

LOGO

Projektpartner

**DIPLOMARBEIT**

**THEMA der DIPLOMARBEIT (Haupttitel)**

(Untertitel)

eingereicht von

Max Mustermann Verantwortlich für IT: HTML CSS, BWL: Kaufvertrag

Susanne Sorglos Verantwortlich für IT: HTML CSS, BWL: Kaufvertrag

Otto Normalverbraucher Verantwortlich für IT: HTML CSS, BWL: Kaufvertrag

Felicitas Glücklich Verantwortlich für IT: HTML CSS, BWL: Kaufvertrag

eingereicht bei (Betreuungslehrer/innen)

BHAK/HAS Imst

Ort, Datum

**Eidesstattliche Erklärung**

Ich erkläre an Eides statt, dass ich die vorliegende Diplomarbeit selbst verfasst und keine anderen als die angeführten Behelfe verwendet habe. Alle Stellen, die wörtlich oder inhaltlich den angegebenen Quellen entnommen wurden, sind als solche kenntlich gemacht.

Ich bin damit einverstanden, dass meine Arbeit öffentlich zugänglich gemacht wird.

Ort, Datum

|  |  |  |
| --- | --- | --- |
| Max Mustermann |  | Susanne Sorglos |
|  |  |  |
|  |  |  |
|  |  |  |
| Otto Normalverbraucher |  | Felicitas Glücklich |

**Abnahmeerklärung**

Hiermit bestätigt der Auftraggeber ………………………... , dass das übergebene Produkt dieser Diplomarbeit den dokumentierten Vorgaben entspricht. Des Weiteren verzichtet der Auftraggeber auf unentgeltliche Wartung und Weiterentwicklung des Produktes durch die Projektmitglieder bzw. die Schule.

Ort, Datum

|  |  |  |  |
| --- | --- | --- | --- |
| Auftraggeber |  |  |  |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |

**Vorwort**

z. B. Hinweise, wie das bearbeitete Thema gefunden wurde oder Dank für die Betreuung (Kooperationspartner/in, Betreuer/innen, Sponsoren) etc.

**Abstract (Deutsch)**

(ca. ½ bis max. 2 Seiten)

Kurzbeschreibung von Aufgabenstellung und Problemlösung.

**Abstract (Englisch)**

(ca. ½ bis max. 2 Seiten)

**Inhaltsverzeichnis**

[1 Projektmanagement 11](#_Toc467482586)

[1.1 Metainformationen 11](#_Toc467482587)

[1.1.1 Team 11](#_Toc467482588)

[1.1.2 Betreuer 11](#_Toc467482589)

[1.1.3 Partner 11](#_Toc467482590)

[1.1.4 Ansprechpartner 11](#_Toc467482591)

[1.2 Vorerhebungen 11](#_Toc467482592)

[1.2.1 Projektzieleplan 11](#_Toc467482593)

[1.2.2 Projektumfeld 11](#_Toc467482594)

[1.2.3 Risikoanalyse 11](#_Toc467482595)

[1.3 Pflichtenheft 11](#_Toc467482596)

[1.3.1 Zielbestimmung 11](#_Toc467482597)

[1.3.2 Produkteinsatz und Umgebung 11](#_Toc467482598)

[1.3.3 Funktionalitäten (FURPS) 12](#_Toc467482599)

[1.3.4 Testszenarien und Testfälle 12](#_Toc467482600)

[1.3.5 Liefervereinbarung 12](#_Toc467482601)

[1.4 PLANUNG 12](#_Toc467482602)

[1.4.1 Projektstrukturplan 12](#_Toc467482603)

[1.4.2 Meilensteine 12](#_Toc467482604)

[1.4.3 Gant-Chart 12](#_Toc467482605)

[1.4.4 Abnahmekriterien 12](#_Toc467482606)

[1.4.5 Pläne zur Evaluierung 12](#_Toc467482607)

[1.4.6 Ergänzungen und zu klärende Punkte 12](#_Toc467482608)

[2 Vorstellung des Produktes 12](#_Toc467482609)

[3 Eingesetzte Technologien 12](#_Toc467482610)

[4 Problemanalyse 13](#_Toc467482611)

[4.1 USE-Case-Analyse 13](#_Toc467482612)

[4.2 Domain-Class-Modelling 13](#_Toc467482613)

[4.3 User-Interface-Design 13](#_Toc467482614)

[5 Systementwurf 14](#_Toc467482615)

[5.1 Architektur 14](#_Toc467482616)

[5.2 Datenhaltungskonzept 14](#_Toc467482617)

[5.3 Ein- Ausgabekonzept 14](#_Toc467482618)

[5.4 Konzept der Ausnahmebehandlung 14](#_Toc467482619)

[5.5 Sicherheitskonzept 15](#_Toc467482620)

[5.6 Design der Testumgebung 15](#_Toc467482621)

[5.7 Design der Ausführungsumgebung (Environment) 15](#_Toc467482622)

[5.8 Detailentwurf 15](#_Toc467482623)

[6 Implementierung 16](#_Toc467482624)

[7 Deployment 16](#_Toc467482625)

[8 Tests 16](#_Toc467482626)

[8.1 Systemtests 16](#_Toc467482627)

[8.2 Akzeptanztests 17](#_Toc467482628)

[9 Projektevaluation 17](#_Toc467482629)

[10 Benutzerhandbuch 17](#_Toc467482630)

[11 Zusammenfassung 17](#_Toc467482631)

[12 Quellenverzeichnis 18](#_Toc467482632)

[13 Abbildungsverzeichnis 19](#_Toc467482633)

[14 Tabellenverzeichnis 20](#_Toc467482634)

[15 Codeverzeichnis 21](#_Toc467482635)

[16 Anhang 22](#_Toc467482636)

**Einleitende Bemerkungen**

**Notationen**

Beschreibung wie Code, Hinweise, Zitate etc. formatiert werden

# Projektmanagement

## Metainformationen

### Team

### Betreuer

### Partner

### Ansprechpartner

## Vorerhebungen

### Projektzieleplan

Projektziele-Hierarchie - SMART

### Projektumfeld

* Identifizierung der Stakeholder
* Charakterisierung der Stakeholder
* Maßnahmen
* Grafische Darstellung des Umfeldes

### Risikoanalyse

* Incl. Risikomatrix

## Pflichtenheft

### Zielbestimmung

* Projektbeschreibung
* IST-Zustand
* SOLL-Zustand
* NICHT-Ziele (Abgrenzungskriterien)

### Produkteinsatz und Umgebung

* Anwendungsgebiet
* Zielgruppen
* Betriebsbedingungen
* Hard-/Softwareumgebung

### Funktionalitäten (FURPS)

* MUSS-Anforderungen
* Funktional
* Nicht-funktional
* KANN-Anforderungen
* Funktional
* Nicht-funktional

### Testszenarien und Testfälle

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

### Liefervereinbarung

* Lieferumfang
* Modus
* Verteilung(Deployment)

## PLANUNG

### 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

# Problemanalyse

## USE-Case-Analyse

* 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)

## Domain-Class-Modelling

* "Dinge" (Rollen, Einheiten, Geräte, Events etc.) identifizieren, um die es im Projekt geht
* ER-Modellierung oder Klassendiagramme
* Zustandsdiagramme (zur Darstellung des Lebenszyklus von Domain-Klassen darstellen)

## User-Interface-Design

* Mockups
* Wireframes

# Systementwurf

## Architektur

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),
* 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. Plugin, Pipes and Filters)
  + MVC
  + Schichten
  + Pipes
  + Request Broker
  + Service-Oriented

## Datenhaltungskonzept

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

## Ein- Ausgabekonzept

* Design des UIs
* Dialoge, Dialogsteuerung, Ergonomie, Gestaltung, Eingabeüberprüfungen

## Konzept der Ausnahmebehandlung

* Systemweite Festlegung, wie mit Exceptions umgegangen wird
* Exceptions sind primär aus den Bereichen UI, Persistenz, Workflow-Management

## Sicherheitskonzept

* Design der Security-Elemente
* Design von Safety-Elementen (Fehlertoleranz, Verfügbarkeit etc.)

## Design der Testumgebung

* wie wird getestet (Unit-Testing, Integrationstesting, Systemtests, Akzeptanztests)
* Testumgebung, Testprozess, Teststrategie, Testmethoden, Testfälle

## Design der Ausführungsumgebung (Environment)

* Deployment (DevOps)
* Betrieb (besonders Hoch- und Hertunerfahren der Anwendung)

## Detailentwurf

Design jedes einzelnen USE-Cases

* Design-Klassendiagramme vom Domain-Klassendiagramm ableiten (incl. detaillierter Darstellung und Verwendung von Vererbungshierarchichen, abstrakten Klassen, Interfaces)
* Sequenzdiagramme vom System-Sequenz-Diagramm ableiten
* Detaillierte Zustandsdiagramme für wichtige Klassen

Design-Klassen für jeden einzelnen USE-Case können sein:

* UI-Klassen
* Data-Access-Klassen
* Entity-Klassen (Domain-Klassen)
* Controller-Klassen
* Business-Logik-Klassen
* View-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

# Implementierung

Detaillierte Beschreibung der Implementierung aller Teilkomponenten der Software entlang der zentralsten Use-Cases:

* GUI-Implementierung
* Controllerlogik
* Geschäftslogik
* Datenbankzugriffe

Detaillierte Beschreibung der Teststrategie:

* UNIT-Tests (Funktional)
* Integrationstests

Zu Codesequenzen:

* kurze Codesequenzen direkt im Text (mit Zeilnnummern auf die man in der Beschreibung verweisen kann)
* lange Codesequenzen in den Anhang (mit Zeilennummer) und darauf verweisen

# Deployment

* Umsetzung der Ausführungsumgebung
* Deployment
* DevOps-Thema

# Tests

## Systemtests

aller implementierten Funktionalitäten lt. Pflichtenheft

* Beschreibung der Teststrategie
* Testfall 1
* Testfall 2
* Tesfall 3
* …

## Akzeptanztests

# Projektevaluation

siehe Projektmanagement-Unterricht

# Betriebswirtschaftlicher Kontext

BW-Teil

# Benutzerhandbuch

falls im Projekt gefordert

# Zusammenfassung

* Etwas längere Form des Abstracts
* Detaillierte Beschreibung des Outputs der Arbeit

# Quellenverzeichnis

# Abbildungsverzeichnis

# Tabellenverzeichnis

# Codeverzeichnis

# Anhang

(inkl. Beratungsprotokollen mit Betreuungslehrer/in!)

Lange Codebeispiele