

# Aufgabenstellung: Architekturdokumentation

**Aufgabe:** Erstellen Sie auf Grundlage der Ihnen vorliegenden Informationen zum Fallbeispiel "WohnGutWissen" eine Systemarchitektur. Dokumentieren Sie diese auf Grundlage des arc42-Templates.

## Randbedingungen

Dazu gelten folgende Randbedingungen:

- Abzugeben ist die Ausarbeitung in digitaler Form (PDF) bis zum **28.02.2021 AoE** (Anywhere on Earth/UTC+12).
- **Jeder Studierende** hat individuell eine Ausarbeitung abzugeben.
- Verwenden Sie als Ausgangspunkt Ihrer Ausarbeitung das arc42-Template, welches Sie unter <https://arc42.org/download> finden. Bei der konkreten Wahl der Technologie (Word, Markdown, AsciiDoc, Latex, ...) sind sie frei.
- Es ist zu erwarten, dass die Ausarbeitung einen Umfang von ca. 20 Seiten haben wird (inkl. Deckblatt, Inhaltverzeichnis, etc.) (geschätzt für A4, 12pt, 3cm Rand umlaufend, einfacher Zeilenabstand)
- Verwenden Sie für Diagramme im Allgemeinen die UML-Notation (v2.5) oder geben Sie eine entsprechende Legende für das Diagramm an.

## Erwarteter Inhalt

Hinweis: Sie können, werden und müssen nicht alle Aspekte des arc42-Templates verwenden und mit Inhalt füllen. Dies ist NICHT Ziel des Templates und nicht Ziel der Dokumentation.

Folgende Punkte sollten Sie jedoch in die Dokumentation aufzunehmen:

- Darstellung der zentralen/architekturelevanten Anforderungen
- Aufstellung der zentralen Stakeholder (inkl. deren Erwartungshaltung)
  - Stakeholder in Bezug auf das System unter beachtung aller Lebensphasen des Systems, insbesondere der Phasen Entwicklung, Nutzung, Betrieb und Weiterentwicklung
- Darstellung des fachlichen Systemkontextes (Ebene 0)
- Darstellung von mind. zentralen 3 Qualitätszielen/Anforderungen
- Fachliche Zerlegung des Systems auf Zerlegungsebene 1 (Whitebox-Darstellung)

- Fachliche oder technische Zerlegung auf Ebene 2 (Blackbox-Darstellung)
- Für die Darstellungen der Zerlegungen sind jeweils mindestens folgende Punkte anzugeben:
  - graphische Darstellung
  - Begründung der Zerlegung
  - Beschreibung der Bausteine
  - Beschreibung der Schnittstellen
- Darstellung des fachlichen Kernmodells (Domain model)
- Darstellung von zwei zentralen Abläufen/Geschäftsprozessen
- Darstellung des Deployments
- Glossar

# Fallbeispiel: Verwaltungsprogramm für wohnungswirtschaftliche Unternehmen

## Entwicklungsziel "WohnGutWissen"

Für das wohnungswirtschaftliche Unternehmen "WohnGut" soll eine internes System realisiert werden, in welchem der Bestand aller Mietobjekte, die dazugehörigen Mietverträge sowie die zugehörigen Mieter verwaltet werden können. Der Projekttitle lautet "WohnGutWissen" (WGW).

Im Bestand der WohnGut befinden sich ca. 1.000 Wohneinheiten und ca. 250 Büro- und Gewerbeeinheiten, welche an Privatpersonen und Unternehmen (jur. Personen) vermietet werden.

Die Anwendung "WohnGutWissen" soll durch die Mitarbeiter des Unternehmens verwendet werden, um Mieter, Mietverträge, Mietobjekte erfassen und verwalten zu können. Weiterhin soll WohnGutWissen jedem Mieter die Überblicksinformationen über seine Mietverträge geben können. Dabei sollen alle Mietverträge berücksichtigt werden, unabhängig davon, ob die Mietverträge beendet, laufend oder unterzeichnet aber noch nicht begonnen sind.

Die Anwendung soll sowohl für die internen Mitarbeiter als auch die Mieter mittels Browser aufrufbar sein.

In einer späteren Ausbaustufe soll WGW unter anderem um Funktionen erweitert werden, die es dem Mieter auch erlaubt, bestehende Mietverträge zu kündigen oder Serviceanfragen für bestehende Mietobjekte auszulösen (z.B. Hausmeisterdienst bzgl. Defekt informieren). Das zu erstellende System muss diese geplanten Erweiterungen strukturell berücksichtigen.

WohnGutWissen soll zum zentralen System innerhalb der WohnGut ausgebaut und über Jahre hinweg eingesetzt werden.

# Anforderungen

## Funktionale Anforderungen:

### Erfassen von Mietobjekten

Im WGW sollen *alle* Mietobjekte der WohnGut erfasst werden können. Neben Adresse des Mietobjektes sollen Art des Mietobjektes (Wohnung, Büroeinheit, Ladenlokal), Grundfläche, Baujahr, Zeitpunkt der letzten Sanierung hinterlegt werden können.

### Erfassung von neuen Mietverträgen

Im WGW sollen alle Mietverträge der WohnGut erfasst und verwaltet werden können. Mietverträge werden immer genau einem Mietobjekt und einem Hauptmieter zugeordnet. Neben dem Hauptmieter können bis zu drei Nebenmieter in den Mietvertrag eingetragen werden. Der Mietvertrag hat immer einen Vertragsbeginn. Das Vertragsende ist im Allgemeinen offen und wird erst durch Kündigung bzw. anderweitige Beendigung des Vertrages bekannt.

### Mieterstammdaten

Die Mietobjekte der WohnGut werden sowohl an Privatpersonen als auch an Unternehmen vermietet. WGW muss beide Arten von Mietern unterstützen. Je Mieter, also Privatperson oder Unternehmen, wird eine EMailadresse erfasst, welche auch zur Identifizierung bei der Anmeldung am System verwendet wird.

### Kündigen von Mietverträgen

Den internen Mitarbeitern ist es möglich, die Kündigung oder Beendigung eines Mietvertrages unter Angabe eines Enddatums und ggf. eines Kommentars zu erfassen.

### Integration zu anderen Systemen

Läuft ein Mietvertrag aus, so muss WGW eine Benachrichtigung an das Aufgabenverwaltungssystem "WohnGutTuDas" senden. Dazu stellt WohnGutTuDas eine nachrichten-basierte Schnittstelle zur Verfügung.

### Export aktueller Stammdatensatz für MieterInformationsPortal

Für das externe System "MieterInformationsPortal" soll ein Datenexport erstellt werden. Die Exportfunktion soll per HTTP-Schnittstelle zur Verfügung stehen und eine JSON-Datenstruktur ausgeben. In dieser Datenstruktur sind

- alle Mieter
- alle Mietobjekte
- alle Mietverträge

auszugeben.

## **Bekannte Qualitätsanforderungen**

### **Verfügbarkeit**

Das WGW-System soll für die Mieter und die Mitarbeiter der WohnGut zu 99,5% der Zeit zugreifbar und nutzbar sein.

### **Vertraulichkeit**

Jeder Mieter darf nur die Informationen sehen, welche ihn persönlich betreffen.

### **Wiederherstellung**

Das WGW muss nach einem Systemausfall innerhalb von 24h wieder herstellbar sein.

### **Reaktionsgeschwindigkeit Frontend**

Das WGW-Frontend muss für den Anwender in 95% der Fälle innerhalb von 1 Sekunde nach der Anfrage die angefragten Informationen darstellen, in 99,5% der Fälle innerhalb von 5 Sekunden.

### **Erweiterbarkeit**

Das WGW System muss so aufgebaut sein, das sehr einfach für externe Systeme ein zusätzlicher technischer Zugriff auf die im WGW hinterlegten Informationen und Funktionen realisiert werden kann.

### **Effizient**

Das WGW-System muss auf "normaler" Serverhardware innerhalb der Räumlichkeiten der WohnGut durch die WohnGut-EDV-Abteilung betrieben werden können.

### **Anpassbarkeit**

Der Aufbau des WGW-Systems muss es ermöglichen, dass zukünftig alternative Nutzerschnittstellen implementiert werden können.

### **Benutzbarkeit**

Das Webfrontend für die Mieter muss für einen durchschnittlich technisch interessierten Menschen einfach nutzbar sein.

### **Mehrsprachkeit**

Die WGW-Nutzeroberfläche muss in deutscher Sprache gehalten sein. Die UI muss jedoch darauf vorbereitet sein später auch in anderen Sprachen angeboten werden zu können.

## **Restriktionen:**

### **Webbasierte Anwendung**

Auf das zu entwickelnde System soll über eine Webanwendung durch die Mitarbeiter des wohnungswirtschaftlichen Unternehmens und Mieter zugegriffen werden.

### **Datenbankmanagementsystem**

Die Daten müssen in ein Microsoft MS SQL Datenbank gespeichert werden, da die WohnGut-EDV-Abteilung bereits einen solchen Server betreibt.

### **Umsetzungssprache/Laufzeitumgebung**

Das WGW Backend ist in der Programmiersprache Java umzusetzen. Der Ausführungsserver wird auf Basis von Linux betrieben. Als Laufzeitumgebung darf nur ein kommerziell nutzbares und gepflegtes JDK zum Einsatz kommen.

## **Zentrale Geschäftsprozesse**

Folgende Geschäftsprozesse werden unter anderem als zentral angesehen:

- Neuer Mieter (bisher dem System nicht bekannte natürliche Person) schließt einen Mietvertrag ab
- Mietvertrag endet
- Export von Stammdaten für MieterInformationsPortal