, kann nur durch ein Klick auf das richtige Symbol betätigt werden. Dadurch wird das Login-Menü geöffnet. Indem der Actor seine Zugangsdaten (Name und Passwort) einträgt und bestätigt, werden die Daten mit der Benutzer Datenbank abgeglichen. Da er alles richtig eingegeben hat, gibt die Datenbank das zurück und der Actor kommt mit dem erfolgreichen Einloggen in das Stationsübersichtsmenü. Hier kann er durch den „Station hinzufügen“-Knopf eine oder mehrere Stationen aus der Liste von Stationen in seine Übersicht hinzufügen. Der Actor hat hier „Station 1“ hinzugefügt, die auch mit einer neuen Lebensline angezeigt wird. Durchs Bestätigen kommt er wieder zurück in das Übersichtsmenü. Möchte er nun auf die Station 1 zugreifen, klickt er einmal drauf(wählt also diese Station in der Stationsübersicht). In der Station 1 kann er nun durch den

„Patient hinzufügen“-Knopf Patienten hinzufügen. Dadurch wird auch ein Bearbeitungs-fenster geöffnet. Hier fügt er alle Daten ein und speichert diese ab. Der Actor erstellt direkt „Patient 1“. Danach ist der Actor wieder im Menü von Station 1. Nun möchte der Actor gerne Patient1 intervenieren, dafür klickt er auf den Patienten und dort auf den Interventionsknopf. Dadurch kommt man wieder in das Bearbeitungs-menü, indem man die Interventionsdaten eintragen kann. Speichert man dieses kommt man wieder in die Übersicht von Patient1. Dann geht der Actor wieder zurück in die Übersicht von Station1. Von da aus kommt er durch den Zurückknopf ins Stationsübersichtsmenü. Da der Actor jetzt nur einen Patienten hatte, ist sein Turnus beendet. Durch den „Turnus beendet“-Knopf können die Daten mit dem Server synchronisiert werden. Nachdem die Arbeit fertig ist, loggt sich Actor aus und ist wieder im Login-Menü. Durch das „Verlassen“ kann er nun auch die Applikation beenden.

3.2 Webseite

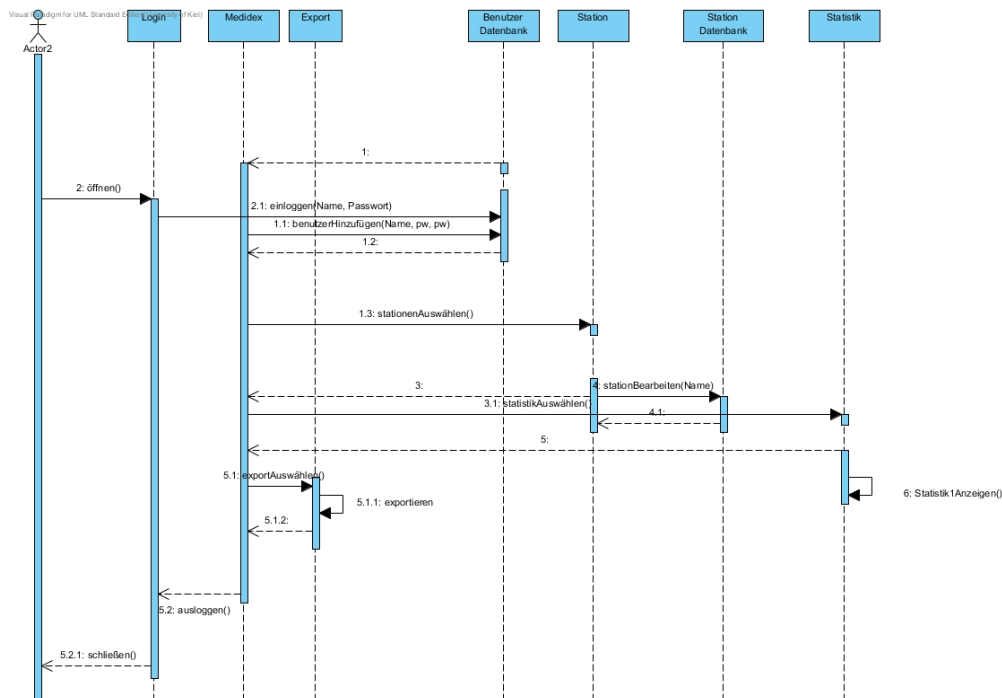


Abbildung 3.2: Sequenzdiagramm der Android-App

Nachdem der Actor seine Visite beendet hat, geht er an den PC und öffnet die Medidex-Website. Es erscheint das Login-Fenster. Sowie bei der App, erfolgt auch hier jede Funktion durch ein klicken auf das jeweilige Symbol. Hier gibt er, wie auf dem Handy, Name und Passwort ein. Durchs Bestätigen werden diese Daten mit der Benutzerdatenbank abgeglichen. Da der Actor die richtigen Daten eingetragen hat, gelangt er auf die Medidex-Seite, der Benutzer-Reiter ist geöffnet. Der Actor will hier nun einen neuen Arbeitskollegen in die Benutzer Datenbank eintragen. Das geht indem er Name, Passwort und die Wiederholung des Passwortes einträgt und auf „Benutzer hinzufügen“ klickt. Als nächste klickt Actor auf „Station“, Dadurch wird die Stationsseite aufgerufen. Hier kann er unter anderem eine Station bearbeiten in dem er den neuen Namen einträgt und bestätigt. Dies wird mit der Datenbank abgeglichen. Anschließend möchte der Actor gern die Statistik 1 betrachten. Dazu klickt er auf den Statistikreiter, um die Statistiken zu öffnen. Hier kann er sich dann direkt die Statistik 1 anzeigen lassen. Nun möchte der Actor dies als CSV-Datei exportieren und klickt auf den Exportreiter und kann nun da durch ein Klicken dies exportieren. Möchte er nun die Seite verlassen loggt er sich aus, womit er direkt ins Login-Fenster kommt. Nun kann er die Seite schließen.

4 Paketdiagramm

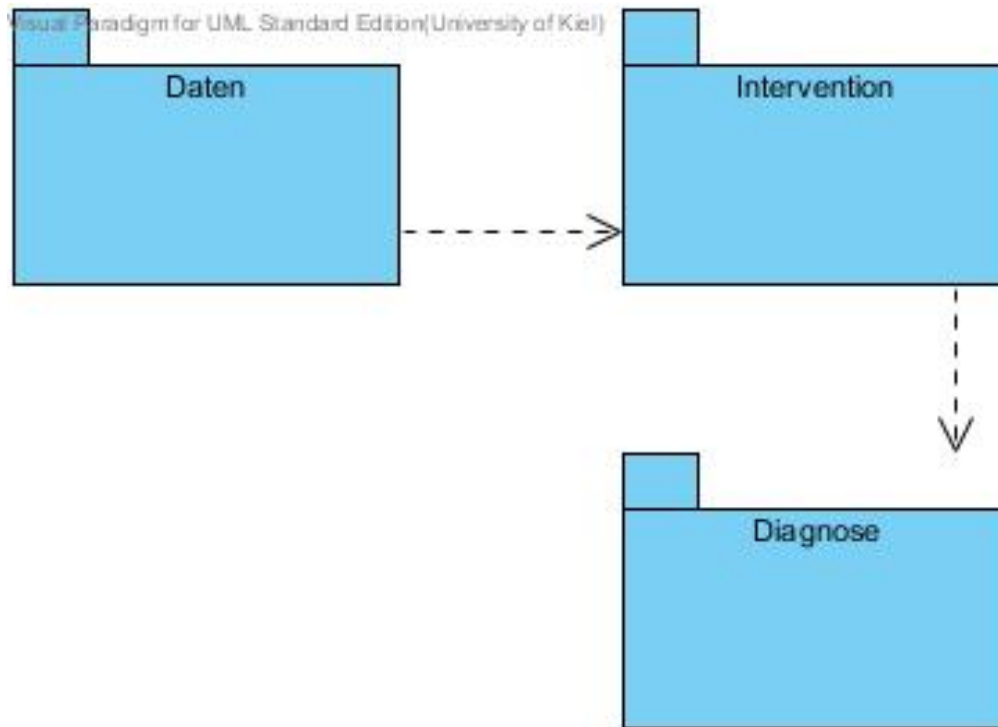


Abbildung 4.1: Paketdiagramm

In unserem Paketdiagramm haben wir drei Pakete, „Daten“, „Intervention“ und „Diagnose“. Die drei Pakete sind in der Abhängigkeit, dass Daten von Intervention und Intervention von Diagnose abhängig ist. Mit dem Paketdiagramm fassen wir die voneinander abhängigen Modellelemente unserer Klassendiagramme in jeweils ein Paket zusammen. Hierbei befinden sich die Modellelemente „Staerke“, „Häufigkeit“ und „Antibiotikum“ im Paket „Daten“. Jedes Antibiotikum hat seine eigene Häufigkeit und auch seine individuelle Stärke, die nicht verändert werden kann. Somit kann man diese zu einem Paket zusammenfassen. „AntibiotikumsPatient“, „ArztAntibiotikum“, „InterventionsAntibiotikum“ und „IntMoeglichkeit“ sind individuell auf jeden Patienten bezogen und können verändert werden. Somit werden sie im Paket „Intervention“ zusammengepackt. Die anderen Modellelemente „InfDiagnose“, „Patient“ bzw. „Turnus“ und „Station“ befinden sich im Paket „Diagnose“, da sie alle voneinander abhängig sind. Aus diesem Grund kann man diese zu einem Paket zusammenpacken.

5 Komponentendiagramm

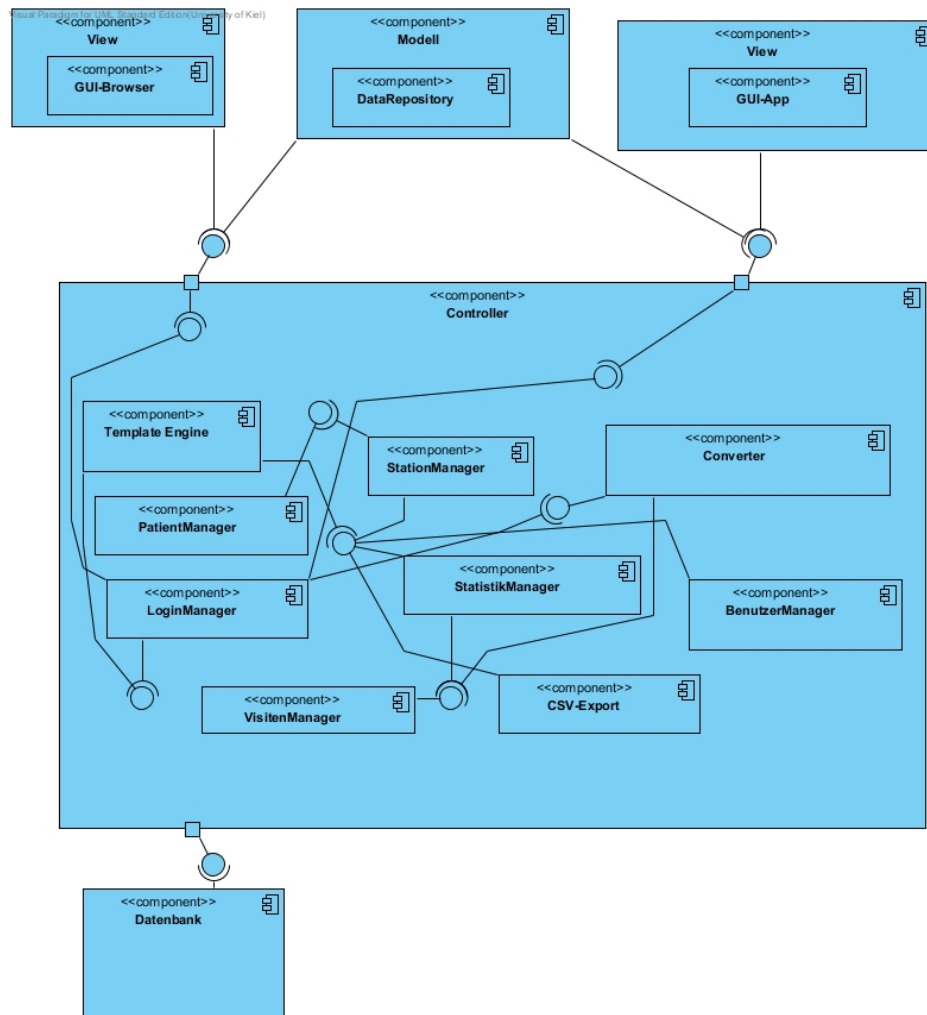


Abbildung 5.1: Komponentendiagramm

Wir benötigen für unser Projekt die Komponenten Controller, Modell und View für die Android-App und für die Webapplikation. Die Komponenten der View beinhalten auch eine Komponente für die GUI und die des Modells eine DataRepository Komponente. Diese werden benötigt um die Oberflächen anzuzeigen und die beinhalteten Daten und Funktionen bereit zu stellen. Der Controller beinhaltet folgende Komponenten: LoginManager, BenutzerManager, PatientManager, StationManager, VisitenManager, StatistikManager, CSV-Export, den Converter und das TemplateEngine. Zwischen dem LoginManager und App, sowie der Webanwendung ist ein Interface für die Verschlüsselung, das dafür verantwortlich ist, dass die Daten für den Login verschlüsselt werden. Der Controller hat einen Port, der für den Datenaustausch mit der Komponente der Datenbank über ein Interface Datentransfer zuständig ist.

6 Entwicklungsdiagramm

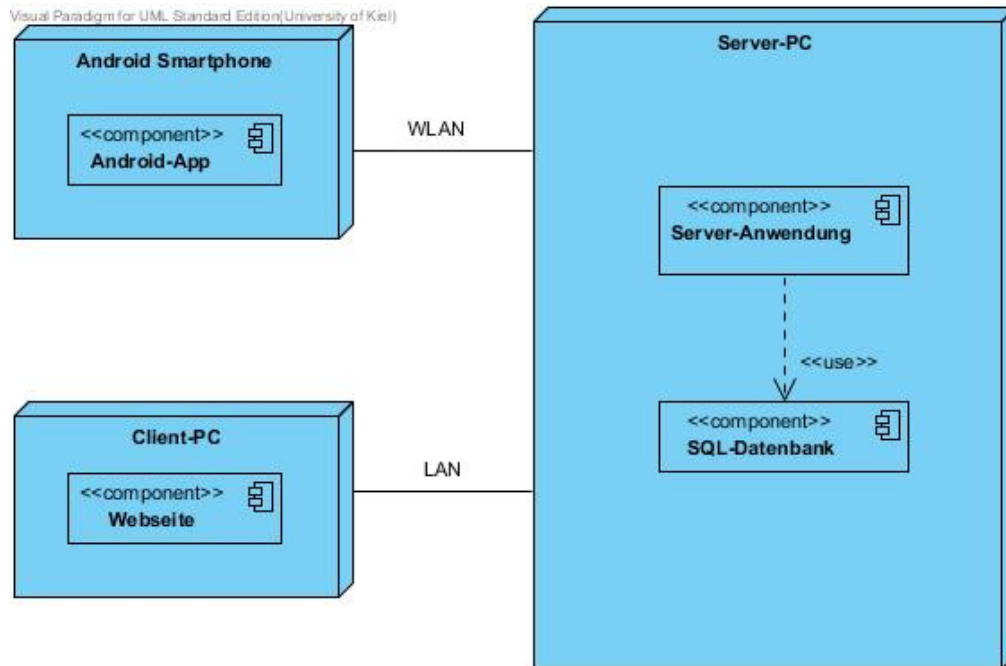


Abbildung 6.1: Entwicklungsdiagramm

MediDex besteht aus drei Knoten:

- Dem Android Smartphone
- Dem Server-PC
- Dem Client-PC

Die Komponente, die auf dem Android Smartphone laufen soll, wird eine Android-App sein. Als Komponente auf dem Client-PC wollen wir die Webseite laufen haben. Auf dem Server-PC Knoten wird es zwei Komponenten geben. Die Server-Anwendung welche Zugriff auf die dort ebenso angelegte SQL-Datenbank hat. Die Kommunikation zwischen dem Android Smartphone und dem Server-PC läuft über einen WLAN Verbindung. Die Kommunikation zwischen dem Client-PC und dem Server-PC läuft über LAN.

Abbildungsverzeichnis

2.1	Klassendiagramm der Android-App	6
2.2	Klassendiagramm der Webanwendung	7
3.1	Sequenzdiagramm der Android-App	8
3.2	Sequenzdiagramm der Android-App	10
4.1	Paketdiagramm	11
5.1	Komponentendiagramm	12
6.1	Entwicklungsdiagramm	14