

**POS PHP-Kleinprojekt Planungsdokumentation**

**E-Banking App**

Xia, Meilinger

erstellt von

Julian Meilinger Verantwortlich für Frontend: HTML, CSS

Liuming Xia Verantwortlich für Backend: PHP

Imst, 1.11.2021

**Inhaltsverzeichnis**

[1 Projektmanagement 4](#_Toc1384008)

[1.1 Metainformationen 4](#_Toc1384009)

[1.1.1 Team 4](#_Toc1384010)

[1.1.2 Partner (fiktiv) 4](#_Toc1384011)

[1.2 Vorerhebungen 4](#_Toc1384012)

[1.2.1 Projektzieleplan 4](#_Toc1384013)

[1.2.2 Projektumfeld 4](#_Toc1384014)

[1.2.3 Risikoanalyse 4](#_Toc1384015)

[1.3 Pflichtenheft 4](#_Toc1384016)

[1.3.1 Zielbestimmung 4](#_Toc1384017)

[1.3.2 Produkteinsatz und Umgebung 4](#_Toc1384018)

[1.3.3 Funktionalitäten (FURPS) 4](#_Toc1384019)

[1.3.4 Testszenarien und Testfälle 5](#_Toc1384020)

[1.4 PLANUNG 5](#_Toc1384021)

[1.4.1 Projektstrukturplan 5](#_Toc1384022)

[1.4.2 Meilensteine 5](#_Toc1384023)

[1.4.3 Gant-Chart 5](#_Toc1384024)

[1.4.4 Abnahmekriterien 5](#_Toc1384025)

[1.4.5 Pläne zur Evaluierung 5](#_Toc1384026)

[1.4.6 Ergänzungen und zu klärende Punkte 5](#_Toc1384027)

[2 Vorstellung des Produktes 5](#_Toc1384028)

[3 Eingesetzte Technologien 5](#_Toc1384029)

[4 Anforderungsanalyse 5](#_Toc1384030)

[4.1 Einleitung 6](#_Toc1384031)

[4.1.1 Zweck des Systems 6](#_Toc1384032)

[4.1.2 Scope des Systems 6](#_Toc1384033)

[4.1.3 Ziele und Erfolgskriterien für das Projekt 6](#_Toc1384034)

[4.1.4 Definitionen, Akronyme, Abkürzungen 6](#_Toc1384035)

[4.2 Anforderungen 6](#_Toc1384036)

[4.2.1 Überblick 6](#_Toc1384037)

[4.2.2 Funktionale Anforderungen 6](#_Toc1384038)

[4.2.3 Nichtfunktionale Anforderungen 6](#_Toc1384039)

[4.3 System-Modellierung 7](#_Toc1384040)

[4.3.1 Szenarios 7](#_Toc1384041)

[4.3.2 Use-Case-Model 8](#_Toc1384042)

[4.3.3 User interface mockups 8](#_Toc1384043)

[4.3.4 Systementwurf 8](#_Toc1384044)

[5 Glossar 10](#_Toc1384045)

# Projektmanagement

## Metainformationen

### Team

### Partner (fiktiv)

## Vorerhebungen

### Projektzieleplan

Projektziele-Hierarchie - SMART

### Projektumfeld

* Projektumfeldanalyse (PUMA)
* Identifizierung der Stakeholder
* Charakterisierung der Stakeholder
* Maßnahmen
* Grafische Darstellung des Umfeldes

### Risikoanalyse

* Incl. Risikomatrix

## Pflichtenheft

### Zweck des Systems

Beschreibung des Einsatzzweckes des Systems.

### Zielbestimmung

* Projektbeschreibung
* IST-Zustand (ggf. fiktiv)
* SOLL-Zustand
* NICHT-Ziele (Abgrenzungskriterien)

### Produkteinsatz und Umgebung

* Anwendungsgebiet
* Zielgruppen
* Betriebsbedingungen
* Hard-/Softwareumgebung

### Funktionale Anforderungen

*Functional requirements describe the high-level functionality of the system. This section list all functional requirements and additionally presents the dependencies between them.*

### Nichtfunktionale Anforderungen

*Nonfunctional requirements describe user-level requirements that are not directly related to functionality. This includes usability, reliability, performance, supportability, implementation, interface, operational, packaging, and legal requirements. The section list all these non-functional requirements and additionally presents the dependencies between them.*

### Definitionen, Akronyme, Abkürzungen

### Testszenarien und Testfälle

* Beschreibung der Testmethodik
* Testfall 1
* Testfall 2
* …

## Projektplanung

### Projektstrukturplan

### Meilensteine

### Gant-Chart

### Abnahmekriterien

### Pläne zur Evaluierung

### Ergänzungen und zu klärende Punkte

# Vorstellung des Produktes

Vorstellung des fertigen Produktes anhand von Screenshots, Bildern, Erklärungen.

# Eingesetzte Technologien

* Kurzbeschreibung aller Technologien, die verwendet wurden.
* Technologien die aus dem Unterricht bekannt sind, nur nennen und deren  Einsatzzweck im Projekt beschreiben, nicht die Technologien selbst.
* Technologien die aus dem Unterricht nicht bekannt sind, im Detail beschreiben incl. deren Einsatz im Projekt
* Fokus aus eingesetzten Frameworks

# System-Modellierung

*The System models include scenarios, use cases, and the analysis object model for the system. This section should contain the complete functional specification, including mock-ups, paper-based prototypes or storyboards illustrating the user interface of the system and navigational paths representing the sequence of screens.*

## Szenarios

Ein Szenario beschreibt einen Anwendungsfall eines Akteurs im Detail. Zu jedem Anwendungsfall im Use-Case-Diagramm muss so ein Szenario beschrieben werden.

|  |  |  |
| --- | --- | --- |
| **Scenario name** | *Name* | |
| **Participating actors** | **Initiated by** *actor* | |
| **Flow of events** | **1)** *User step description* | |
|  |  | **2)** *System step description* |
|  | **3)** *User step description* | |
|  |  | **4)** *System step description* |
|  | … | |
| **Entry conditions** | *Description* | |
| **Exit conditions** | *Description* | |
| **Invariants** | *Description* | |
| **Quality requirements** | *Description* | |

* UseCases auf Basis von Benutzerzielen identifizieren:
  + Benutzer eines Systems identifizieren
  + Benutzerziele identifizieren (Interviews)
  + Use-Case-Liste pro Benutzer definieren
* UseCases auf Basis von Ereignissen identifizieren:
  + Externes Event triggert einen Prozess
  + zeitliches Event triggert einen Prozess (Zeitpunkt wird erreicht)
  + State-Event (Zustandsänderung im System triggert einen Prozess)
* Werkzeuge:
  + USE-Case-Beschreibungen (textuell, tabellarisch)
  + USE-Case-Diagramm
  + Aktivitätsdiagramm für den Use-Case (Interaktion zwischen Akteur und System abbilden)
  + System-Sequenzdiagramm (Spezialfall eines Sequenzdiagramms: Nur 1 Akteur und 1 Objekt, das Objekt ist das komplette System, es geht um die Input/Output Requirements, die abzubilden sind)

## Use-Case-Model

Use-Case-Diagramme auf Basis der Szenarios erstellen

## User interface mockups

Zu den Use-Cases müssen passende Mockups / Wireframes etnworfen werden, die genau zeigen, wie die Benutzeroberfläche aussehen wird.

* Design des UIs über Mockups / Wireframes
* Dialoge, Dialogsteuerung, Ergonomie, Gestaltung, Eingabeüberprüfungen, Fehlermeldungen

Tools: Adobe XD oder Balsamiq

## Systementwurf

### Architektur

Beschreibung der Architektur-Design im Großen (grobe Systemstruktur die im Laufe des OO-Design laufend verbessert wird):

* die statische Zerlegung des Systems in seine physischen Bestandteile (Komponenten, Komponentendiagramm, Schichten),
* die (textuelle) Beschreibung des dynamischen Zusammenwirkens aller Komponenten und
* die (textuelle) Beschreibung der Strategie für die Architektur, d. h. wie die Architektur in Statik und Dynamik funktionieren soll.
* Verwendung von Referenzarchitekturen bzw. Architekturmustern (als Schablonen, z.B. MVC., Schichten etc.)
  + MVC
  + Schichten
  + Service-Orientiert etc.

### Datenhaltungskonzept

* Design der Datenbank (ER-Modell)
* Design des Zugriffs auf diese Daten (Datenhaltungskonzept)
* Caching, Transaktionen

### Analysis Object Model

Es sind **UML-Diagramme** zu erstellen für:

* Klassendiagramme für Domain-Klassen für "Dinge" (Rollen, Einheiten, Geräte, Events etc.) identifizieren, um die es im Projekt geht
* Klassendiagramme für weitere Klassen im System (incl. detaillierter Darstellung und Verwendung von Vererbungshierarchichen, abstrakten Klassen, Interfaces)
* Sequenzdiagramme oder Aktivitätsdiagramme zur Darstellung von Abläufen

Klassen können dabei sein:

* UI-Klassen / View-Klassen
* Data-Access-Klassen (DAO, ActiveRecord)
* Entity-Klassen (Domain-Klassen)
* Controller-Klassen
* Business-Logik-Klassen

Verwendung von CRC-Cards (Class, Responsibilities, Collaboration) für die Klassen

* um Verantwortlichkeiten und Zusammenarbeit zwischen Klassen zu definieren und
* um auf den Entwurf der Geschäftslogik zu fokussieren

Optimierung des Entwurfs (Modularisierung, Erweiterbarkeit, Lesbarkeit):

* Kopplung optimieren
* Kohäsion optimieren
* SOLID
* Entwurfsmuster einsetzen (z.B. Singleton, DAO, ActiveRecord, Strategy etc.)

Wichtige UML-Diagrammtypen:

* Komponentendiagramm: <https://www.youtube.com/watch?v=aYVa1CC1rJo>
* Klassendiagramme:  
  <https://www.youtube.com/watch?v=ug2EeU13yDA>
* Sequenzdiagramme  
  <https://www.youtube.com/watch?v=pCK6prSq8aw>
* Aktivitätsdiagramme  
  <https://www.youtube.com/watch?v=aM1Fu9qCocI>
* Use-Case-Diagramm  
  <https://www.youtube.com/watch?v=dYfr5Opkw38>

Viele weitere Hinweise zum Prozess der Softwareentwicklung / Anforderungsanalyse etc.:

**Gute Videos in Deutsch: Playlist** [**Softwareengineering - Wie entwickelt man Software**](https://www.youtube.com/watch?v=r4u_6OPHiMo&list=PLNmsVeXQZj7qNMn6zimfu4JPUklG-4Uu4)**:**

<https://www.youtube.com/playlist?list=PLNmsVeXQZj7qNMn6zimfu4JPUklG-4Uu4>

# Glossar

*A glossary of important terms used in the project and in the system model ensures consistency in the specification and a common understanding of terms used by the client.*