



Patient Management System

Task 02 - Requirements Document
Gruppe Black
BTI7081 Software Engineering



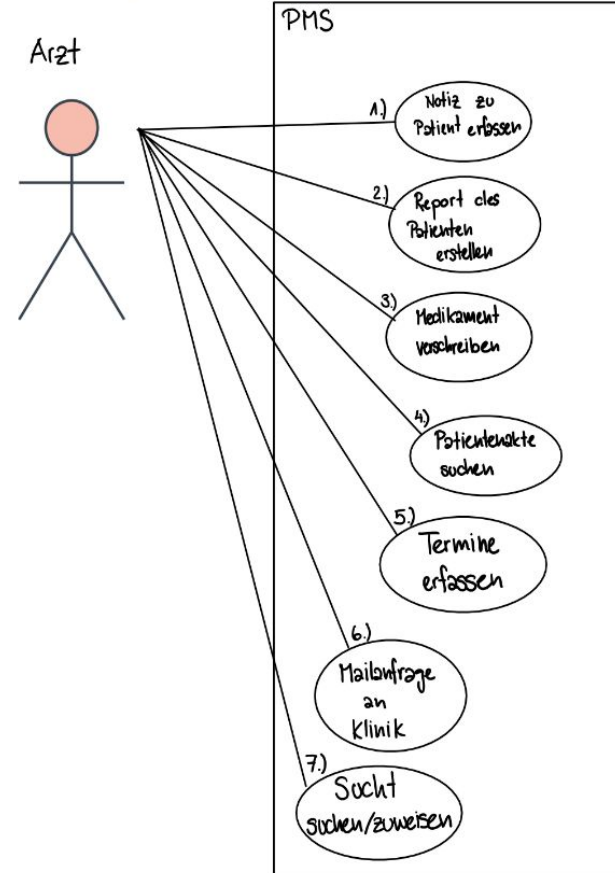


Agenda

- User Requirements
- System Architecture
- System Requirements
- System Models
- System Evolution
- Testing

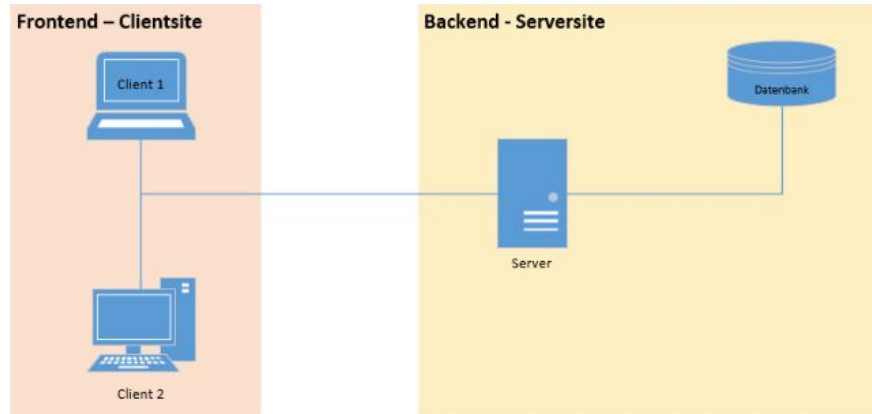
User Requirements

- Notiz zu Patient erfassen
- Report erstellen
- Medikament verschreiben
- Patientenakte suchen
- Termin erfassen
- Mailanfrage an Klinik
- Sucht suchen/zuweisen

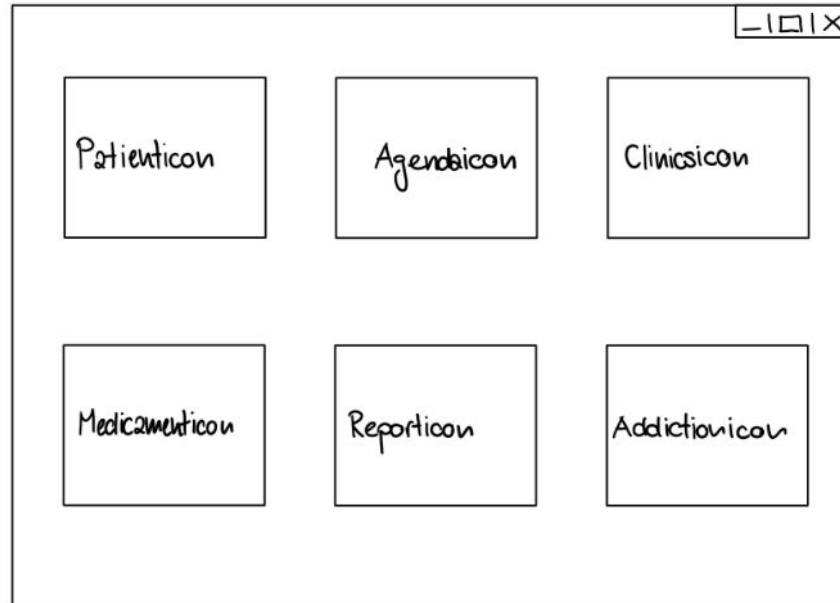


System Architecture

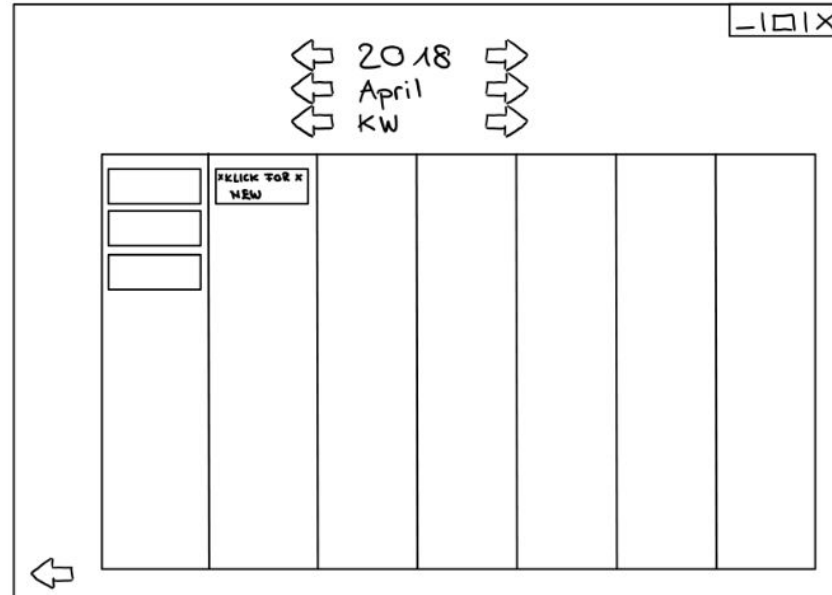
- Standard Frontend / Backend Webapplikation
- Relationale Datenbank



System Requirements - Funktional



System Requirements - Funktional



System Requirements - Funktional

A hand-drawn diagram of a functional system requirements form. The form is enclosed in a rectangular border. In the top right corner, there are three small squares representing window controls (minimize, maximize, close). The form contains the following fields and controls:

- Patient:** A text input field followed by a dropdown arrow.
- Date:** A text input field.
- Clinic:** A text input field followed by a dropdown arrow.
- Start:** A text input field.
- End:** A text input field.
- Comments:** A large rectangular text area for notes.
- Navigation:** A left-pointing arrow icon in the bottom left corner.
- Buttons:** Two buttons labeled "Cancel" and "Save" in the bottom right corner.

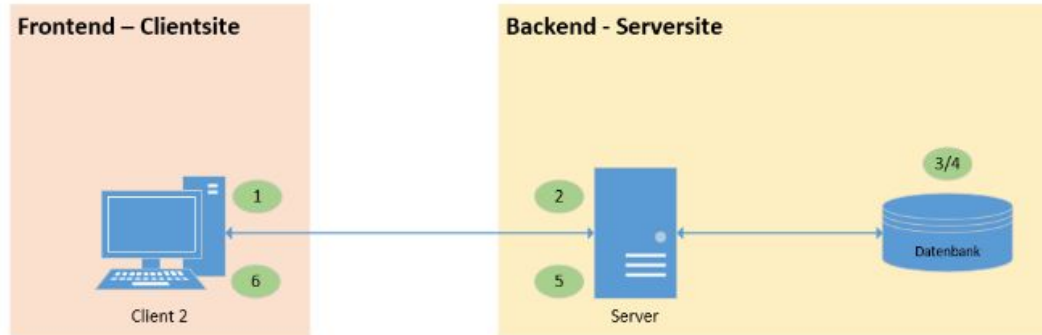
System Requirements - Nicht-funktional

- Usability
 - Intuitive und angenehme Benutzeroberfläche
- Performance
 - Multiuser Fähigkeit
 - Zugriffe sind effizient
- Sicherheit
 - Zugriffe sind autorisiert
 - Datenschutz ist gewährleistet



System Models

- Client-Server Kommunikation
- Datenbank für persistente Datenspeicherung





System Evolution

- Plattformunabhängige Webapplikation
- Moderne offene Standards
 - HTML5
 - CSS3
- Open Source
 - Kein Vendor Lock-in
 - Portierbarkeit
 - Verbreiteter Stack (bspw. nginx + OpenJDK + Tomcat + MariaDB)



Testing

- Unit-Tests
 - Klassen und Methode
 - Continuous Testing (bspw. bei Pull Requests)
- Manuelle Tests
 - GUI und Logik
- Integrationstests
 - Acceptance Tests vor Sprint Closure
- Lasttests
 - Performance Tests (bspw. mit PhantomJS & Siege)

Danke
Merci
Grazie

Gruppe Black
BTI7081 Software Engineering

