

Entwicklung eines webbasierten Ressourcenverwaltungssystems



Fachinformatiker – Anwendungsentwicklung

Abschlussprüfung Sommer 2020

e.Consult AG

Projektdokumentation

Ali Krezan

Inhalt

Entwicklung eines webbasierten Ressourcenverwaltungssystems	
1. Projektbeschreibung	1-1
1.1. Der Ausbildungsbetrieb.....	1-1
1.2. Die Ausgangssituation	1-1
1.3. Das Ziel dieses Projektes	1-1
1.4. Das Projektumfeld	1-2
1.5. Der Projektbeteiligte	1-2
2. Projektplanung	2-2
2.1. Ist-Analyse	2-2
2.2. Soll-Konzept.....	2-3
2.3. Lösungskonzept.....	2-4
2.4. Projektablauf	2-5
2.5. Zeitplanung.....	2-6
2.6. Ressourcenplanung	2-6
2.7. Kostenplanung.....	2-6
3. Projektdurchführung	3-8
3.1. Aufbau und Struktur der Datenbank.....	3-8
3.2. Benutzeroberfläche	3-9
3.3. Programmiersprache.....	3-11
3.4. Technische Dokumentation.....	3-13
4. Projektabschluss.....	4-13
4.1. Qualitätssicherung.....	4-13
4.2. Aufgetretene Probleme & Maßnahmen	4-15
4.3. Fazit und Ausblick.....	4-15
5. Technische Dokumentation.....	5-1
6. Benutzerdokumentation	6-1
7. Quellenverzeichnis	I
7.1. Literatur	I
7.2. Weblinks	I
7.3. Abbildungen	I
7.4. Testprotokoll	II
7.5. Zeitplanung.....	IV
7.6. Pflichtenheft	V

1. Projektbeschreibung

1.1. Der Ausbildungsbetrieb

Die Firma e.Consult AG ist ein im Jahr 2000 gegründetes IT-Unternehmen mit Sitz in der Landeshauptstadt Saarbrücken. Das Unternehmen entwickelt digitale Kommunikationslösungen und Möglichkeiten zum einfachen und sicheren elektronischen Datenaustausch.

Mit über 13.000 Anwaltskanzleien und 24 namhaften Versicherern (u.a. Allianz, HUK Coburg, ADAC Rechtsschutz) ist die e.Consult AG eines der innovativsten Unternehmen der LegalTech-Szene in Deutschland und mit über 3,5 Millionen Schadenseinsendungen Marktführer in der elektronischen Rechtskommunikation zwischen Versicherern und Anwälten. Des Weiteren zählen auch Wirtschaftsprüfer, Steuerberater, Notare, Autohäuser, Handwerker und Gutachter zum Kundenkreis.

Folgende Leistungen werden u.a. angeboten:

- ✓ Sicherer Datenaustausch zwischen Anwälten, Mandanten und weiteren Kommunikationspartnern.
- ✓ Elektronische Ermittlungsaktenbesorgung.
- ✓ Kostenlose Rechtsschutzabwicklung für Anwälte.
- ✓ Sicherer Datenaustausch per Smartphone.
- ✓ Digitalisierung von Partnernetzwerken.
- ✓ Schnittstellen für zur Integration in bestehende Prozesse.
- ✓ Elektronische Beauftragung externer Dienstleister.

1.2. Die Ausgangssituation

Es besteht Unzufriedenheit mit der zurzeit üblichen Praxis, die Belegung bzw. den Bedarf an Belegzeiten des Konferenzraumes händig an der Tür per Notizzettel anzumelden. Es ergibt sich so die Notwendigkeit der physischen Anwesenheit, was die Koordinierung und Terminierung fälliger Termine und Zeiten erheblich beeinträchtigt

Der Leiter der e.Consult Entwicklungsabteilung hat deshalb vorgeschlagen, ein neues webbasiertes Programm zu entwickeln, das alle Anforderungen hinsichtlich dieser Problematik abdecken kann. Mit diesem neuen Programm sollen sich sämtliche Arten von Veranstaltungen und auch anderen Ressourcen mit Leichtigkeit verwalten lassen.

1.3. Das Ziel dieses Projektes

Das Ziel dieses Projektes ist die einfache Handhabung und Einsichtnahme der Belegzeiten des Konferenzraumes und anderer Ressourcen, zu jeder Zeit und an jedem Ort des

Unternehmens, über ein Webinterface.

Freie Kapazitäten können so leicht eruiert und terminliche Konflikte bereits im Vorfeld sicher vermieden werden.

Für die Datenhaltung wird eine neue Datenbank installiert, auf die später die neue Anwendung zugreifen wird. Die Anwendung liegt zentral auf einem Server, wodurch der Aufwand bei Installationen auf einem Arbeitsplatz-PC und bei eventuellen Programmupdates entfällt.

Die Authentifizierung der Anwender erfolgt durch ein Active Directory. So meldet sich der jeweilige Anwender mit einem Benutzernamen und einem Passwort an. Zusätzlich muss der Benutzer sich bei der erstmaligen Anmeldung mit weiteren Daten wie Vorname, Nachname und E-Mail-Adresse für die Anwendung registrieren.

1.4. Das Projektumfeld

Die Entwicklung dieser neuen Applikation findet in den Räumen der e.Consult AG statt. Alle Arbeiten werden in der Abteilung Anwendungsentwicklung durchgeführt.

1.5. Der Projektbeteiligte

Um das Projektziel zu erreichen, ist eine enge Zusammenarbeit mit den Mitarbeitern der e.Consult notwendig. Ansprechpartner sind Herr Traut und Herr Ariganello. Sie werden die Teilergebnisse begutachten und bewerten. Den Piloteinsatz für das Projekt wird Herr Hossain durchführen, da er verantwortlich für die Tests ist.

Projektschnittstellen	
Herr Gino Ariganello	Leiter Operation (Projektverantwortlicher)
Herr Julian Traut	Technical Consultant
Herr Shantu Asif Hossain	Senior Softwareentwickler (Tester)
Ali Krezan	Umschüler (Entwickler)

Tabelle: Schnittstellen 1

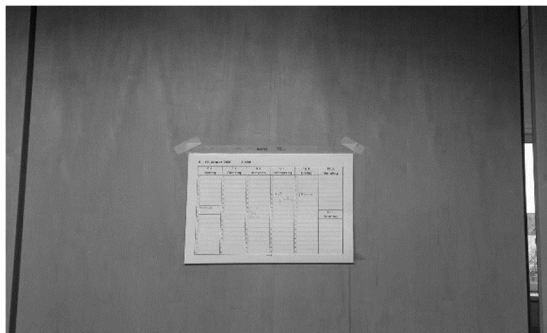
2. Projektplanung

2.1. Ist-Analyse

Die Mitarbeiter der Firma e.Consult benötigen immer wieder Zugriff auf bestimmte räumliche Ressourcen. Dazu gehören in erster Linie Konferenzräume, aber auch Hardware, wie Beamer und TVs für Präsentationen oder sonstige Ausstattung. Ressourcen dieser Art stehen allen Abteilungen zur Verfügung und können bei Bedarf reserviert und genutzt werden.

Im ersten Entwicklungsschritt steht die Reservierung der Konferenzräume im Mittelpunkt. Die Verwaltung und Reservierung der Räume finden derzeit in Papierform statt (siehe Abbildung 1). Diese Form der Reservierung stellt sich jedoch als aufwendig, fehleranfällig und unkomfortabel heraus.

Um die Reservierungen einsehen zu können, ist der Gang zum Reservierungszettel an der entsprechenden Tür notwendig. Teilnehmer einer Konferenz müssen vom Ersteller oder einem anderen Teilnehmer persönlich oder über ein weiteres System wie Outlook informiert werden. Es ist unter diesen Umständen schwer, die Reservierung zu verändern oder zu löschen. Für die Verwaltung und Nutzung anderer Ressourcen (z.B.: Beamer) existiert derzeit kein System bzw. Reservierungsprozess.



Dieses Problem soll in Zukunft digital gelöst werden. Dafür ist ein webbasiertes Softwaresystem vorgesehen,

Deshalb erfolgt die Bedienung über den Browser, wobei hier ein Mobile-First-Ansatz zu wählen ist.

Die Persistenz wird durch die Verwendung eines relationalen Datenbankmanagementsystems (RDBMS) realisiert. Das System soll skalier-, wart- und erweiterbar sein.

Abbildung 2: Aushang an der Tür des Konferenzraumes

2.2. Soll-Konzept

Gegenstand dieser Arbeit ist die Realisierung der Ressourcenreservierungsverwaltung. Das Ziel ist, dass die zu entwickelnde Applikation zukünftig die veraltete Methode gänzlich ersetzen soll. Die neue Anwendung soll die Reservierung von Ressourcen, insbesondere der Konferenzräume, einfacher und übersichtlicher gestalten. Zudem sollen zukünftige Reservierungen korrekt abgearbeitet werden.

Die Hauptseite beinhaltet den Überblick über die Reservierungen in Form eines Kalenders. Diese ist von jedem Firmenangehörigen einsehbar. Zum Setzen oder Verändern von Reservierungen bzw. Kalendereinträgen wird eine Anmeldung vorausgesetzt. Die Anmeldung greift auf das Active-Directory des Unternehmens zurück. Die Einsicht und Änderung von Einträgen wird durch ein Rechte- und Rollensystem gesteuert und entsprechend begrenzt.

Die Änderung oder Löschung von Reservierungen hängt von der Rolle ab (Ersteller oder Admin). Hierfür wird zunächst ein relationales DBMS mit den erforderlichen Tabellen und Beziehungen erstellt.

Wenn ein Benutzer eine neue Reservierung erstellt und einen oder mehrere Teilnehmer wählt, bekommt jeder Teilnehmer eine Nachricht mit dem zugehörigen Link per E-Mail. Dieser Link verweist auf den Tab „Polls“. Hier erhält der Nutzer nach erfolgreicher Anmeldung die entsprechenden Veranstaltungen, die zu- oder abgesagt werden können.

Im Rahmen jeder Reservierung kann eine Checkliste angelegt werden. Diese Liste kann neben den benötigten Materialien, Verpflegung und Geräten auch z.B. mit einem Raumreinigungsplan erweitert werden. Die Applikation soll flexibel anwendbar sein. Das heißt dass diese so aufgebaut ist, dass sie auf allen Browsern und Mobilgeräten verwendet werden kann.

Der Einsatzbereich der Anwendung wird sich – wie bereits erwähnt - nicht nur auf die Raumreservierung beschränken. Sie wird auch für andere Bereiche, wie z.B. die Verwaltung von Geräten (Laptops) eingesetzt werden.

2.3. Lösungskonzept

Das ursprüngliche Pflichtenheft befindet sich als Anlage im Anhang 7.6-Pflichtenheft

Eine Auswahl der Werkzeuge, sowie ein Vergleich der Vorteile oder Nachteile, also eine Abwägung zwischen den technologischen Möglichkeiten der eigentliche Teil eines Lösungskonzepts darstellen sollte entfällt an dieser Stelle.

Von dem Unternehmen vorgegeben für die Applikation ist ASP.NET und C# mit Visual Studio Professional 2015 von Microsoft. Die Firma e.Consult hat eine homogene hauseigene Softwareentwicklung mit ASP.NET und C#.

Dies gilt auch für die Datenhaltung. Als Test Umgebung wird ein Microsoft SQL-Server 2016 Express vorgesehen, der auf einem Microsoft Windows-Server 2012 läuft.

2.4. Projektablauf

2.4.1. Model und Ansatz

Die Auswahl eines Entwicklungsmodells bei einem "Ein Personen Projekt" ist äußerst begrenzt. Zu Realisierung des Projekts wurde ein Paralleles Phasenmodell mit agilen Ansätzen (Hohe Kommunikation mit dem Auftraggeber, Testgetriebene Entwicklung etc.) eingesetzt.

Das heißt, in der Regel war es so, dass selbst, wenn eine Projektphase noch nicht ganz abgeschlossen war, vielleicht schon mit der nächsten Projektphase begonnen wurde, da Tätigkeiten anfallen, bevor andere beendet sind. Das führt so zum parallelen Phasenmodell.

2.4.2. Bestimmung der Anforderungen des Unternehmens

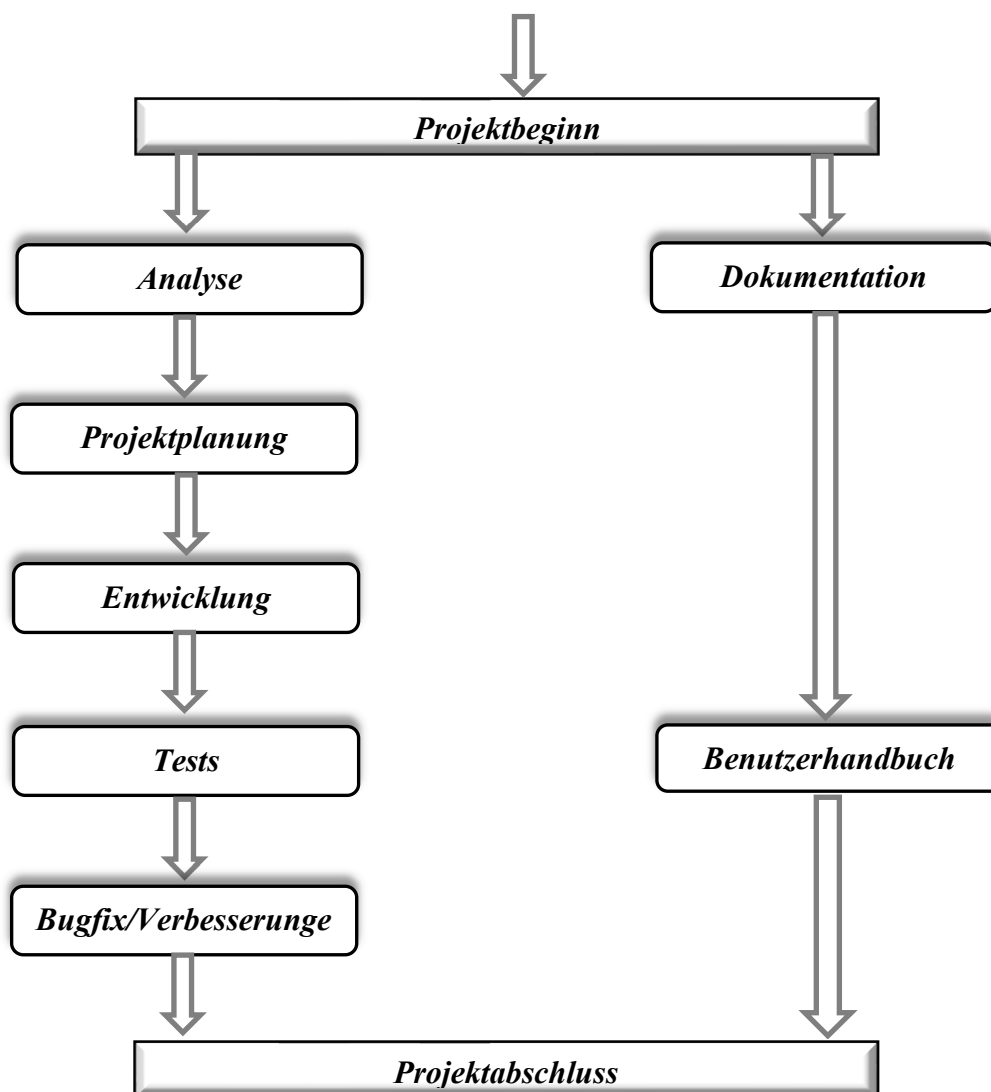


Abbildung 3: Projektablauf mit den einzelnen Projektschritten

2.5. Zeitplanung

Die Zeitplanung zur Durchführung des Projektes wurde auf den Zeitraum vom 16.03.2020 bis 03.04.2020 definiert. Die Projektdurchführung erfolgt in mehreren Projektphasen, deren Teilaufgaben und die jeweilige Zeiteinschätzung sind der folgenden Tabelle zu entnehmen. Eine detaillierte Zeitplanung befindet sich in den Anlagen unter dem Punkt [6.5].

Vorgang		Dauer in Stunden
Analyse		8
Projektplanung		7
Entwicklung	Entwurf	7
	Realisierung	30
Testphase	Funktionale Tests	12
	Nichtfunktionale Tests	
	Bugfix / Verbesserung	
Projektabschluss	Integration	4
	Abnahme	2

Abbildung 4: Zeitplanung

2.6. Ressourcenplanung

Planung, Durchführung, Tests und Dokumentation des Projektes finden an einem Standard-Entwickler-PC mit Windows 10 statt. Die verwendeten Werkzeuge sind Microsoft Visual Studio 2015 für die Programmierung und Microsoft Office 2010 für die Dokumentation. Die Datenbank liegt auf einem Microsoft SQL-Server 2016 Express, der auf einem Microsoft Windows-Server 2012 läuft. Für das Management der Datenbank wird das Tool Microsoft SQL-Server Management Studio v18.2 verwendet. Grafische Darstellungen, wie z.B. UML-Diagramme, werden mit draw.io angefertigt.

2.7. Kostenplanung

Als Kalkulationsansatz für das Projekt wurde eine Ableitung der klassischen Industriekalkulation verwendet.

Einzelkosten	Stunden h	Kosten / h	Gesamt
Entwicklerstunden	70	75,- €	5.250, - €
Materialeinsatz			

Gemeinkosten	Stunden h	Kosten / h	Gesamt
Entwicklerarbeitsplatz	70	90, - €	6.300, - €

Herstellkosten		11.550, - €
-----------------------	--	-------------

Abbildung: 5 Kostenplan

Da die Entwicklung im Rahmen eines Ausbildungspraktikums zum Fachinformatiker erfolgt, wurden, ausgehend von einem durchschnittlichen Entwicklerlohn von 75 €, diese als Lohnkostensatz angesetzt.

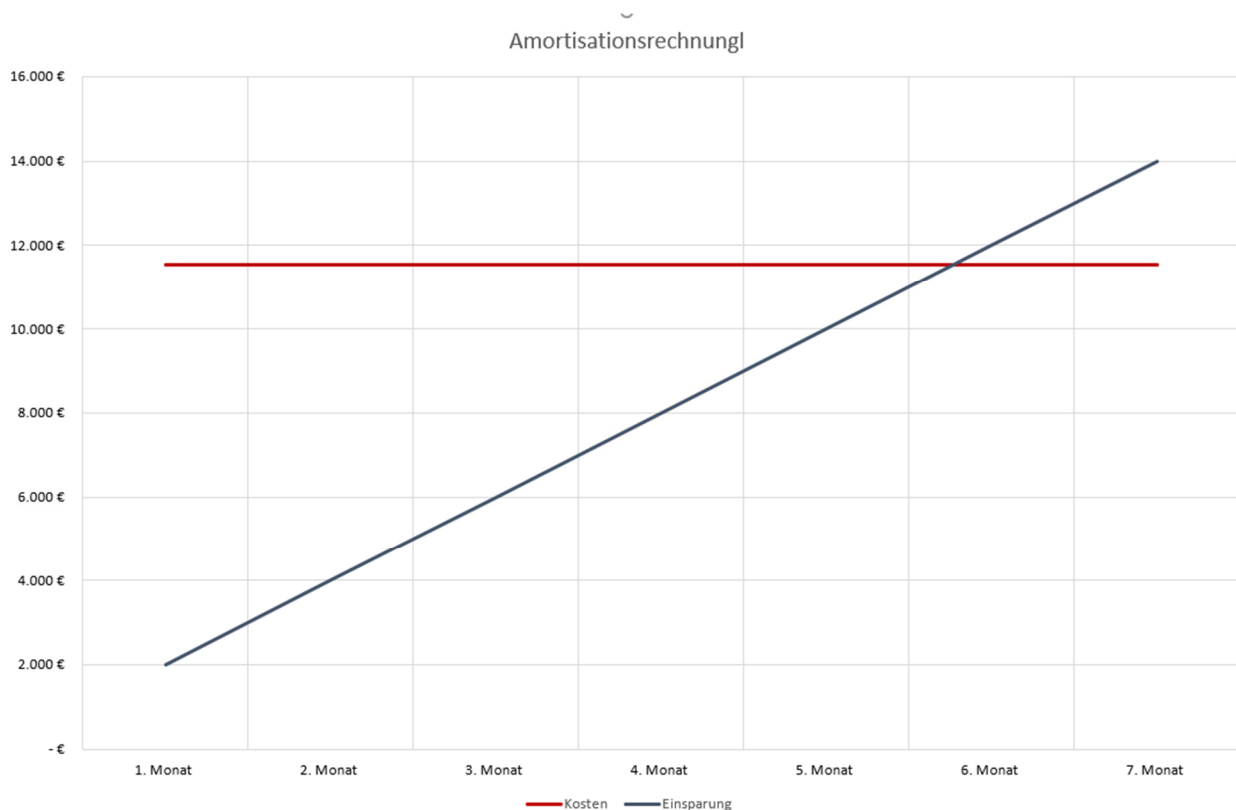
2.7.1. Nutzen und ROI (Return of Investment)

Ohne den entstehenden Imageschaden bei doppelter Belegung von Konferenzräumen bei Kundengesprächen oder Ausweichen auf ungeeignete Räume ist die momentane Situation sehr Zeit aufwendig. Dieser Zeitaufwand soll durch den Einsatz der Anwendung minimiert werden, was langfristig gesehen dann auch zu einer merklichen Kosteneinsparung führt.

Stundenkosten	Stunden h	Kosten / h	Gesamt
Arbeitsstundendurchschnitt im Monat	20	100; -€	2.000 - €

2.7.2. Amortisation des Projekts

Nach Einsatz der Software hat sich das Projekt im 6. Monat amortisiert. Also die Kostenersparnis übersteigt die eingesetzten Mittel.



3. Projektdurchführung

3.1. Aufbau und Struktur der Datenbank

Als Datenbank wird ein Relationale Datenbankmodell verwendet (Microsoft SQL-Server). Es basiert, das auf der Speicherung von Informationen in verschiedenen Tabellen, die untereinander über Beziehungen (Relationen) verknüpft sind. Bei dem Datenbankschema handelt es sich um einen Database Snapshot aus dem Microsoft SSMS-Studio. Die Darstellung der Keys und Indexes sind leider nicht optimal dargestellt (Schlüssel Feld zu Schlüssel Feld). Wobei dieses Feature des SSMS-Studios seit der Version 18.0 nicht mehr stabil interagiert. Zur besseren Veranschaulichung wird auf die Beziehungstabelle im Anschluss verwiesen.

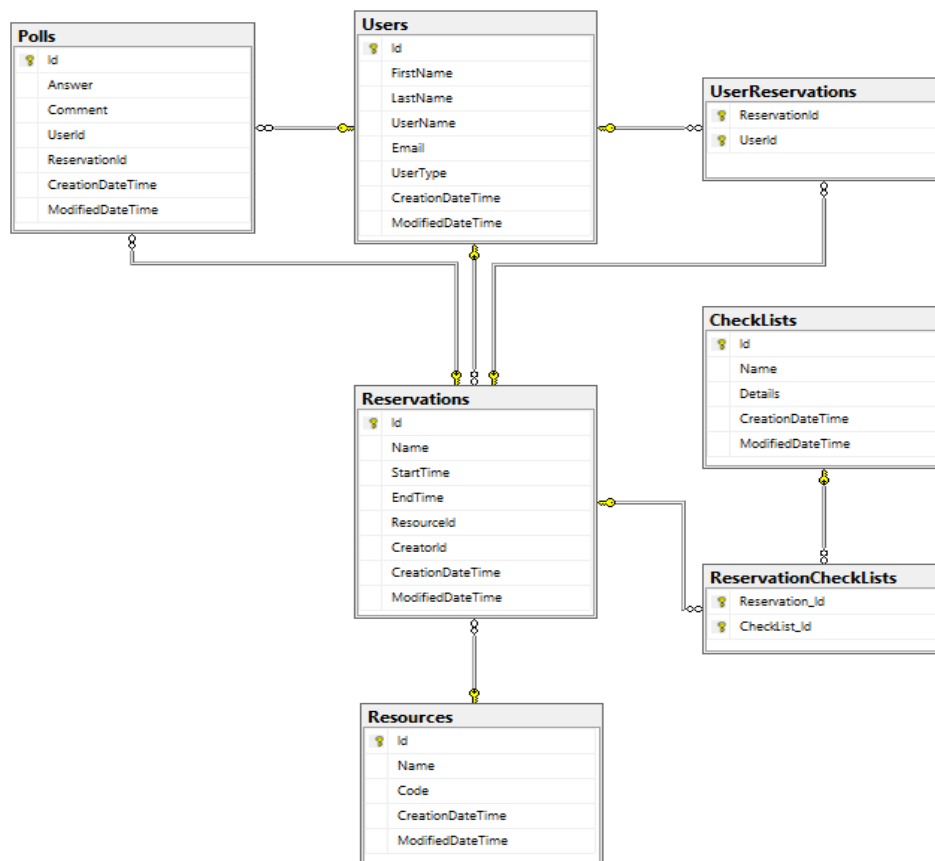


Abbildung 6: Tabellen und Beziehungen in der Datenbank

Tabelle	Primärschlüssel	Fremdschlüssel
Reservations	Id	ResourceId CreatorId
Users	Id	
Resources	Id	
Polls	Id	UserId ReservationId
CheckLists	Id	
ReservationCheckLists	Reservation_Id CheckList_Id	
UserReservations	ReservationId UserId	

Abbildung 7: Tabellen und Beziehungen in der Datenbank

3.2. Benutzeroberfläche

Die Benutzeroberfläche, GUI, des Programmes bietet ein intuitiv und einfach zu handhabendes User-Interface.

Planung der Benutzeroberfläche erfolgte über den Einsatz von Use Case-Diagrammen (Aktivitätsdiagrammen). Ein Werkzeug aus der UML zur Entwicklung von Interaktionen zwischen Akteuren und Anwendungen.

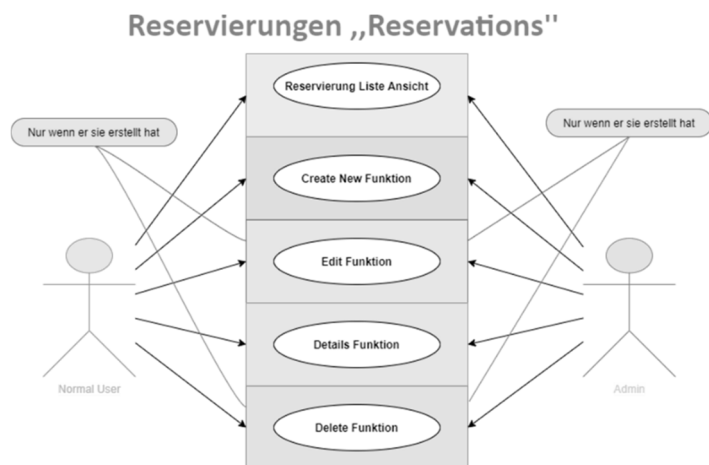
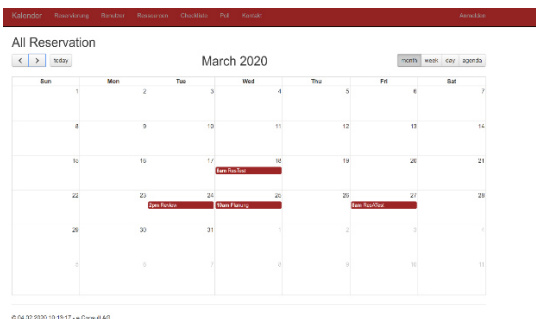


Abbildung 8 Use-Case-Reservation

3.2.1. Verhalten der Benutzeroberfläche



© 04.30.2020 18:15:17 - e.Consult AG

Abbildung 9: Kalenderansicht in der Benutzeroberfläche

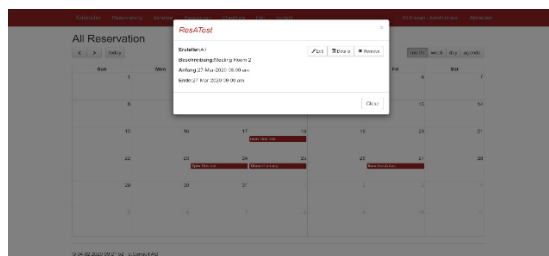


Abbildung 10: Dialogansicht zu einer eingetragenen Reservierung

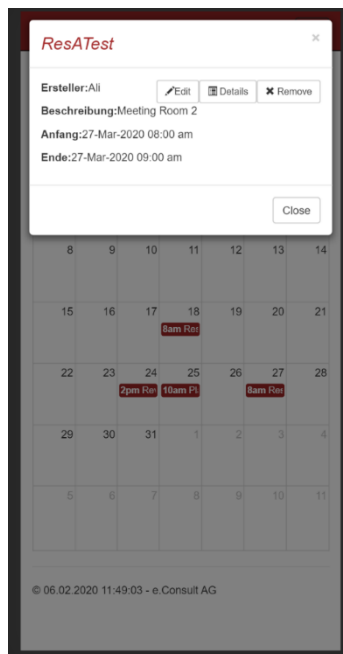
Hier sehen wir die Kalenderansicht der Kalenderoberfläche mit allen eingetragenen Events, jeweils Monatsweise durchzublättern.

Wird ein Event [Reservierung] ausgewählt, öffnet sich ein Dialog, in welchem der Name des Events, der Ersteller, die Beschreibung der Ressource, sowie deren Datum, Zeit und Zeitfenster angezeigt wird. Die Buttons, Edit, Details und Remove sind verfügbar.

Wird Edit ausgewählt, öffnet sich ein Formular zur Editierung des Events [Reservierung]

Abbildung 11: Editieransicht zu einer Reservierung

3.2.2. Benutzeroberfläche mobilen Ansicht



Dies ist die Benutzeransicht der Mobilansicht, welche einfach gehalten, alle notwendigen Funktionen und Infos anbietet, sowie Edit, Details und Remove Button

Abbildung 12: Dialogansicht zu einer eingetragenen Reservierung

Eine Vollständige Darstellung der Dialoge ihrer Funktionalitäten und den entsprechenden Use-Cases befindet sich in der (Kundendokumentation) unter Punkt 5 Technische Dokumentation im Anschluss an die Projektdokumentation.

3.3. Programmiersprache

Dieses Projekt wurde mit Visual Studio Professional 2015 von Microsoft realisiert. Zur Umsetzung einer mobilen Weboberfläche wurde das Framework ASP.NET MVC benutzt. Alle Entwickler der Firma e.Consult haben Erfahrung mit ASP.NET und C#, was zukünftige Erweiterungen durch die Kollegen ermöglicht.

C# bietet vollständige Unterstützung für objektorientierte Programmierung, einschließlich Kapselung, Vererbung und Polymorphie. Kapselung bedeutet, dass eine Gruppe verwandter Eigenschaften, Methoden sowie anderer Nutzer als eine Einheit bzw. ein Objekt behandelt wird. Vererbung beschreibt die Fähigkeit, neue Klassen basierend auf einer vorhandenen Klasse zu erstellen.

3.3.1. Polymorphismus Beispiel [7.1-Technische Dokumentation]

Polymorphismus bedeutet, dass Sie mehrere Klassen untereinander austauschen können, obwohl jede Klasse dieselben Eigenschaften oder Methoden auf unterschiedliche Art und Weise implementiert.

```
public class TestPolymorphie
{
    public int Add(int a, int b, int c)
    {
        return a + b + c;
    }
    public int Add(int a, int b)
    {
        return a + b;
    }
}

class Program
{
    static void Main(string[] args)
    {
        TestPolymorphie testClass = new TestPolymorphie();
        int add2 = testClass.Add(45, 34, 67);
        int add1 = testClass.Add(23, 34);
    }
}
```

Abbildung 13: Polymorphismus Beispiel

3.3.2. Razor [7.1 Technische Dokumentation]

ASP.NET Razor View Engine (Razor) ist die in ASP.NET verwendete Sprache für Webseiten, bei der man C# in HTML-Seiten einbetten kann. Programmblöcke werden hierbei mit dem Zeichen @ eingeleitet (im Gegensatz zur klassischen ASP.NET View Engine, die die Codeblöcke in <% %> einrahmen muss).[1]

```
@{
    ViewBag.Title = "Home Page";
    var ressourceId = TempData["RessourceId"];
    var ressourceName = ViewBag.RessourceName;
}
@{
    int Id = Convert.ToInt32(ressourceId);
    if (Id == -1) {
        string reservationname = "Alle Reservierungen";
        <h2>
            @reservationname
        </h2>
    }
    else
    {
        <h1>
            @ressourceName
        </h1>
    }
}
```

Abbildung 14: Quellcodebeispiel Razor

3.3.3. JavaScript [5- Technische Dokumentation]

Für die Interaktion des Benutzers mit dem Kalender kommt AJAX (Asynchrones JavaScript and XML) zum Einsatz. Hierdurch lässt sich die Webseite aktualisieren, ohne dass ein kompletter Pageload notwendig ist. Dadurch reagiert die Anwendung schneller und ist somit benutzerfreundlicher.

```
<div id="calender"></div>
<script>
    $(document).ready(function () {
        var events = [];
        var selectedEvent = null;
        var id = @ressourceId;

        $.ajax({
            beforeSend: function(request) {
                request.setRequestHeader("ApiKey", '@ConfigurationManager.AppSettings["ValidApiKey"]');
            },
            type: "GET", url:
                @ConfigurationManager.AppSettings["host"] + "/api/ReservationApi/" + id,
            success: function (data) {
                $.each(data.Data, function (i, v) {
                    events.push({
                        eventID: v.EventID,
                        title: v.Subject,
                        description: v.Description,
                        creator: v.Creator,
                        start: moment(v.Start),
                        end: v.End != null ? moment(v.End) : null,
                        color: v.ThemeColor,
                        allDay: v.IsFullDay
                    })
                })
                GenerateCalender(events);
            }
        });
    });
</script>
```

Abbildung 15: Quellcodebeispiel JavaScript

3.3.4. Externe Komponente [5- Technische Dokumentation]

Es wurde für dieses Projekt ein JavaScript-Veranstaltungskalender mit FullCalendar.io benutzt. Dieses MIT-lizenzierte FullCalendar-Projekt, stellt eine flexible Möglichkeit dar, damit die Ereignissammlung an die FullCalendar-Initialisierungsfunktion übergeben wird, um sie in einem in die Website eingebetteten Monatskalender anzuzeigen

```
<script src="//cdnjs.cloudflare.com/ajax/libs/fullcalendar/3.4.0/fullcalendar.min.js"></script>
```

Abbildung 16: Einbettung des FullCalendar-Objects

3.4. Technische Dokumentation

Für eingehendere Informationen über Datenbankaufbau, Programmfunktionen, eingesetzte Technologien wird an dieser Stelle auf die Kundendokumentation (Technische Dokumentation) im Anschluss an die Projektdokumentation verwiesen. Dort befinden sich die entsprechenden Erläuterungen zu den Quellcodes, Use-Cases etc.

4. Projektabschluss

4.1. Qualitätssicherung

Als Maßnahmen zur Qualitätssicherung werden verschiedene Tests durchgeführt und es wird auch ein Testprotokoll geführt.

4.1.1. Test

Für den Test wird ein separater SQL-Server benutzt, auf dem eine abgeänderte Version der Produktivdatenbank zur Verfügung gestellt wird. So ist gewährleistet, dass der Test keine Störungen im laufenden Betrieb verursachen kann.

4.1.2. Funktionale Tests

Es werden Unit Tests geschrieben, um das fehlerfreie Arbeiten der genutzten Klassen und Funktionen zu gewährleisten. Diese Tests bewegen sich in erster Linie auf der Ebene der Controller des verwendeten MVC-Frameworks.

4.1.3. Nicht-funktionale Tests

In der laufenden Anwendung werden (ohne Kenntnis des Programmcodes) alle möglichen Benutzer-Eingaben durchgeführt. Alle Ergebnisse werden einer Fehler- und Plausibilitätsprüfung unterzogen. Es werden auch Tests zur Security der Anwendung durchgeführt, um u.a. sicherzustellen, dass das Benutzer- und Rollensystem korrekt funktioniert.

4.1.4. Testprotokoll

Der Verantwortliche für die Software-Test der Unternehmung "Herr Hossain (Senior Softwareentwickler)" überprüfte das Testprotokoll. Dieses Verfahren zur Überprüfung der Richtigkeit und Gültigkeit aller Benutzereingaben soll vorab, Fehler während der Programmentwicklung sicher erkennen und dadurch.

Testobjekte:

- ✓ Kalender
- ✓ Authentifizierung
- ✓ Reservierung
- ✓ Benutzer
- ✓ Ressourcen
- ✓ Checkliste
- ✓ Poll
- ✓ Mobile Ansicht

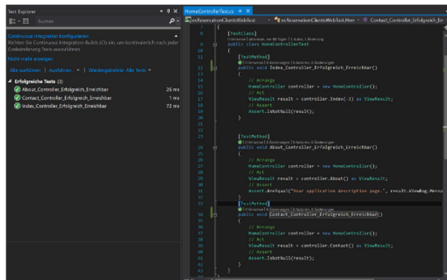
	SITU.	
6	Mit klicken des „Create New“ Button öffnet die Seite „Create Ressource“ und die neue Ressource kann hier erstellt werden.	✓
F	Checkliste	
1	Mit klicken des „Checkliste“ Button öffnet eine Seite, in der kann man alle Checklisten sehen, diese Seite enthält der Name der Checkliste und Einzelheiten über dieser Checkliste, sowie die Buttons „Edit“, „Details“, „Delete“.	✓
2	Mit klicken des „Edit“ Button öffnet die Seite „Edit Checkliste“ und die Checkliste kann bearbeitet werden.	✓

Abbildung 17: Ausschnitt aus dem Testprotokoll

4.1.5. Feedback

Aufgetretene Fehler beim Blackbox-Test durch Herrn Hossain wurden via E-Mail geschickt, um zeitnah behoben zu werden.

4.1.6. Unittests



Das Projekt hat Unittest Methoden in HomeControllerTest.cs definiert. Diese überprüfen, ob der Code wie erwartet funktioniert, indem Komponententests ausgeführt werden. Diese Tests werden als „Unittests“ bezeichnet, da Sie die Funktionalität des Programms, in einzelnes testfähiges

Verhalten gliedert, die man als einzelne Komponenten testen kann. Mit dem Test-Explorer von Visual Studio kann man Komponententests flexibel und effizient ausführen und die Ergebnisse in Visual Studio anzeigen.

Komponententests dienen der Qualitätssicherung des Codes, da sie ein integraler Bestandteil des Softwareentwicklungsworkflows sind. So wird die Eingabe von Standarddaten, falschen Daten und Daten an der Grenze des Gültigkeitsbereichs überprüft. Zudem bieten die Tests die Möglichkeit, alle im Code enthaltenen expliziten oder impliziten Annahmen zu überprüfen.

4.2. Aufgetretene Probleme & Maßnahmen

Die erste Implementierung der Authentifizierung wurde ohne Aktive Directory durchgeführt. Deshalb wurde in der Datenbank ein Feld für die Passwörter in der Tabelle des Users angelegt. Im Anschluss wurde erst die Authentifizierung durch das Aktive Directory implementiert, sodass die Speicherung der Passwörter in der Anwendung nicht mehr notwendig war. Dementsprechend wurde das Feld wieder entfernt.

Es gab ein Problem mit dem Löschen einer Reservierung oder eines Benutzers aufgrund des Foreign-Keys für Teilnehmer und Polls. So konnte ein Benutzer die Reservierung nicht löschen, weil ihr bereits ein oder mehrere Teilnehmer zugeordnet war. Jeder Teilnehmer war wiederum mit Polls verbunden. Dasselbe Problem trat auch beim Löschen von Benutzern auf, wenn sie Teilnehmer oder Ersteller einer Reservierung waren. Das sind also typische Probleme, welche beim kaskadierenden Löschen von Objekten auftreten.

Als Lösung wurde eine Klasse „ReservationService“ erstellt, damit zunächst die Polls der Teilnehmer, dann die Zuordnungen zu den Teilnehmern und schließlich die Reservierung in der Datenbank gelöscht werden. Analog dazu wurde auch eine Klasse „UserService“ erstellt, um Benutzer sauber aus dem System zu löschen, indem zunächst alle mit dem Benutzer verbundenen Objektbeziehungen entfernt werden.

4.3. Fazit und Ausblick

Das Projektziel wurde erreicht. Die beschriebenen Anforderungen in dieser Projektdokumentation wurden erfolgreich umgesetzt und abgeschlossen. Ein kleines Lob konnte ich mir bereits für die gelungene Umsetzung der verschiedenen Module abholen. Die Ressourcenreservierungsverwaltung fällt dank meines Projektes nun erheblich leichter. Die neue Oberfläche ist außerdem gut geeignet, um weitere Funktionen zu ergänzen.

4.3.1. Zukünftige Anforderung

Es ist später auch möglich den Zugriff auf die Informationen zu Ressourcen zu erleichtern, indem der an die Ressource angehängte QR-Code gescannt wird.

5. Technische Dokumentation

6. Benutzerdokumentation

7. Quellenverzeichnis

7.1. Literatur

[1] Community for Developers and IT Professionals.

<https://www.c-sharpcorner.com/UploadFile/ff2f08/understanding-polymorphism-in-C-Sharp/>

[2] IT-Vision Dr. Holger Schwichtenberg

<https://www.it-visions.de/lserver/artikeldetails.aspx?b=7299>

[3] Externe Komponente fullcalendar.io

<https://fullcalendar.io/>

<https://github.com/fullcalendar/fullcalendar/releases/download/v4.3.1/fullcalendar-4.3.1.zip>

7.2. Weblinks

Galileo Computing - Objektorientierte Programmierung

http://openbook.rheinwerk-verlag.de/oop/oop_kapitel_01_001.htm

Abruf: 07.04.2014 - 15:55

Microsoft Developer Network -OOP C#

<https://docs.microsoft.com/de-de/dotnet/csharp/programming-guide/concepts/object-oriented-programming>

Abruf: 07.04.2014 - 16:04

Wikipedia - die freie Enzyklopädie

[https://de.wikipedia.org/wiki/Datenkapselung_\(Programmierung\)](https://de.wikipedia.org/wiki/Datenkapselung_(Programmierung))

Abruf: 08.04.2014 - 13:28

7.3. Abbildungen

Tabelle: Schnittstellen 1	1-2
Abbildung 2: Aushang an der Tür des Konferenzraumes	2-3
Abbildung 3: Projektablauf mit den einzelnen Projektschritten	2-5
Abbildung 4: Zeitplanung	2-6
Abbildung: 5 Kostenplan	2-6
Abbildung 6: Tabellen und Beziehungen in der Datenbank	3-8
Abbildung 7: Tabellen und Beziehungen in der Datenbank	3-8
Abbildung 8 Use-Case-Reservation	3-9
Abbildung 9: Kalenderansicht in der Benutzeroberfläche	3-9
Abbildung 10: Dialogansicht zu einer eingetragenen Reservierung	3-9
Abbildung 11: Editieransicht zu einer Reservierung	3-10
Abbildung 12: Dialogansicht zu einer eingetragenen Reservierung	3-10
Abbildung 13: Polymorphismus Beispiel	3-11
Abbildung 14: Quellcodebeispiel Razor	3-12
Abbildung 15: Quellcodebeispiel JavaScript	3-12
Abbildung 16: Einbettung des FullCalender-Objects	3-13
Abbildung 17: Ausschnitt aus dem Testprotokoll	4-14

7.4. Testprotokoll

Nr.	Welche Funktionalität wird getestet? Erwartetes Ergebnis - Bemerkungen	OK	Nicht OK
A	Kalender		
1	Alle Mitarbeiter können alle Reservierungen ohne Anmeldung sehen	✓	
2	Durch Klicken der „Vor“- und „Zurück“-Buttons (< >) kann zu einem anderen Monat gewechselt werden.	✓	
3	Durch Klicken des „Today“-Buttons kann man zum aktuellen Monat wechseln	✓	
4	Durch Klicken des „Month“-Buttons kann man die monatliche Kalender Ansicht sehen.	✓	
5	Durch Klicken des „Week“-Buttons kann man die wöchentliche Kalender Ansicht sehen.	✓	
6	Durch Klicken des „Day“-Buttons kann man die tägliche Kalender Ansicht sehen.	✓	
7	Durch Klicken des „Agenda“-Buttons kann man die stündliche Kalender Ansicht sehen.	✓	
8	Durch Klicken auf eine Veranstaltung öffnet sich ein Dialog, in welchem der Name der Veranstaltung, der Ersteller, die Ressource, die Terminierung, sowie die Buttons „Edit“, „Details“, und „Remove“ zu finden sind.	✓	
9	Durch Klicken auf ein Datumsfeld öffnet sich ein Dialog zum Erstellen eines neuen Eintrages.	✓	
B	Authentifizierung		
1	Die Anmeldung muss durch das Active-Directory des Unternehmens authentifiziert werden.	✓	
C	Reservierung		
1	Mit klicken des „Reservierung“ Button kann man alle Reservierung sortiert nach Erstellungsdatum sehen, in welchem der Name der Veranstaltung, der Ersteller, die Ressource, die Terminierung zu finden sind.	✓	
2	Mit klicken des „Edit“ Button von - Ersteller der Reservierung - öffnet die Seite „Edit Reservation“ und die Reservierung kann bearbeitet werden.	✓	
3	Mit klicken des „Details“ Button öffnet eine Seite, in welchem der Name der Veranstaltung, der Ersteller, die Ressource, die Terminierung, Datum und Uhrzeit der erstellen, Datum und Uhrzeit der Änderung, sowie Checkliste und die Teilnehmer mit ihrer Antwort zu finden sind.	✓	
4	Mit klicken des „Delete“ Button - von Ersteller der Reservierung - öffnet die Seite „Delete Reservation“ und die Reservierung kann gelöscht werden.	✓	
D	Benutzer		
1	Mit klicken des „Benutzer“ Button kann der Benutzer „nur mit Admin Rolle“ die Infos der Benutzer sortiert nach alphabetisch in Benutzer Seite sehen, in welchem der First Name, der Last Name, die User Name, die E-Mail und Benutzer Rolle zu finden sind.	✓	
2	Mit klicken des „Edit“ Button in der Benutzer Seite öffnet eine Seite, die nur die Rolle dem Benutzer enthält, und die Rolle kann bearbeitet werden. Wenn Sie jedoch auf den Namen des aktuellen Benutzers klicken, enthält die Seite alle zugehörigen Informationen, und die Info kann bearbeitet werden.	✓	
3	Mit klicken des „Details“ Button öffnet eine Seite, in welchem der First Name dem Benutzer, die Last Name, die User Name, die E-Mail, die Rolle und Uhrzeit der erstellen, Datum und Uhrzeit der Änderung zu finden sind.	✓	
4	Mit klicken des „Delete“ Button öffnet die Seite „Delete User“ und den Benutzer kann gelöscht werden.	✓	
E	Ressourcen		

1	Mit klicken des „Ressourcen“ Button öffnet eine Seite, in der kann man alle Ressourcen sehen, diese Seite enthält der Name und dem Code der Ressource, sowie die Buttons „Edit“, „Details“, „Delete“ und „Calander“.	✓	
2	Mit klicken des „Edit“ Button öffnet die Seite „Edit Ressource“ und die Ressource kann bearbeitet werden.	✓	
3	Mit klicken des „Details“ Button öffnet eine Seite, in welchem der Name der Ressource, Datum und Uhrzeit der erstellen, Datum und Uhrzeit der Änderung zu finden sind.	✓	
4	Mit klicken des „Delete“ Button öffnet die Seite „Delete Ressource“ und die Ressource kann gelöscht werden.	✓	
5	Mit klicken des „Calander“ Button öffnet die Hauptseite „Vollständig Calander“, in welchem kann man sehen nur die Reservierung für zu dieser Ressource gehören sind.	✓	
6	Mit klicken des „Create New“ Button öffnet die Seite „Create Ressource“ und die neue Ressource kann hier erstellt werden.	✓	
F	Checkliste		
1	Mit klicken des „Checkliste“ Button öffnet eine Seite, in der kann man alle Checklisten sehen, diese Seite enthält der Name der Checkliste und Einzelheiten über dieser Checkliste, sowie die Buttons „Edit“, „Details“, „Delete“.	✓	
2	Mit klicken des „Edit“ Button öffnet die Seite „Edit Checkliste“ und die Checkliste kann bearbeitet werden.	✓	
3	Mit klicken des „Details“ Button öffnet eine Seite, in welchem der Name der Checkliste, die Einzelheiten der Checkliste, Datum und Uhrzeit der erstellen, Datum und Uhrzeit der Änderung zu finden sind.	✓	
4	Mit klicken des „Delete“ Button öffnet die Seite „Delete Checkliste“ und die Checkliste kann gelöscht werden.	✓	
5	Mit klicken des „Create New“ Button öffnet die Seite „Create Checkliste“ und die neue Checkliste kann hier erstellt werden.	✓	
G	Poll		
1	Mit klicken des „Poll“ Button öffnet eine Seite, in dieser Seite kann man sehen alle Polls gehören zu ihm sortiert nach Erstellungsdatum, diese Seite enthält der Name der Reservierung, der Ersteller der Reservierung, Antwort dem Benutzer, Kommentar dem Benutzer, Datum und Uhrzeit der erstellen, Datum und Uhrzeit der Änderung, sowie die Buttons „Edit“, „Details“. Wenn der Benutzer noch nicht Teilnehmer in einer Reservation ist, findet den Benutzer keine Poll in der Seite aber mit Alert, dass er noch keine Poll hat.	✓	
2	Mit klicken des „Edit“ Button öffnet die Seite „Edit Poll“ und die Antwort und Kommentar über dieser Reservierung kann bearbeitet werden.	✓	
3	Mit klicken des „Details“ Button öffnet eine Seite, in welchem der Name der Reservierung, der Ersteller der Reservierung, Antwort der Benutzer (Teilnehmer), der Kommentar der Benutzer, Datum und Uhrzeit der erstellen, Datum und Uhrzeit der Änderung zu finden sind.	✓	
H	Mobile Ansicht		
	A -G wird mit dem Mobilgerät wiederholt	✓	

7.5. Zeitplanung

Projektphasen	Stunden	Gesamt
Analyse		8
Soll / Ist-Analyse	4	
Funktionen und Anforderungen der DB-Anwendung		
Schnittstellen / Abgrenzungen		
Definition von Zielen, Qualität, Kosten, Zeitbedarf	2	
Erstellung und Besprechung Pflichtenheft, Projektauftrag	2	
Planung		7
Aufgaben- und Ablaufplan	2	
Zeit- und Ressourcenplan	2	
Qualität und Lieferobjekte	3	
Entwurf		7
Objekte, Attribute und Datentypen	2	
ER-Modell und Tabellen	2	
Benutzeroberfläche	3	
Realisierung		30
Backend entwickeln		18
Tabellen und Beziehungen	2	
Attribute und Datentypen	2	
Controllers implementieren	4	
Services implementieren	4	
Serverseitige Validierung Eingabedaten	4	
Rest API programmieren	2	
Frontend entwickeln		12
Startseite, Schaltflächen	2	
Seiten zum vollständigen Kalendar, Reservation, Users, Ressource, Checklist, Polls	6	
Externe Komponente Integration	4	
Testphase		12
Funktionstest der Benutzeroberflächen	3	
Sicherheitstests	3	
Unit Test	6	
Integration		4
Deployment der Webanwendung	2	
Einrichtung der Datenbank	1	
Einweisung	1	
Abnahme		2
Gesamt		70

7.6. Pflichtenheft