

TAU INF202 Software Engineering
Individuelles Projekt
Pflichtenheft

Projektdokumentation

Version: 2023.v1.0

Status: abgegeben

Projekt: Schuleverwaltungsprogramm

Verantwortliche/r:

Mohamad Eyad Abras, e170501104@stud.tau.edu.tr

Muhammed Alobayd, e170501109@stud.tau.edu.tr

Stakeholder: Dr. Ömer Karacan, omer.karacan@tau.edu.tr

Dokumentenverwaltung

Dokument-Historie

Version	Status *)	Datum	Verantwortlicher	Änderungsgrund
v1.0	Entwurf	13.04.2023	Alle Mitglieder	Projektdokumentation entwerfen
V1.0	abgegeben	17.04.2023	Alle Mitglieder	-

**) Sofern im Projekt nicht anders vereinbart, sind folgende Statusbezeichnungen zu verwenden (in obiger Tabelle und am Deckblatt):*

Dokument-Status: Entwurf / in Review / freigegeben (abgegeben)

Dokument wurde mit folgenden Tools erstellt:

Microsoft Office Word

Microsoft Office Power Point

Inhaltsverzeichnis

1. Einleitung.....	4
2. Ausgangssituation und Ziele	5
2.1 Einleitung	5
2.2 Problemstellung (Funktionalität)	5
2.3 Stakeholder (Anwender):	5
2.4 Systemumfeld (Einsatzumgebung)	5
2.5 Rahmenbedingung (Einschränkungen).....	6
2.6 Ziele (Lösung)	6
3. Gesamtarchitektur	7
3.1 Einleitung	7
3.2 Gesamtarchitektur	7
3.3 Externe Schnittstellen	7
4. Funktionale Anforderungen	9
4.1 Einleitung	9
4.2 UI Use Cases	9
4.3 Technischen und fachliche Anforderungen	10
4.4 Datenmodel	11
5. Nichtfunktionale Anforderungen	12
5.1 Einleitung	12
5.2 Nicht-funktionale Anforderungen an die Systemarchitektur (Architekturmuster, Deployment).....	12
5.3 Nicht-funktionale Anforderungen an die Entwicklungsumgebung.....	12
5.4 Nicht-funktionale Anforderungen an die Entwicklungswerkzeuge (Sprache, IDE, Frameworks)	12
5.5 Nicht-funktionale Anforderungen an die Teststrategie (Qualitätssicherung)	12
6. Abnahmekriterien.....	13
7. Projekt Meilensteine	14
8. Referenzen.....	15

1. Einleitung

Dieses Dokument ist nur ein Prototyping eines Schulverwaltungssystem Anforderungen zu definieren. Die Use Cases und Anforderungen beschreiben in diesem Dokument die Wünschen von Stakeholders.

Im Kapitel 2 „Ausgangssituation und Ziele“ sind Hauptrunden und Ziele, die zeigen, welche Rolle das Schulverwaltungssystem spielt, die täglichen Aktivitäten für Schulen zu verwalten und automatisieren.

Im Kapitel 3 „Gesamtarchitektur“ für ein Schulverwaltungssystem-Projekt ist ein wichtige Komponente und Aspekt für die erfolgreiche Entwicklung der Anwendung. Dies bietet für zukünftige verschiedene Möglichkeiten für eine effiziente Implementierung und Wartung des Systems.

Im Kapitel 4 „Funktionale Anforderungen“ ist ein wichtiger Bestandteil für das Projekt, um ihre täglichen Verwaltungsaufgaben effektiv und effizient zu erledigen. Unser Projekt ist ein komplexes Projekt, das spezifische Anforderungen erfüllen muss, um die Bedürfnisse der Lehrer und der Schulleitung zu erfüllen. Dies beschreibt Anforderungen durch User Stories, Use Cases und technischen und fachlichen Anforderungen und Datenmodellen.

Im Kapitel 5 „Nichtfunktionale Anforderungen“ in unserem System beinhaltet die Beschreibung von Anforderungen, die nicht direkt mit der Funktionalität des Systems zusammenhängen, sondern andere Aspekte vom System abhängig sind.

Im Kapitel 6, den Abnahmekriterien, werden die Artefakte genau definiert, die bei der Abgabe des Projekts an den Stakeholder unverändert bleiben müssen. Es ist nicht erlaubt, zu manipulieren, ohne dass der Stakeholder darüber informiert.

Im Kapitel 7 „Projekt Meilensteine“ sind Punkte im Verlauf eines Projekts, an dem bestimmte Ziele erreicht werden sollen. also es enthält die wichtigsten Termine, an denen bestimmte Ergebnisse erreicht werden sollen.

Im Kapitel 8 „Referenzen“ sind die wichtigsten Quellenangaben aufgelistet.

2. Ausgangssituation und Ziele

2.1 Einleitung

Mit fortschreitender Technologie muss die Dokumentation digitalisiert werden, um menschliche Fehler zu reduzieren, die Genauigkeit und Leistung zu erhöhen und Zeit und Mühe zu sparen. In diesem Kapitel werden die Ausgangssituation diskutiert, die Rahmenbedingung bestimmt, und welche Probleme durch den Einsatz des neuen Systems gelöst wird und die Ziele definiert.

2.2 Problemstellung (Funktionalität)

Das Schulverwaltungssystem wurde entwickelt, um verschiedene Herausforderungen anzugehen, mit denen Schulen konfrontiert sind, darunter ineffiziente Verwaltungsprozesse, schlechte Kommunikation zwischen Lehrern und Schülern und Schwierigkeiten bei der Verwaltung von Schülerdaten. Die aktuellen manuellen Prozesse zur Verwaltung von Schülerakten, Anwesenheitsverfolgung und Unterrichtsplanung sind zeitaufwändig und fehleranfällig.

Durch die Implementierung eines Schulverwaltungssystems können diese Herausforderungen überwunden werden. Das System bietet eine benutzerfreundliche Oberfläche, die Verwaltungsaufgaben rationalisiert und es Lehrern und Administratoren ermöglicht, Schülerdaten zu verwalten, die Anwesenheit zu verfolgen und den Unterricht effizienter zu planen. Schließlich bietet das System robuste Sicherheitsfunktionen zum Schutz der Schülerdaten und gewährleistet Vertraulichkeit und Integrität.

2.3 Stakeholder (Anwender):

Das Schulverwaltungssystem ist so konzipiert, dass es mehreren Interessengruppen innerhalb der Schulgemeinschaft, einschließlich Lehrern, Schülern und Administratoren, zugute kommt, als Beispiel:

- Für Administratoren bietet das System eine Plattform zur Verwaltung des Schulbetriebs und der Ressourcen. Dazu gehören Funktionen zur Verwaltung von Schülerdaten, Terminplanung und Budgetierung.
- Für Lehrer bietet das System Lehrern eine benutzerfreundliche Plattform zur Verwaltung von Schülerdaten und zur Verfolgung des Fortschritts der Schüler. Das System rationalisiert Aufgaben wie die Benotung.

Insgesamt ist das Schulverwaltungssystem so konzipiert, dass es allen Interessengruppen innerhalb der Schulgemeinschaft zugute kommt, indem es die Effizienz administrativer Aufgaben verbessert und Echtzeitdaten und Feedback zur Schülerleistung bereitstellt.

2.4 Systemumfeld (Einsatzumgebung)

Das Schulmanagementsystem wird innerhalb der bestehenden technischen Infrastruktur der Schule in Betrieb genommen. Dazu gehören ein lokales Netzwerk (LAN), das alle Computer und Geräte innerhalb der Schule verbindet, sowie ein Internetzugang für den Fernzugriff und Online-Ressourcen.

Das System wird über eine webbasierte Schnittstelle zugänglich sein, auf die autorisierte Benutzer von jedem Computer oder Gerät mit Internetzugang zugreifen können. Das System wird auf einem dedizierten Server innerhalb des Schulnetzwerks gehostet, der ausreichend Rechenleistung und Speicherplatz zur Verfügung stellt, um den Betrieb des Systems zu unterstützen.

2.5 Rahmenbedingung (Einschränkungen)

Die wichtigsten Einschränkungen befinden sich in, der Wahl der Software Tools:

- Als Software-Entwicklungstool soll entweder Eclipse oder IntelliJ verwendet werden,
- Die Backend Applikationen sollen mit Java Framework und Spring Framework,
- Die Frontend Applikationen mit JavaFX Rich Client Technologie realisiert werden, und
- Die persistenten Daten sollen in einer SQL-Datenbank abgespeichert werden.

Auch als technische Vorgaben:

- Unterstützung mehrerer Benutzerrollen, und
- sichere Datenübertragung und -speicherung

Auch als Proof-of-Concept zu erreichen, Die Entwickler muss liefern:

- Eine detaillierte Demonstration der Systemmerkmale und -funktionalität

2.6 Ziele (Lösung)

Durch Ersetzung von das SV-Programm sollen die folgenden Ziele erreicht werden:

- **Effizienz verbessern**, Das System sollte die Effizienz des Schulbetriebs verbessern indem:
 - Es Verwaltungsaufgaben automatisiert.
 - Den Papierkram reduziert.
 - Den Benutzern eine optimierte Schnittstelle zur Verwaltung verschiedener Aspekte des Schulbetriebs bietet.
- **Verbesserte Datengenauigkeit**, Das System sollte
 - Genaue und aktuelle Daten zu Schülerleistungen, und anderen wichtigen Kennzahlen liefern,
 - Durch die Bereitstellung genauer Daten kann die Schule fundiertere Entscheidungen über die Schülerplatzierung, die Entwicklung des Lehrplans und die Ressourcenzuweisung treffen.
- **Transparenz erhöhen**, Das System sollte Transparenz über die Schülerleistungen und den Schulbetrieb bieten, indem:
 - Es Lehrern und Administratoren ermöglicht, Echtzeitdaten zu Noten und anderen Metriken einzusehen.
- **Gewährleistung der Sicherheit**, Das System sollte die Sicherheit und Vertraulichkeit von Schüler- und Benutzerdaten gewährleisten, indem:
 - Robuste Sicherheitsmaßnahmen implementiert werden.
 - Best Practices für den Datenschutz eingehalten werden.

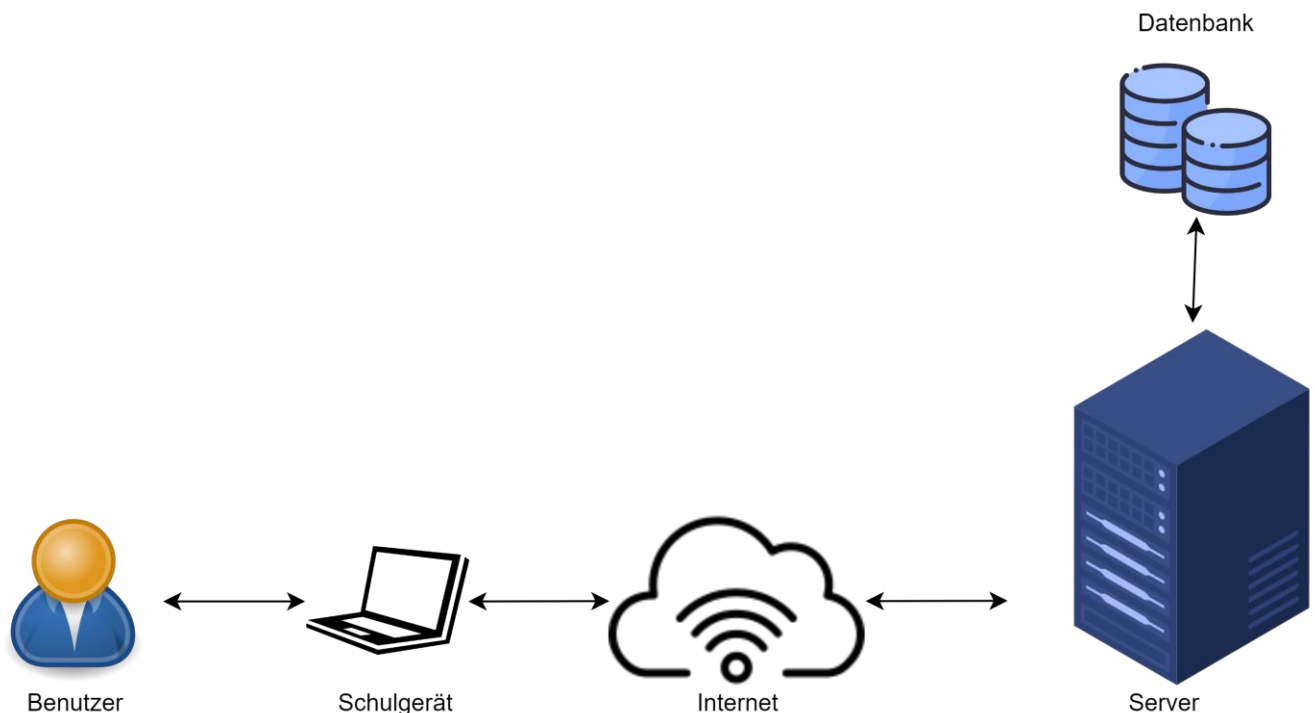
3. Gesamtarchitektur

3.1 Einleitung

In diesem Kapitel werden die Systemgrenzen definiert, die mit entsprechenden Illustrationen spezifiziert werden. Die externen Schnittstellen werden vom Anwender unter Berücksichtigung der Systemgrenzen definiert. Die notwendigen Systemkomponenten und Datenstrukturen werden definiert und modelliert.

3.2 Gesamtarchitektur

Unser Schulverwaltungssystem folgt einer Benutzer/Server-Architektur. Der Benutzer kann Webbrowser verwenden, um über das lokale Netzwerk (LAN) der Schule oder überall über das Internet mit dem HTTPS-Protokoll auf das System zuzugreifen.



Der Server präsentiert dem Benutzer die Website und steuert die Geschäftslogik. Es steuert die Interaktionen zwischen der Anwendung und dem Benutzer.

Die Daten/Datenbank verwaltet die Daten der Anwendung wie Schülerdaten, Lehrerdaten usw. Sie speichert diese Daten in einem relationalen Datenbankverwaltungssystem wie MySQL. Die Benutzerebene interagiert mit dem Server, um Anforderungen zu stellen und Daten aus der Datenbank abzurufen, und zeigt dann dem Benutzer die vom Server abgerufenen Daten an.

3.3 Externe Schnittstellen

Die externen Schnittstellen sind:

- Log-In Schnittstelle: Wo der Benutzer seinen eigenen Benutzernamen und sein eigenes Passwort eingeben muss

- Modul-Auswahl Schnittstelle: Wo der Benutzer eines der Module auswählen kann.(Schüler-Lehrer-Schülernoten-Gebühren)
- Schüler-Modul Schnittstelle: Information der Studenten einzufügen, zu bearbeiten ode zu löschen
- Lehrer-Modul Schnittstelle: Information der Lehrer einzufügen, zu bearbeiten ode zu löschen
- Gebühren-Modul Schnittstelle: Gebühren der Studenten zu erfassen oder zu berchnen
- Schülernoten-Modul Schnittstelle: Noten der Studenten einzufügen, zu bearbeiten ode zu löschen

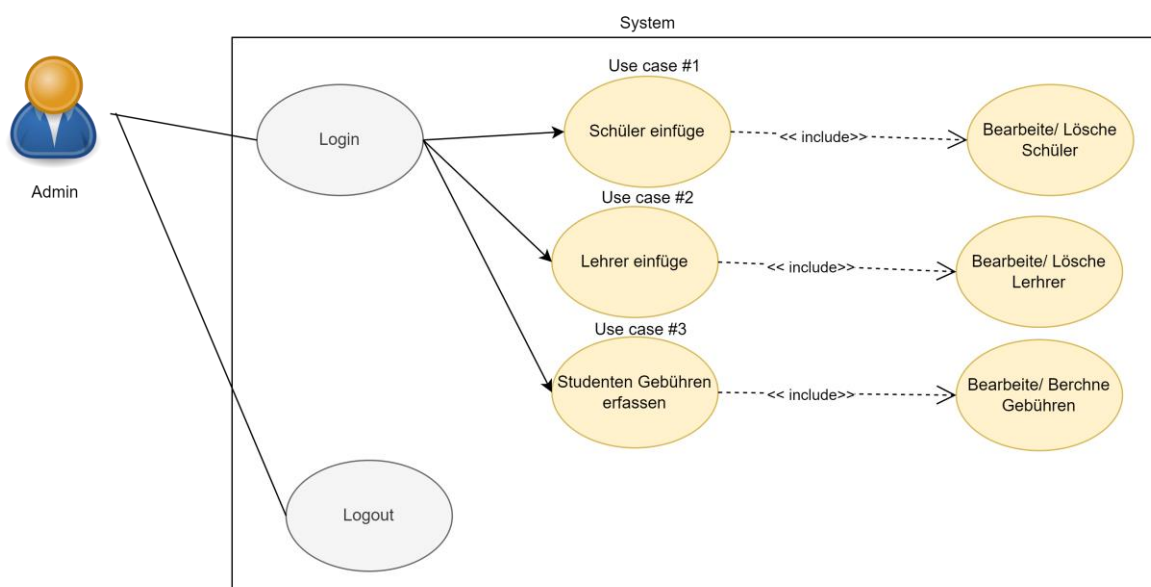
4. Funktionale Anforderungen

4.1 Einleitung

Dieses Kapitel beschreibt Anforderungen durch User Stories, Use Cases, technischen und fachlichen Anforderungen und Datenmodellen. Außerdem wird eine Gui Prototyp gezeigt.

4.2 UI Use Cases

Wie es aus der unten folgenden Diagramm deutlich zu sehen, dass System ermöglicht den Benutzer viele verschiedene Szenarien oder Funktionen, die der Benutzer (in dieser Szenerio ist der Benutzer ist der Schulverwalter) machen kann.

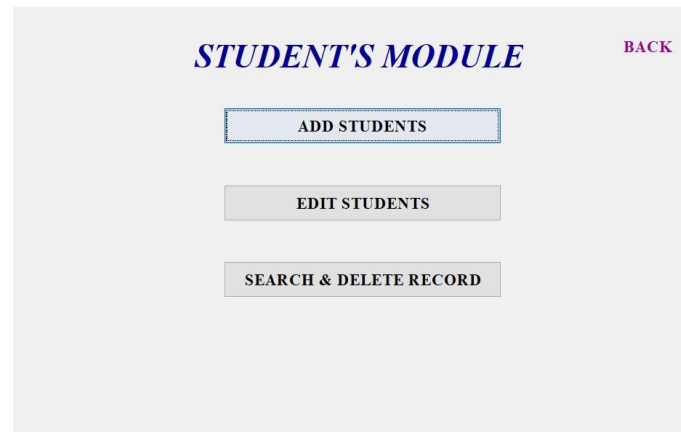


Als Beispiel dazu **Use-case #1: (Schüler einfüge)**

- Step #1: Die Login-Seite geöffnet wird: nachdem der Anwender das Programm gestartet hat. Nun ist es notwendig, einen Username und Passwort abzugeben.

The screenshot shows the login page of the School Management System. It features the logo of the University of Applied Sciences (HAW) and the title "Schulverwaltungssystem". Below the title, there is a "LOGIN PAGE" section with input fields for "User Name:" and "Password:". A "Submit" button is located to the right of the password field. At the bottom, there is a link for "Forgot Password ?" and a "Enter The Secret Code" field with a "submit" button.

- Step #2: Student-Modul auswählen: Nachdem der Benutzer das Schüler-Modul auswählen, kann er der gewünschte Funktion/Aufgaben erledigen. Die zugängliche Aufgaben sind: Student einfügen, Student bearbeiten oder Student löschen.



ID	NAME	FATHER'S NAME	PHONE	FATHER'S PHONE	CLASS	ROLL NUMBER	ADDRESS	FTI NUMBER

- Step #3: Füllen der Lücken: Hier kann der Benutzer oder der Administrator die Kappen mit den Informationen des neuen Schülers füllen, die später in der Datenbank gespeichert werden. Diese neuen Daten können später überprüft werden, wann immer der Administrator möchte.
- Step #4: Löschen von Schülerinformationen: Wenn der Administrator die Informationen eines Schülers löschen möchte, kann er dies einfach tun, indem er die Rollennummer des Schülers in der Schülerliste sucht und dann die gesuchten Schülerinformationen löscht

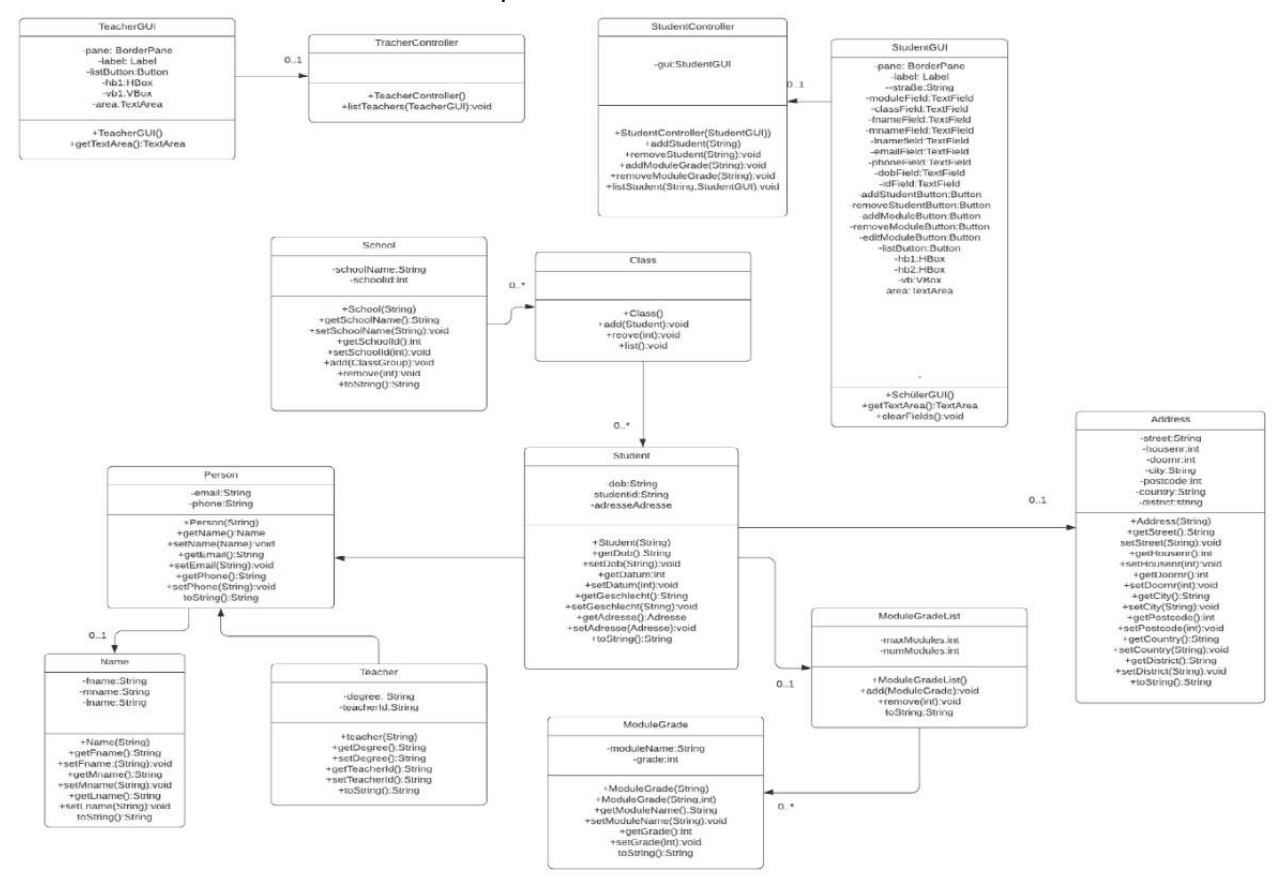
4.3 Technischen und fachliche Anforderungen

- XAMPP ist eine plattformübergreifende Apache-Distribution, die MySQL, PHP und Perl enthält.
- Webserver mit all seinen Komponenten auf unserem lokalen Rechner zu installieren. es erhältet die Basis um z. B. Content Management Systeme (Joomla, Typo3 etc.) oder Ihre eigenen PHP-Skripte.
- NetBeans IDE als Hauptsächlichprogrammierungstool. Es ist eine freie integrierte Entwicklungsumgebung
- Das System wird in Java Programmiersprache entwickelt.
- MySQL Workbench, ermöglicht das Entwerfen, Erstellen, Durchsuchen der Datenbankschemas und Bearbeiten mit Datenbankobjekten.
- Die Anwendung der MySQL Datenbank als Haupt Datenbankspeicher
- Es ist wichtig, dass das System einfach zu bedienen und benutzerfreundlich ist, um sicherzustellen, dass alle Benutzer das System effektiv nutzen können.



4.4 Datenmodell

Wie deutlich unten zu sehen ist, wird der Zusammenhang zwischen GUI und Klassendiagramm in UML Klassendiagramm modelliert, also es wird API-Schnittstellen berücksichtigen, Um wichtige Informationen und Daten effizient zu speichern.



5. Nichtfunktionale Anforderungen

5.1 Einleitung

In diesem Kapitel werden die nicht-funktionalen Anforderungen deutlich beschrieben und darüber hinaus wird die Qualität und Geschwindigkeit für das Gesamtsystem festgestellt.

5.2 Nicht-funktionale Anforderungen an die Systemarchitektur (Architekturmuster, Deployment)

Die Entwicklungswerkzeuge sollten die Möglichkeit bieten, neue Funktionen und Komponenten hinzuzufügen oder zu erweitern, um die Bedürfnisse des Projekts zu erfüllen. Mit anderen Worten sollten Komponenten des Systems in eigenen Containern oder Prozessen ausgeführt werden, um die Skalierbarkeit und Ausfallsicherheit zu verbessern.

Das System sollte eine geeignete Datenbank verwenden, um die Datenintegrität und -sicherheit zu gewährleisten.

5.3 Nicht-funktionale Anforderungen an die Entwicklungsumgebung

Der Entwickler muss mit dem Kunden das System überprüfen und weiterentwickeln.

5.4 Nicht-funktionale Anforderungen an die Entwicklungswerkzeuge (Sprache, IDE, Frameworks)

- Der Applikation wird in NetBeans mit Java 11 und swing geschrieben.
- MySQL wird verwendet um Schema für die Datenbank zu entwerfen und erstellen

5.5 Nicht-funktionale Anforderungen an die Teststrategie (Qualitätssicherung)

Es wird erwartet, dass das System eine angemessen kurze Reaktionszeit hat. Der Administrator sollte sich anmelden können und innerhalb von 2-3 Sekunden eine Antwort auf diese Anfrage erhalten. Die Leistung des Systems ist je nach Modus unterschiedlich.

6. Abnahmekriterien

Die Abnahmekriterien ist durch **5W1H-Methode** definiert:

❖ Wer wird das Produkt verwenden?	<ul style="list-style-type: none">• Schulverwalter, Lehrer und Schüler• Jeder Benutzer hat je nach Rolle unterschiedliche Zugriffsebenen
❖ Was macht das Produkt?	<p>Das System muss verschiedenen Aufgaben erledigen, Beispiel:</p> <ul style="list-style-type: none">• Verwalten von Studentenakten, einschließlich persönlicher Informationen, akademischer Leistungen usw.• Verwalten von Lehrer- und Personalakten, einschließlich personenbezogener Daten.• Erstellen von Berichten über Schüler- und Schulleistungen.• Erfassung und Berechnung des Studienbeitrags
❖ Wann wird das Produkt?	während des gesamten Schuljahres täglich verwendet werden.
❖ Warum wird das Produkt verwendet?	<ul style="list-style-type: none">• Der Verwaltungsaufgaben zu automatisieren• Datengenauigkeit verbessern• Verwaltungskosten und -sdauern reduzieren
❖ Wo wird das Produkt eingesetzt?	Im lokalen Netzwerk der Schule verwendet werden
❖ Wie wird das Produkt verwendet?	Benutzer können das Schulverwaltungssystem über eine webbasierte App finden. Die App wird eine benutzerfreundliche Oberfläche haben, die einfach zu bedienen ist. Benutzer melden sich mit ihrem eigenen Benutzernamen und Passwort beim System an, um eine oder mehrere schwerwiegende Aufgaben zu erledigen.

Das Projekt wird mit den folgenden Artefakten abgegeben:

- Dokumentation:
 - Pflichtenheft: INF202-Schulverwaltungsprogeamm-Pflichtenheft-2023.v1.0
- Software
 - Link zu GitHub Projekt: <tbd>
- Evidenz:
 - System/Software-Demo via Videoclip: <tbd>

Anm.: Die Abgabetermine der Projetartefakts werden durch den Stakeholder festgelegt!

7. Projekt Meilensteine

	Dauer	Deadline	Beschreibung
Meilensteine #1	4 wochen	24.03.2023	Lastenheft ist fertiggestellt und mit dem Stakeholder abgestimmt
Meilensteine #2	3 wochen	14.04.2023	Pflichtenheft ist fertiggestellt und mit dem Stakeholder abgestimmt
Meilensteine #3	3 wochen	05.05.2023	Diese Aufgaben müssen erledigt werden: <ul style="list-style-type: none">• Komponenten-basierte Architektur ist entworfen und komponentenweise implementiert.• Die externen Schnittstellen (Webservices) wurden entworfen/implementiert.• Die GUI Komponente ist ansatzweise fertig.
Meilensteine #4	3 wochen	26.05.2023	Diese Aufgaben müssen erledigt werden: <ul style="list-style-type: none">• Die Test-Cases wurden aus den Use Cases abgeleitet und ansatzweise beschrieben.• Die Testumgebung ist vorbereitet und ein Smoke Test ist durchgeführt.
Meilensteine #5	2 wochen	07.06.2023	Endabgabe, Dokumentationsmappe

8. Referenzen

1. Bhati, N. (2022) API examples and use cases, Tech Blog | Mobile App, eCommerce, Salesforce Insights. Available at: <https://www.emizentech.com/blog/api-examples.html> (Accessed: April 15, 2023).
2. Luke Welling, L. T. PHP and MySQL Web Development (4th Edition) 4th Edition
3. Hartman, J. (2023) Java Swing Tutorial: How to create a GUI application in Java, Guru99. Available at: <https://www.guru99.com/java-swing-gui.html#:~:text=like%20the%20Frame,-.What%20is%20GUI%20in%20Java%3F,can%20interact%20with%20an%20application>. (Accessed: April 11, 2023).