



# Programmentwurf - Hochzeitsplaner

---

des Studiengangs Angewandte Informatik  
an der Dualen Hochschule Baden-Württemberg Karlsruhe

von

**Slavisa Markovic und Fabian Fellhauer**

Matrikelnummer	6605872 & 7068712
Vorlesung	Softwareengineering
Dozent	Dr.-Ing. Richard Lutz

**4/1/2016**

---

# Inhaltsverzeichnis

1. Aufgabenstellung.....	4
1.1. Einleitung.....	4
1.2. Lastenheft.....	4
1.2.1. Zielsetzung.....	4
1.2.2. Anwendungsbereiche.....	4
1.2.3. Zielgruppen, Benutzerrollen und Verantwortlichkeiten .....	4
1.2.4. Zusammenspiel mit anderen Systemen .....	4
1.2.5. Produktfunktionen .....	5
1.2.6. Produktdaten.....	5
1.2.7. Produktleistungen .....	6
1.2.8. Qualitätsanforderung .....	6
1.3. Vereinfachungen für den Programmentwurf.....	6
2. Analyse .....	7
2.1. Einleitung.....	7
2.2. Lastenheft.....	7
2.2.1. Zielsetzung.....	7
2.2.2. Anwendungsbereiche.....	14
2.2.3. Zielgruppen, Benutzerrollen und Verantwortlichkeiten .....	14
2.2.4. Zusammenspiel mit anderen Systemen .....	16
2.2.5. Produktfunktionen .....	17
2.2.6. Produktdaten.....	28
2.2.7. Produktleistungen .....	29
2.2.8. Qualitätsanforderung .....	30
2.3. Vereinfachungen für den Programmentwurf.....	30
3. Analyseklassendiagramm .....	31
3.1. Analyse der verschiedenen Objekte.....	31
3.2. Verwendete Analysemuster .....	39
3.2.1. Liste .....	39
3.2.2. Baugruppe .....	39
3.2.3. Rollen.....	39
3.2.4. Gruppe.....	40
4. Use-Case-Diagramm .....	41

4.1.	Rollenerläuterung der Anwendung .....	41
4.1.1.	Standard-Nutzer (Pseudorolle).....	41
4.1.2.	Hochzeitsmanager .....	41
4.1.3.	Unterhaltungsmanager .....	42
4.1.4.	Administrator .....	42
4.2.	Use-Case-Kompaktansicht.....	42
4.3.	Use-Case Diagramm: Aktivität bearbeiten .....	46
4.4.	Use-Case Diagramm: Suchen.....	48
5.	Sequenzdiagramm.....	50
5.1.	Szenariobetrachtung: Catering anlegen.....	50
5.2.	Diagrammbetrachtung: Catering anlegen.....	51
5.3.	Pseudocode: Catering anlegen.....	53
5.4.	Szenariobetrachtung: Essen anlegen .....	57
5.5.	Diagrammbetrachtung: Essen anlegen .....	57
6.	Aktivitätsdiagramm .....	59
6.1.	Szenariobetrachtung: Unterhaltsbeitrag anlegen.....	59
6.2.	Diagrammbetrachtung: Unterhaltsbeitrag anlegen.....	60
6.3.	Pseudocode: Unterhaltsbeitrag anlegen.....	62
7.	Entwurfsklassendiagramm .....	64
7.1.	Pakete.....	64
7.1.1.	Paket Modellklassen.....	64
7.1.2.	GUI.....	65
7.1.3.	UIEventInterfaces .....	66
7.1.4.	ExceptionHandling.....	67
7.1.5.	Util .....	67
7.1.6.	Kern .....	68
7.1.7.	Systemverwaltung .....	68
7.1.8.	Datenkonvertierung .....	68
7.1.9.	Datenbankverbindung.....	69
7.2.	Verwendete Entwurfsmuster .....	69
7.2.1.	Singleton.....	69
7.2.2.	Lazy Loading .....	70
7.2.3.	Fassade .....	70
8.	GUI-Entwurf.....	71

8.1.	Hochzeitsübersicht .....	72
8.2.	Aktivitätensicht.....	73
8.3.	Detailansicht.....	74
9.	Relationales Datenbank Mapping .....	75
10.	Besonderheiten .....	77
10.1.	Backups.....	77
10.2.	Import & Export.....	77
10.3.	Übersicht der Aktivitäten im Kalender .....	77
10.4.	E-Mail Versand über die Software.....	77
10.5.	Globaler Server für alle Benutzer .....	77

# 1. Aufgabenstellung

## 1.1. Einleitung

Wir die EMSIG GmbH (Event Management Schulze Irrwisch Gimpel GmbH) sind ein führendes mittelständisches Unternehmen für die Planung und Durchführung mittelgroßer Veranstaltungen (ca. 100 -1000 Teilnehmern). Hierfür setzen wir seit Jahren ein bewährtes Softwarewerkzeug ein.

Speziell für die Planung von Hochzeiten benötigen wir ein neues Werkzeug, welches zum einen für unsere eigene Firma eingesetzt und zum anderen auch für Privatpersonen als günstige Planungssoftware angeboten werden soll.

## 1.2. Lastenheft

### 1.2.1. Zielsetzung

Ziel des Entwicklungsauftrags soll eine Software für die Verwaltung von Hochzeiten sein. Dabei soll auf eine zentrale Datenbasis zugegriffen werden können (Server), damit sämtliche Daten von mehreren PCs und Laptops aus verwaltet werden können. Daneben sollen mehrere Personen gemeinsam an der Hochzeitsplanung teilnehmen können.

Ein Import und Export ausgewählter Daten muss zur besseren Wiederverwendbarkeit, für Backups und zum Datenaustausch möglich sein.

Eine intuitive, leicht bedienbare Benutzeroberfläche setzen wir als selbstverständlich voraus.

Es sollen keine besonderen Computerkenntnisse zur Bedienung der Software erforderlich sein.

### 1.2.2. Anwendungsbereiche

Die Software soll ausschließlich für die Verwaltung von Hochzeiten eingesetzt werden. Sie soll bei uns in der Firma im Tagesgeschäft eingesetzt werden sowie von Privatpersonen erwerbbar sein.

### 1.2.3. Zielgruppen, Benutzerrollen und Verantwortlichkeiten

Als Zielgruppe kommen zwei Rollen infrage: die eigentliche planungsverantwortliche Person, welche auf sämtliche Daten lesend und schreibend Zugriff hat (Hochzeitsmanager).

Ausnahme: da oftmals das Brautpaar selbst planen und managen will, soll es möglich sein, die persönlichen Unterhaltungsbeiträge für die Hauptplaner zu verstecken. Hierfür soll es eine zweite Rolle geben, die lesenden Zugriff auf die grundlegenden Hochzeitsdaten hat (Zeiten, Datumsangaben, Orte, ... ), ansonsten aber ausschließlich die Unterhaltungsbeiträge verwalten kann (Unterhaltungsmanager).

### 1.2.4. Zusammenspiel mit anderen Systemen

Das zu entwickelnde Softwaresystem soll auch ohne Netzverbindung lauffähig sein. Hierzu sollen sämtliche Daten einer Hochzeit lokal gespeichert und auf Wunsch des Benutzers mit den Serverdaten synchronisiert werden können.

### 1.2.5. Produktfunktionen

- /LF10/ Eine Zugangsberechtigung soll mittels eines einfachen Loginvorgangs verifiziert werden.  
Der Zugriff auf einzelne Daten soll je nach Berechtigung unterschiedlich erfolgen.  
→ siehe Abschnitt 2.3: „Zielgruppen, Benutzerrollen und Verantwortlichkeiten“
- /LF20/ Der jeweilige Benutzer muss die Möglichkeit haben, über eine grafische Benutzeroberfläche (GUI) alle für ihn relevanten Daten einfach und übersichtlich verwalten zu können.
- /LF30/ Eine Hochzeitsveranstaltung fasst viele Einzelaktionen zusammen: angefangen von der standesamtlichen und kirchlichen Trauung, Buchung bzw. Reservierung von Veranstaltungsorten, Catering (Essen und Getränke getrennt verwaltbar) über Erstellung und/oder Druck von Einladungen, Tischkarten, diverse Besorgungen, Organisation und Buchung von Übernachtungen und dem Hochzeitsfahrzeug bis zur Organisation und Durchführung der Dekoration (in Standesamt, Kirche und Festsaal) uvm.
- /LF40/ Jede Aktion beginnt und endet zu einem bestimmten Zeitpunkt, es müssen verantwortliche Personen und Teilnehmer der Aktion benannt werden können. Die verantwortlichen Personen und Teilnehmer sollen automatisch per E-Mail benachrichtigt werden können (z.B. durch starten eines vorhandenen Mail-Tools über die grafische Benutzeroberfläche).  
  
Jede Aktion kann an mehreren Orten stattfinden, mit Hilfsmitteln aus einer Liste durchgeführt werden. Anfallende Kosten (Rechnungen, Belege) sollen jeweils mit angegeben werden können.  
  
Für den Benutzer soll es leicht möglich sein, die aktuellen Zustände aller Aktionen zu erkennen (geplant, in Arbeit, abgeschlossen, usw.) und zu ändern. Der Benutzer soll geeignet dabei unterstützt werden, bestimmte vorgegebene Aktionsarten anlegen und durchführen zu können. Dabei soll der Benutzer einfach erkennen können, welche Aktionen bereits angelegt sind und welche noch nicht.  
  
Es soll darüber hinaus möglich sein, einer Aktion verschiedene Medien (Dokumente, Bilder, Videos, usw.) zuzuordnen.
- /LF50/ Jede verantwortliche Person und jeder Teilnehmer kann bei den üblichen Kontaktdaten mehrere E-Mail-Adressen und mehrere Telefonnummern besitzen.
- /LF60/ Das Catering kann entweder von einem kommerziellen Catering-Service als auch von ausgewählten Personen durchgeführt werden.
- /LF70/ Die oben erwähnte Liste der Hilfsmittel soll auf einfache Weise erweiterbar und zuweisbar sein. Sie sollen für sämtliche Hochzeitsveranstaltungen im System verfügbar sein.
- /LF80/ Zur Kostenkontrolle soll es möglich sein, sämtliche bisher angefallenen Kosten auf einfache Weise addieren zu können. Zur Kostenabschätzung sollen auch geschätzte Kosten angegeben und addiert werden können.
- /LF90/ Die Auswahl der Daten soll möglichst über (eventuell durchsuchbare) Auswahllisten erfolgen. Dies gilt vor allem für Zuordnungen von ausgewählten Personen zu den Aktionen usw.
- /LF100/ Vor dem Hinzufügen von neuen Daten soll eine Überprüfung stattfinden, ob diese eventuell schon vorhanden sind. Das gilt in besonderem Maße für Personen und Aktionen.

### 1.2.6. Produktdaten

- /LD10/ Die Daten sollen zentral verwaltet und in einer Datenbank abgespeichert werden.

### 1.2.7. Produktleistungen

- /LL10/ Das Laden gewünschter Daten soll für eine sinnvolle Benutzung im Sekundenbereich erfolgen.
- /LL20/ Die Anzahl der zu verwaltenden Elemente wird auf ca. 100000 geschätzt.
- /LL30/ Die Daten müssen bei unserer eigenen Verwendung aus rechtlichen Gründen 10 Jahre online verfügbar sein.
- /LL40/ Um bei Anschaffungen und Neuerungen flexibel zu bleiben, ist auf Plattformunabhängigkeit besonders zu achten.

### 1.2.8. Qualitätsanforderung

Produktqualität	sehr gut	gut	normal	nicht relevant
Funktionalität	X			
Zuverlässigkeit		X		
Effizienz			X	
Benutzbarkeit	X			
Änderbarkeit			X	
Übertragbarkeit			X	
Gestaltung	X			

## 1.3. Vereinfachungen für den Programmentwurf

1. Es muss nicht dafür gesorgt werden, dass auf dieselben Daten der Datenbank nicht gleichzeitig zugegriffen werden kann, d.h. es ist kein Locking-Mechanismus erforderlich.
2. Eine Protokollierfunktion ist für die Anwendung nicht erforderlich (in der Realität natürlich schon!).
3. Ein Loginvorgang und eine Benutzerverwaltung müssen in den Klassendiagrammen nicht modelliert und später auch nicht implementiert werden
4. Die Synchronisation der Daten muss bei der Implementierung nicht berücksichtigt werden.

P.S.: Kopieren Sie den Aufgabentext (d.h. ohne Frontseite) vollständig als erstes Kapitel „Aufgabenstellung“ an den Anfang Ihrer PE-Dokumentation und verwenden Sie den Aufgabentext zusätzlich als Rahmen für Ihre Lastenheftanalyse („ausfüllen“ mit Fragen und Antworten)!

## 2. Analyse

**WICHTIGER VERMERK:** Die angeforderten LF/LL/LD wurden mit gelber Schrift, sowie Fettdruck markiert.

### 2.1. Einleitung

Wir die EMSIG GmbH (Event Management Schulze Irrwisch Gimpel GmbH) sind ein führendes mittelständisches Unternehmen für die Planung und Durchführung mittelgroßer Veranstaltungen (ca. 100 -1000 Teilnehmern). Hierfür setzen wir seit Jahren ein bewährtes Softwarewerkzeug ein.

Speziell für die Planung von Hochzeiten benötigen wir ein neues Werkzeug, welches zum einen für unsere eigene Firma eingesetzt und zum anderen auch für Privatpersonen als günstige Planungssoftware angeboten werden soll.

### 2.2. Lastenheft

#### 2.2.1. Zielsetzung

Ziel des Entwicklungsauftrags soll eine Software für die Verwaltung von Hochzeiten sein. Dabei soll auf eine zentrale Datenbasis zugegriffen werden können (Server), damit sämtliche Daten von mehreren PCs und Laptops aus verwaltet werden können. Daneben sollen mehrere Personen gemeinsam an der Hochzeitsplanung teilnehmen können.

**Handelt es sich bei der Datenbasis um eine Datenbank oder mehrere Textdateien?**

Bei der Datenbasis handelt es sich um eine Datenbank.

**Existiert bereits eine zentrale Datenbasis?**

Ja wir besitzen bereits eine Datenbank für die anderen Eventplanungen unserer Firma.

**Wenn ja, welche Datensätze existieren bereits?**

Für die Hochzeitssoftware existieren keine Datensätze.

**Welche Daten sollen konkret in der Datenbasis gespeichert werden?**

Es sollen folgende Entitäten gespeichert werden: Hochzeitprojekte, Personen, Aktivitäten, Hilfsmittel, Hilfsmittelplanungen, Caterer, Verpflegung, Getränke, Essen, Lokalität, Nutzerdaten

**Wie ist ein Hochzeitsprojekt definiert?**

Es hat folgende Attribute: Hochzeitsdatum (definiert als Datum der standesamtlichen Trauung), Braut (Datentyp Person), Bräutigam (Datentyp Person), Trauungsort (Lokalität), Aktivitäten, Hochzeits-ID (eindeutige ID), obligatorische standesamtliche Trauung und eine optionale kirchliche Trauung, bezahlte Kosten (Zahl), geplante Kosten (Zahl), gesamte Kosten (Zahl), Liste mit Nutzer (Nutzerdaten, nur für den Login, damit man weiß, welcher Nutzer in welchen Hochzeiten eingetragen ist, wird nicht näher behandelt)

**Wie sind Personen definiert?**

Haben folgende Attribute: Vorname (Text), Nachname (Text), Geburtsdatum (Datum), Personen-ID (eindeutige ID), Liste an E-Mail (E-Mail), Liste an Telefonnummern (Telefonnummer)



### Wie ist eine E-Mail definiert?

Eine E-Mail hat folgende Attribute: Adresse (typische E-Mail nach RFC 5321 und RFC 5322), Klassifizierung (Auswahl aus Werteliste: privat, geschäftlich, sonstige), ist primäre E-Mail (boolescher Wert, kann immer nur bei einer Adresse pro Person definiert sein)

### Wie ist eine Telefonnummer definiert?

Eine Telefonnummer hat folgende Attribute: Nummer (Zahl), Klassifizierung (Auswahl aus Werteliste: privat, geschäftlich, mobil, sonstige)

### Wie ist eine Aktivität definiert?

Hat folgende Attribute: Titel (Text), Startdatum (Datum), Startuhrzeit (Zeit), Enddatum (Datum), Enduhrzeit (Zeit), Liste mit Orten (Lokalität), Liste mit Hilfsmittelplanungen (Hilfsmittelplanungen), Liste mit Verpflegungen (Verpflegung), Kommentarfeld (Text), eindeutige Aktivitäts-ID (eindeutige ID), Geheime Aktivität (Boolescher Wert), Unterhaltsbeitrag (boolescher Wert), Liste mit Teilnehmer (Person), Liste zuständiger Personen (Person), Zustand der Aktivität (Werteliste mit folgenden möglichen Werten: in Arbeit, geplant, aktiv, beendet, abgebrochen), Liste mit Medien (Medium), bezahlte Kosten (Zahl), geplante Kosten (Zahl), gesamte Kosten (Zahl)

### Welche Attribute umfasst die standesamtliche Trauung?

Die standesamtliche Trauung umfasst folgende Attribute: Trauungsdatum (Datum), Uhrzeit (Zeit), Braut (Datentyp Person), Bräutigam (Datentyp Person), Name des Standesbeamten (Datentyp Person), Trauungsort (Datentyp Lokalität), Liste der Trauzeugen (Liste mit Personen), Liste mit Hilfsmittelplanungen (Hilfsmittelplanungen), Liste mit Verpflegungen (Verpflegung), Kommentarfeld (Text), eindeutige Aktivitäts-ID (eindeutige ID), Liste mit Teilnehmer (Person), Zustand der Aktivität (Werteliste mit folgenden möglichen Werten: in Arbeit, geplant, aktiv, beendet), Liste mit Medien (Medium), bezahlte Kosten (Zahl), geplante Kosten (Zahl), gesamte Kosten (Zahl)

### Welche Attribute umfasst die kirchliche Trauung?

Die kirchliche Trauung umfasst folgende Attribute: Trauungsdatum (Datum), Uhrzeit (Zeit), Braut (Datentyp Person), Bräutigam (Datentyp Person), Name des Pastors/der trauenden Person (Datentyp Person), Trauungsort (Datentyp Lokalität), Liste mit Hilfsmittelplanungen (Hilfsmittelplanungen), Liste mit Verpflegungen (Verpflegung), Kommentarfeld (Text), eindeutige Aktivitäts-ID (eindeutige ID), Liste mit Teilnehmer (Person), Zustand der Aktivität (Werteliste mit folgenden möglichen Werten: in Arbeit, geplant, aktiv, beendet), Liste mit Medien (Medium), bezahlte Kosten (Zahl), geplante Kosten (Zahl), gesamte Kosten (Zahl)

### Wie ist eine Lokalität definiert?

Hat folgende Attribute: Straße (Text), Hausnummer (Zahl), Ort (Text), Postleitzahl (Zahl), Land (Text), Titel (Text), eindeutige Lokalitäts-ID (eindeutige ID), Buchungsstatus (Werteliste mit folgenden möglichen Werten: geplant, reserviert, gebucht, storniert)

### Wie ist die Verpflegung definiert?

Hat folgende Attribute: Anlieferungsdatum (Datum), Anlieferungszeit (Zeit), Liste mit Essen, Liste mit Getränken, zuständiger Caterer, eindeutige Verpflegungs-ID (eindeutige ID), Liste mit Kosten (Kosten)

### Wie ist Essen definiert?

Es ist Teil der Verpflegung.

Hat folgende Attribute: Name des Essens (Text), Menge in Stück (Zahl), Allergene Informationen (Text), eindeutige Essens-ID (eindeutige ID), Gang (Zahl)

### Wie ist Getränk definiert?

Es ist Teil der Verpflegung.

Hat folgende Attribute: Name des Getränks (Text), Menge in Stück (Zahl), Allergene Informationen (Text), eindeutige Getränke-ID (eindeutige ID), Alkoholgehalt (Wert: ja oder nein)

Sollen Essen und Getränke bei jeder Zuweisung neu erstellt werden oder aus einer bestehenden Liste ausgewählt werden?

Sie sollen aus einer bestehenden Liste ausgewählt werden.

Dadurch kann die Menge aber nicht für dieselbe Einheit an Essen geändert werden, ohne sie gleichzeitig bei der anderen Einheit ebenfalls zu ändern. Sollen wir eine zusätzliche Essens und Getränkeplanung einführen, damit dies funktioniert?

Ja

### Wie ist der Caterer definiert?

Hat folgende Attribute: Name (Text), Ort (Lokalität), Telefonnummer (Zahl), E-Mail Adresse, Kontaktperson (Person), eindeutige Caterer-ID (eindeutige ID), ist kommerziell (boolescher Wert)

### Wie ist eine Hilfsmittelplanung definiert?

Es ist die Zuweisung von Hilfsmitteln zu einer entsprechenden Aktivität.

Hat folgende Attribute: Hilfsmittel (Hilfsmittel), zuständige Person (Person), Menge in Stück (Zahl), Anlieferungsdatum (Datum), Anlieferungszeit (Zeit), eindeutige Hilfsmittelplanungs-ID (eindeutige ID), Liste mit Kosten (Kosten)

### Wie ist ein Hilfsmittel definiert?

Es ist Teil der Hilfsmittelplanung.

Hat folgende Attribute: Titel (Text), Beschreibung (Text), eindeutige Hilfsmittel-ID (eindeutige ID)

### Wie ist ein Medium definiert?

Ein Medium hat folgende Attribute: Titel (Text), Dateipfad (Text), Datentyp (Text)

### Wie sind Kosten definiert?

Kosten haben folgende Attribute: Kostenposten (Text), Betrag (Zahl), Klassifizierung (Auswahl aus Werteliste mit folgenden Werten: geplant, bezahlt), Liste mit Belegen (Bild)

### Wie sind Nutzerdaten definiert?

Nutzerdaten haben folgende Attribute: Benutzername (Text), Passwort (Text), Rolle (Auswahl aus Werteliste: Hochzeitsmanager, Unterhaltungsmanager, Standard-Nutzer, Administrator),

### Welche Rechte hat der Standard-Nutzer?

Er kann sämtliche Standardaktionen, die das Anzeigen, Bearbeiten und Löschen von Aktivitäten umfassen, ausführen.

### Wer erhält die Standard-Nutzer Rolle?

Die Rolle des Standard-Nutzers wird einem Benutzer nie zugewiesen und wird deswegen auch als Pseudorolle bezeichnet.

**Wie sieht die eindeutige ID aus?**

Die eindeutige ID darf in allen Systemlandschaften nur einmal vorkommen und besteht aus 32 Ziffern und enthält eine Instanznummer des Systems. Die 6-stellige Instanznummer des Systems wird bei der Installation der Software vergeben und ist pro System eindeutig.

**Sollen auch die Daten der privaten Nutzer in der zentralen Datenbasis gespeichert werden?**

Ja, auch die Daten der privaten Nutzer werden in unserer Datenbank gehalten. Dies ist jedoch nicht Pflicht. Private Nutzer haben auch die Möglichkeit die Daten lokal zu speichern.

**Sollen mehrere Personen gleichzeitig Daten bearbeiten können?**

Nein, dies ist nicht erforderlich.

**Gibt es eine maximale Anzahl an Personen, die an der Hochzeitsplanung teilnehmen können?**

Nein, es können beliebig viele Personen zur Planung eingeteilt werden.

**Gibt es Personen die nur Lesezugriffe besitzen dürfen?**

Ja, der Unterhaltungsmanager hat bei den Hochzeitsstammdaten nur lesende Zugriffe.

**Wer entscheidet über das Hinzufügen eines Teilnehmers? Gibt es einen Master-User?**

Der Hochzeitsmanager bestimmt über das Hinzufügen von Teilnehmern.

Ja, es gibt einen Master-User, den Administrator.

**Können während den Planungen Personen nachträglich hinzugefügt werden?**

Ja, Personen können nachträglich zur Planung der Hochzeit vom Hochzeitsmanager und Administrator hinzugefügt werden.

Ein Import und Export ausgewählter Daten muss zur besseren Wiederverwendbarkeit, für Backups und zum Datenaustausch möglich sein.

**Falls eine alte Datenbasis existiert, sollen diese Daten importiert werden können?**

Ja es existiert eine Datenbasis, mit Daten von anderen Eventplanungs-Tools, wovon Daten importiert werden sollen.

**Welche Daten müssen im- und exportiert werden?**

Aus unseren alten Systemen wollen wir Hilfsmittel und Lokalisationen importieren.

Sämtliche Datensätze sollen aus dem neuen System exportiert werden können.

**Welches Datenformat besitzen die zu importierenden Daten?**

Sämtliche Daten aus anderen Systemen werden bereits in CSV-Dateien transformiert. Diese sollen eingelesen werden können.

**Welches Datenformat besitzen die zu exportierenden Daten?**

Um einen einheitlichen Datenaustausch zu gewährleisten sollen wie in unseren anderen Systemen sämtliche exportierte Datensätze im CSV-Format gespeichert werden.

**Sollen die Backups automatisch oder manuell (durch Administrator) durchgeführt werden?**

Manuelle Backups können durch den Hochzeitsmanager erstellt werden. Diese beinhalten nur Daten, die in der Hochzeitsplanung vorhanden sind.

Für unsere firmeninterne Datenbank wollen wir jedoch ein komplettes Backup.

**Wo soll das Backup gespeichert werden?**

Der Nutzer soll den Pfad angeben, auf dem das Backup hinterlegt wird. Alternativ kann auch eine URL angegeben werden, sodass das Backup ins Internet geladen werden kann.

**Welches Dateiformat soll das Backup besitzen?**

Das Backup soll ebenfalls als CSV- Datei gespeichert werden.

**Wie soll der Datenaustausch stattfinden?**

Der Datenaustausch erfolgt manuell durch den Benutzer. Dieser kann entsprechende zu importierende Daten in die firmeninterne Datenbank speichern. Weiterhin kann der Benutzer sämtliche Daten als CSV-Datei exportieren.

Der Hochzeitsmanager kann alle für seine eigene Hochzeit relevanten Daten im- und exportieren.

**Sollen die Daten nur als Backup zur Verfügung stehen oder kann man Hochzeitsdaten auch exportieren um sie später mit anderen teilen? (Wenn eine Freundin ebenfalls ihre Hochzeit ähnlich planen will)**

Es ist möglich, die exportierten CSV-Dateien mit anderen Nutzern zu teilen. Diese können einfach importiert werden.

**Können immer nur alle Daten exportiert werden oder kann man spezielle Datensätze selektieren?**

Der Nutzer soll eine Auswahlmöglichkeit besitzen, welche Datensätze er exportieren möchte.

**Wie sieht diese Auswahlmöglichkeit konkret aus?**

Er hat die Möglichkeiten folgende Daten auszuwählen und zu exportieren:

- Alles exportieren (Backup)
- Aktivitäten
- Hilfsmittel
- Caterer
- Getränke
- Essen
- Lokalitäten

**Wie soll die Auswahl dargestellt werden?**

Bei der Exportfunktion soll es eine Liste geben mit den oben genannten Kategorien. Nach einem Klick auf eine Kategorie werden in einer Selektionsliste sämtliche Datensätze angezeigt, die exportiert werden können. Einzelne Einträge können durch Anklicken ausgewählt und gesammelt exportiert werden.

Bei der Importfunktion soll eine Datei ausgewählt werden können. Diese muss dem CSV-Format entsprechen und kann danach ausgelesen werden. Aus dem ausgewählten Datensatz wird wieder eine Selektionsliste generiert, aus der die gewünschten Importdaten ausgewählt werden können.

**Wie muss das CSV-Format konkret aussehen?**

Das CSV-Format kann frei gewählt werden, da wir von keinen Exportdaten aus alten Systemen abhängig sind. Mit Ausnahme von Hilfsmitteln: Diese existieren bereits als CSV Format aus dem alten System, ein Beispiel hierfür sieht so aus:

Hilfsmittel; Lautsprecher; dient zur Beschallung

**Es werden keine ID's übergeben?**

Nein, es werden keine ID's übergeben, da diese im alten System ein anderes Format besitzen.

Die anderen Formate der anderen Datentypen würden wir dementsprechend ähnlich aufbauen, das heißt wir fangen jeden Eintrag mit der Kategorie (Datentyp) an und darauf folgen die einzelnen Daten eines Eintrages, getrennt durch Semikolon. Listeneinträge werden mit einer geschweiften Klammer geklammert?

Ja, uns ist es prinzipiell egal, welches Format die CSV-Dateien haben, solange der Im- und Export der Hilfsmittel aus anderen Systemen funktioniert.

INTERN: Beispiel einer Aktivitäts-CSV-Datei:

```
Aktivität;Tanzen;12-03-2016;18:45;12-03-2016;20:45;{Lokalität;Getreideweg;4;76313;Karlsruhe,Deutschland;Festsaal;reserviert};{};{};Tanzen zur Musik der Band;false;true;{Person;Fabian;Fellhauer;21-07-1995;{};{}},{Person;Slavisa;Markovic;02-05-1995;{};{}},{Person;Fabian;Fellhauer;21-07-1995;{};{}};geplant;{};0.00;0.00;0.00
```

Eine intuitive, leicht bedienbare Benutzeroberfläche setzen wir als selbstverständlich voraus.

**Wie stellen Sie sich die GUI vor?**

Die GUI soll übersichtlich gegliedert sein. Man soll nach dem Einloggen eine Übersicht über die Hochzeit haben. Weiterhin soll es möglich sein eine chronologische Abfolge aller Aktivitäten einzusehen.

**Soll die chronologische Abfolge der Aktivitäten als Liste dargestellt werden?**

Nein, es soll nicht einfach als Liste dargestellt werden. Das ist zu unübersichtlich. Wir hätten eher eine Art Kalender, der alle Aktivitäten übersichtlich darstellt.

**Wie soll dieser Kalender aussehen?**

Der Kalender soll die einzelnen Aktivitäten ersichtlich darstellen. Es muss das Datum erkennbar sein. Weiterhin soll dieser Kalender interaktiv gestaltet sein.

**Soll der Kalender eine Wochen oder Monatsansicht besitzen?**

Eine Monatsansicht

**Sollen dann für jeden Tag die Termine angezeigt werden?**

Es muss nicht direkt im Kalender stehen. Es reicht eine einfache Markierung, ähnlich wie bei meinem privaten Google Kalender. Bei Klick auf den Tag, sollte ich dann eine Übersicht bekommen, über sämtliche Aktivitäten an dem Tag.

**Wie soll die GUI gegliedert sein?**

Die GUI soll ebenfalls Links bieten, um jeweils wieder auf die Hochzeitsübersicht zu gelangen. Weiterhin ist es uns wichtig, dass es einfach möglich ist Hilfsmittel und Personen anzulegen, ohne diese direkt zuweisen zu müssen.

**Soll es Schnellzugriffsoptionen (Shortcuts) geben?**

Nein, da es sich bei den Benutzern um Menschen mit durchschnittlichen Computerkenntnissen handelt und diese im Normalfall keine Shortcuts verwenden. Außerdem wird die Software in vielen Fällen nur einige Male benutzt (Brautpaar) und daher bringen Shortcuts dem Benutzer keinen Vorteil.

**Soll es möglich sein mehrere Hochzeitsplanungen gleichzeitig zu öffnen und zu bearbeiten?**

Ja, jede Hochzeit soll in einem separaten Fenster geöffnet werden können. Damit ist es Hochzeitsmanagern ermöglicht, Hochzeiten miteinander zu vergleichen.

Es sollen keine besonderen Computerkenntnisse zur Bedienung der Software erforderlich sein.

**Wie sind „keine besonderen Computerkenntnisse“ definiert?**

Darunter wird ein Mensch mit durchschnittlichen Computerkenntnissen verstanden. Diese können den Computer starten, im Internet surfen und auch Suchmaschinen verwenden. Dateien erstellen und auf dem Computer abspeichern/ finden gehören ebenfalls zu den Kenntnissen.

**Welche Personengruppen nutzen die Software in der Regel?**

Der Hochzeitsmanager benutzt die Software, dieser gehört zu unserer Firma und ist geschult mit der Software umzugehen.

Des Weiteren verwenden Personen im Alter zwischen 20 und 40 Jahren die Software. Zur Zielgruppe gehören Paare, die sich Verlobt haben und Heiraten wollen.

**Sehen Sie eine Schulung der Mitarbeiter im Umgang mit der Software vor oder lediglich eine kurze Unterweisung?**

Eine kurze Unterweisung mit den verschiedenen Funktionen ist vollkommend ausreichend. Bei unseren Mitarbeitern handelt es sich um Menschen mit überdurchschnittlichen Computerkenntnissen.

**Soll es ein Benutzerhandbuch geben? Wenn ja, wie ausführlich wünschen Sie sich das Benutzerhandbuch zur Software?**

Ja, es soll ein Benutzerhandbuch geben, damit die Mitarbeiter und die Nutzer der Software jederzeit Funktionalitäten nachschlagen können.

**In welcher Form wünschen Sie sich das Benutzerhandbuch?**

Das Benutzerhandbuch soll sowohl in der Software eingebunden als auch als PDF verfügbar sein.

### 2.2.2. Anwendungsbereiche

Die Software soll ausschließlich für die Verwaltung von Hochzeiten eingesetzt werden. Sie soll bei uns in der Firma im Tagesgeschäft eingesetzt werden sowie von Privatpersonen erwerbbar sein.

**Soll es zwei unterschiedliche Varianten der Software geben?**

Nein, private Nutzer und die Mitarbeiter unserer Firma sollen dieselbe Software nutzen. Dies erleichtert den Support bei Rückfragen der privaten Nutzer.

**Wenn ja, inwiefern?**

Siehe Frage oben.

**Soll das neue Tool in eines ihrer bisherigen Werkzeuge integriert werden?**

Nein das ist nicht notwendig. Wir benötigen keine Schnittstelle zu eines unserer anderen Programmen. Es ist wichtig, dass der Hochzeitsmanager komplett eigenständig arbeiten kann.

### 2.2.3. Zielgruppen, Benutzerrollen und Verantwortlichkeiten

Als Zielgruppe kommen zwei Rollen infrage: die eigentliche planungsverantwortliche Person, welche auf sämtliche Daten lesend und schreibend Zugriff hat (Hochzeitsmanager).

**Welche administrativen Rechte hat der Hochzeitsmanager?**

Der Hochzeitsmanager kann auf sämtliche Stammdaten lesend und schreiben zugreifen. Weiterhin kann der Hochzeitsmanager auf alle Unterhaltungsaktivitäten lesend und schreiben zugreifen, es sei denn die Aktivität wurde durch einen Unterhaltungsmanager versteckt. Er hat auch Zugriff auf Projektstammdaten und kann diese bearbeiten und löschen.

**Darf der Hochzeitsmanager Daten importieren/exportieren? Oder soll es hierfür eine zusätzliche Benutzerrolle geben?**

Der Hochzeitsmanager darf Daten importieren und exportieren. Jedoch soll es noch eine weitere Benutzerrolle geben, den Administrator.

**Was kann der Administrator, was der Hochzeitsmanager nicht kann?**

Der Administrator hat projektübergreifend Zugriffsrechte, verwaltet die zentrale Datenbasis und ist für Nutzer verantwortlich. Besondere Aktionen des Administrators bilden die Datensicherung, sowie die Benutzerverwaltung.

**Kann es mehrere Administratoren geben?**

Ja, es kann mehrere Administratoren geben.

**Kann es mehrere Hochzeitsmanager geben?**

Ja es kann mehrere Hochzeitsmanager geben,

**Wenn ja, wer fügt die weiteren Hochzeitsmanager hinzu?**

Der Administrator ist für das Hinzufügen von Hochzeitsmanagern zuständig.



**Können die Hochzeitsmanager gleichzeitig arbeiten?**

Ja, das ist möglich.

**Wenn ja, gibt es eine Unterscheidung zwischen Brautpaar und externen Manager?**

Nein, es gibt hierbei keine Unterscheidung

Ausnahme: da oftmals das Brautpaar selbst planen und managen will, soll es möglich sein, die persönlichen Unterhaltsbeiträge für die Hauptplaner zu verstecken. Hierfür soll es eine zweite Rolle geben, die lesenden Zugriff auf die grundlegenden Hochzeitsdaten hat (Zeiten, Datumsangaben, Orte, ...), ansonsten aber ausschließlich die Unterhaltsbeiträge verwalten kann (Unterhaltungsmanager).

**Darf der Hochzeitsmanager doch nicht Zugriff auf alle Daten haben?**

Nein der Hochzeitsmanager darf keinen Zugriff auf versteckte Unterhaltsbeiträge haben.

**Wer fügt eine Person zur Nutzerrolle Unterhaltungsmanager hinzu?**

Der Hochzeitsmanager und der Administrator fügen Personen zur Nutzerrolle Unterhaltungsmanager hinzu.

**Kann der Hochzeitsmanager neue Hochzeitsmanager hinzufügen?**

Ja, der Hochzeitsmanager kann neben dem Administrator die Rolle Hochzeitsmanager vergeben.

**Was sind die genauen grundlegenden Hochzeitsdaten?**

Unter grundlegende Hochzeitsdaten verstehen wir sämtliche Stammdaten der Hochzeit (Braut, Bräutigam, Hochzeitsdatum, Liste der Trauzeugen, siehe Definition unter „wie ist ein Hochzeitsprojekt definiert“), sowie sämtliche damit verknüpften Aktivitäten. Editiert können durch den Unterhaltungsmanager nur Aktivitäten, die als solche gekennzeichnet sind. Das heißt, die müssen als Unterhaltsbeitrag gekennzeichnet werden.

**Was umfasst ein Unterhaltsbeitrag?**

Ein Unterhaltsbeitrag umfasst dieselben Attribute, wie eine Aktivität.

**Kann es mehrere Unterhaltungsmanager geben?**

Ja, es kann mehrere Unterhaltungsmanager geben.

**Wenn ja, dürfen andere Unterhaltungsmanager die versteckten Beiträge sehen?**

Ja, alle Unterhaltungsmanager dürfen die versteckten Beiträge sehen. Zwischen den Unterhaltungsmanagern gibt es keine Unterscheidung. Alle haben dieselben Rechte.

**Werden die versteckten Beiträge vor allen Hochzeitsmanager verborgen?**

Ja, die versteckten Beiträge werden vor allen Hochzeitsmanagern verborgen. Selbst wenn es sich bei dem Hochzeitsmanager um einen Mitarbeiter unserer Firma handelt.

**Werden die versteckten Beiträge auch vor dem Administrator versteckt?**

Nein, der Administrator hat vollen Zugriff auf alle Daten und kann die versteckten Beiträge ebenfalls sehen.



**Kann man auch normale Aktivitäten verstecken?**

Nein, es ist nur möglich Unterhaltungsbeiträge zu verstecken.

**Wer kann Unterhaltungsbeiträge verstecken?**

Nur die Unterhaltungsmanager können Unterhaltungsbeiträge verstecken. Hochzeitsmanager sind nicht in der Lage Unterhaltungsbeiträge zu verstecken oder versteckte Beiträge zu sehen.

#### **2.2.4. Zusammenspiel mit anderen Systemen**

Das zu entwickelnde Softwaresystem soll auch ohne Netzverbindung lauffähig sein. Hierzu sollen sämtliche Daten einer Hochzeit lokal gespeichert und auf Wunsch des Benutzers mit den Serverdaten synchronisiert werden können.

**Sollen sie automatisch lokal gespeichert werden?**

Nein, der Benutzer soll den Speichervorgang selbst anregen. Wurde allerdings etwas bearbeitet und der Benutzer hat dies noch nicht gespeichert, soll bei dem Verlassen der Software gefragt werden, ob die Daten gespeichert werden sollen.

**Soll der Nutzer den Synchronisationsprozess selbst starten oder sobald eine Internetverbindung verfügbar ist nach einer Synchronisation gefragt werden?**

Der Nutzer soll sobald eine Internetverbindung verfügbar ist, nach einer Synchronisation gefragt werden.

**Soll der Nutzer die Synchronisation abbrechen können?**

Nein, der Nutzer soll nach dem Start einer Synchronisation nicht mehr in der Lage sein die Synchronisation abubrechen.

**Sollen bei der Synchronisation immer die neusten Einträge in der Datenbank übernommen werden?**

Ja, es soll immer der neuste Stand übernommen werden.

**Sollen verschiedene Synchronisationsversionen gespeichert werden? Ist dabei ein Rücksetzen auf eine ältere Synchronisationsversion möglich?**

Ja, die verschiedenen Versionen, die bereits Synchronisiert wurden, sollen alle verfügbar sein. Das heißt, nur Daten die Bereits auf dem Server waren können bereitgestellt werden. Dabei kann auch eine dieser Versionen ausgewählt und wiederhergestellt werden.

**Handelt es sich bei dem Server um die eigene Datenbasis?**

Nein, bei dem Server handelt es sich um unserer Firmeneigenen Server. Nur auf diesem ist eine Versionsverwaltung gegeben. Der Benutzer kann ansonsten seine Daten nur lokal speichern. Diese werden als CSV Datei lokal dann abgespeichert.

**Können Daten auch nur lokal gespeichert werden? Gibt es die Möglichkeit eine lokale Datenbasis für private Nutzer einzurichten?**

Die Daten können auch lokal gespeichert werden. Dabei wird eine CSV Datei auf dem Rechner des Benutzers abgelegt. Wird dies gemacht, sind allerdings keine älteren Versionen mehr verfügbar. Der Benutzer ist selbst für eine Versionsverwaltung zuständig.

### 2.2.5.

#### Produktfunktionen

/LF10/ Eine Zugangsberechtigung soll mittels eines einfachen Loginvorgangs verifiziert werden.

**Welche Logindaten werden benötigt?**

Ein Benutzername und ein Passwort sind vollkommend ausreichend.

**Wer erstellt einen Benutzer?**

Unsere Firma erstellt für den Hochzeitsmanager einen Benutzer. Dieser kann weitere Hochzeitsmanager oder Unterhaltungsmanager hinzufügen.

Der Zugriff auf einzelne Daten soll je nach Berechtigung unterschiedlich erfolgen.

→ siehe Abschnitt 2.3: „Zielgruppen, Benutzerrollen und Verantwortlichkeiten“

**Soll der Datenzugriff der Benutzerrolle Unterhaltungsmanager (siehe 2.2.3) als extra Übersicht generiert werden? Z.B. als PDF-Datei mit allen wichtigen Informationen oder in der Anwendung ohne Interaktionsmöglichkeit zur Datenveränderung?**

Die Übersicht für die Benutzerrolle Unterhaltungsmanager soll die gleiche sein, wie für den Hochzeitsmanager. Allerdings hat der Unterhaltungsmanager nicht die Möglichkeit eine Datenveränderung vorzunehmen. Dabei sollen die Daten, bei denen der Unterhaltungsmanager nur Lesezugriff hat „ausgegraut“ werden. Daten, die der Unterhaltungsmanager verändern darf, sollen normal angezeigt werden.

/LF20/ Der jeweilige Benutzer muss die Möglichkeit haben, über eine grafische Benutzeroberfläche (GUI) alle für ihn relevanten Daten einfach und übersichtlich verwalten zu können.

**Was sind die relevanten Daten?**

Als relevanten Daten bezeichnen wir die Daten, auf die der Benutzer während der Systembenutzung zugreift. Das heißt, es ist von Nutzung zu Nutzung zwar unterschiedlich, aber die Software muss darauf ausgelegt sein, sämtliche Daten, die im System gespeichert sind, dem Nutzer einfach und übersichtlich wiederzugeben.

**Wie ist einfach und übersichtlich definiert?**

Grundlegende und wichtige Daten sollen hervorgehoben werden. Der Nutzer möchte nicht alle Details der gesamten Hochzeit auf einer Seite haben. Wichtig ist, dass die Stammdaten der Hochzeit als solche erkenntlich sind. Eine chronologische Übersicht über die Aktivitäten der Hochzeit muss danach dem Nutzer präsentiert werden. Hierbei sollen auch nur wichtige Daten angezeigt werden. Bei einem Klick auf die Liste, können entsprechend weitere Daten zu einzelnen Aktivitäten, Personen, Hilfsmitteln, Caterer angezeigt werden.

**Wie sind wichtige Daten der einzelnen Entitäten definiert?**

Hierbei sollen jeweils die Stammdaten der einzelnen Entitäten dargestellt werden (Titel/Name, Datum/Geburtsdatum, möglicher Status), sowie eine Navigationsmöglichkeit um weitere Details einzusehen.

**Sollen die einzelnen Datensätze als Liste angezeigt werden?**

Ja, die einzelnen Datensätze sollen als Liste angezeigt werden.

**Was verstehen Sie und „verwalten“? Nur einsehen oder auch bearbeiten?**

Unter verwalten wird einsehen als auch bearbeiten verstanden. Dabei muss auf die verschiedenen Benutzerrollen und deren Rechte geachtet werden.

**/LF30/**

Eine Hochzeitsveranstaltung fasst viele Einzelaktionen zusammen: anfangen von der standesamtlichen und kirchlichen Trauung, Buchung bzw. Reservierung von Veranstaltungsorten, Catering (Essen und Getränke getrennt verwaltbar) über Erstellung und/oder Druck von Einladungen, Tischkarten, diverse Besorgungen, Organisation und Buchung von Übernachtungen und dem Hochzeitsfahrzeug bis zur Organisation und Durchführung der Dekoration (in Standesamt, Kirche und Festsaal) uvm.

**Welche Attribute umfasst die standesamtliche Trauung?**

Die standesamtliche Trauung umfasst folgende Attribute: Trauungsdatum (Datum), Uhrzeit (Zeit), Braut (Datentyp Person), Bräutigam (Datentyp Person), Name des Standesbeamten (Datentyp Person), Trauungsort (Datentyp Lokalität), Liste der Trauzeugen (Liste mit Personen), Liste mit Hilfsmittelplanungen (Hilfsmittelplanungen), Liste mit Verpflegungen (Verpflegung), Kommentarfeld (Text), eindeutige Aktivitäts-ID (eindeutige ID), Liste mit Teilnehmer (Person), Zustand der Aktivität (Werteliste mit folgenden möglichen Werten: in Arbeit, geplant, aktiv, beendet), Liste mit Medien (Medium), bezahlte Kosten (Zahl), geplante Kosten (Zahl), gesamte Kosten (Zahl)

**Welche Attribute umfasst die kirchliche Trauung?**

Die kirchliche Trauung umfasst folgende Attribute: Trauungsdatum (Datum), Uhrzeit (Zeit), Braut (Datentyp Person), Bräutigam (Datentyp Person), Name des Pastors/der trauenden Person (Datentyp Person), Trauungsort (Datentyp Lokalität), Liste mit Hilfsmittelplanungen (Hilfsmittelplanungen), Liste mit Verpflegungen (Verpflegung), Kommentarfeld (Text), eindeutige Aktivitäts-ID (eindeutige ID), Liste mit Teilnehmer (Person), Zustand der Aktivität (Werteliste mit folgenden möglichen Werten: in Arbeit, geplant, aktiv, beendet), Liste mit Medien (Medium), bezahlte Kosten (Zahl), geplante Kosten (Zahl), gesamte Kosten (Zahl)

**Was ist der Unterschied zwischen einer Buchung und einer Reservierung?**

Bei einer Buchung wurden die Kosten bereits von dem Konto abgebucht bzw. werden auf jeden Fall abgebucht.

Bei einer Reservierung handelt es sich ausschließlich nur um eine Reservierung, aber es sind noch keine Kosten entstanden und die Reservierung kann aufgelöst werden.

**Welche Attribute umfassen Buchung/Reservierung?**

Eine Buchung oder Reservierung umfasst keine Attribute. Diese bilden lediglich einen Status ab, der im entsprechenden Veranstaltungsort hinterlegt ist.

**Wie ist ein Veranstaltungsort definiert?**

Ein Veranstaltungsort ist mit einer Lokalität gleichzusetzen. Die Attribute einer Lokalität wurden oben bereits definiert.

**Wie ist das Catering definiert?**

Das Catering ist mit der Verpflegung gleichzusetzen, und diese wurde oben bereits definiert.

**Soll es möglich sein für Essen und Getränke jeweils einen anderen Caterer zu buchen?**

Zu jeder Verpflegung kann eine eigene Liste mit Essen und Getränken hinzugefügt werden. Weiterhin kann für eine Aktion mehrere Verpflegungen eingetragen werden. Daraus resultiert, dass ebenfalls Essen und Getränke von einem anderen Caterer geliefert werden können.

**Wer erstellt Einladungen und wer druckt diese?**

Der Hochzeitsmanager erstellt die Einladungen und gibt diese in den Druck. Das Ganze soll über eine Aktivität protokolliert werden. Eine Auftragssendung zu einem Einladungsdruckunternehmen wird in der Anwendung nicht gefordert.

**Wie ist dieser Prozess definiert (Verwaltung der Einladungen)?**

Einladungen werden im System als Hilfsmittel abgebildet. Diese können einer Aktivität (z.B. „Einladungen versenden“) hinzugefügt werden. Weiterhin werden sämtliche eingeladen Personen in dieser Aktivität als Teilnehmer hinterlegt. Personen die zugesagt haben können anschließend in einer anderen Aktivität (z.B. „Hochzeitsfeier“) als Teilnehmer hinzugefügt werden.

**Wie sind die Tischkarten definiert?**

Tischkarten werden als Hilfsmittel definiert. Dies wurde oben bereits schon erläutert.

**Was versteht man unter diverse Besorgungen?**

Weitere Besorgungen, die für eine Hochzeit anfallen. Diese können sehr unterschiedlich ausfallen. Das heißt, es ist darauf zu achten, dass man die Besorgungen in einer allg. Darstellungsform anzeigen lassen kann.

**Wie ist die Organisation von Übernachtungen definiert?**

Die Organisation wird ebenfalls als Aktivität verwaltet. Hierbei wird ein Übernachtungsort (Lokalität), sowie die Teilnehmer und die Kosten hinterlegt.

**Gibt es einen Unterschied zu einer konkreten Buchung?**

Eine konkrete Buchung wird als Statuswechsel festgehalten. Das heißt, aus einer organisatorischen Erstplanung wird zunächst eine Reservierung, bzw. anschließend eine feste Buchung, bei der in der Regel Belege bereits hinterlegt werden können.

**Gibt es bei Übernachtungen keine Reservierung?**

Doch eventuelle Reservierungen müssen im System festgehalten werden können, da es Hotels gibt, die eine Reservierung zulassen.

**Wie ist das Hochzeitsfahrzeug definiert?**

**Das Hochzeitsfahrzeug wird ebenfalls als Hilfsmittel definiert.**

**Soll es möglich sein, Wünsche zum Hochzeitsauto abgeben zu können?**

Nein, Wünsche müssen im System nicht hinterlegt werden. Es genügt die Möglichkeit ein Auto oder mehrere Autos als Hilfsmittel hinzuzufügen und anschließend, die nicht gewollten Autos zu entfernen.

**Wie ist die Dekoration definiert?**

Dekoration wird ebenfalls als Hilfsmittel definiert.

**Muss Dekoration vorhanden sein?**

Nein, sämtliche Hilfsmittel sind optional.

**Soll die Dekoration in den genannten Räumen dieselbe sein?**

Nein, auch hierbei wird großen Wert auf Benutzerfreiheit gelegt, sodass es möglich sein soll, in unterschiedlichen Lokalisationen unterschiedliche Dekorationen zu verwenden.

**Gibt es noch andere Lokalisationen außer Standesamt, Kirche und Festsaal?**

Ja, Lokalisationen können vom Hochzeitsmanager angelegt und verwaltet werden. Hierbei kann es die unterschiedlichsten Orte geben.

**Was verbirgt sich hinter „uvm“?**

Hierunter fallen weitere Aktivitäten, die während einer Hochzeit anfallen können. Da diese auch sehr unterschiedlich ausfallen können, ist darauf zu achten, dass Aktivitäten ebenfalls eher allgemein formuliert werden können. Der Benutzer soll die Möglichkeit haben über ein Kommentarfeld Freitext zu jeder Aktivität zu hinterlegen, sodass alles abgedeckt werden kann.

**/LF40/**

Jede Aktion beginnt und endet zu einem bestimmten Zeitpunkt, es müssen verantwortliche Personen und Teilnehmer der Aktion benannt werden können. Die verantwortlichen Personen und Teilnehmer sollen automatisch per E-Mail benachrichtigt werden können (z.B. durch starten eines vorhandenen Mail-Tools über die grafische Benutzeroberfläche).

**Wie ist eine Aktion genau definiert?**

Eine Aktion ist mit einer Aktivität gleichzusetzen. Diese wurde oben bereits definiert.

**Wie ist ein Zeitpunkt definiert?**

Ein Zeitpunkt setzt sich zusammen aus einem Datum (Datumsformat: DD-MM-JJJJ) und einer Uhrzeit (Uhrzeitsformat: HH-MM, 24h Format).

**Was sind die Unterschiede zwischen einer verantwortlichen Person und einem Teilnehmer?**

Eine verantwortliche Person verwaltet die Aktivität, sowie die dazugehörigen Elemente (Verpflegung, Hilfsmittel, Teilnehmer). Ein Teilnehmer ist ebenfalls eine Person, wirkt jedoch lediglich in dieser Aktivität mit. Die Organisation obliegt bei der verantwortlichen Person.

**Welche Eigenschaften müssen die einzelnen Rollen besitzen?**

Die Rollen benötigen keine zusätzlichen Eigenschaften, da diese in einer Aktivität hinterlegt werden. Jedoch soll es möglich sein, zu einer Person herauszufinden, in welchen Aktivitäten diese teilnimmt oder sie sogar organisatorische Aufgaben besitzt.

**Wer benennt diese Personen?**

Durch das Anlegen einer Aktivität wird der aktuelle Nutzer als verantwortliche Person automatisch hinzugefügt. Diese Person hat dann zusätzlich die Möglichkeit weitere Personen einzutragen.

**Wann soll eine E-Mail automatisch versendet werden?**

Eine E-Mail wird versendet, sobald eine Person als verantwortliche Person oder als Teilnehmer hinzugefügt oder entfernt wurde und eine E-Mail Adresse der entsprechenden Person im System hinterlegt ist.

**Wie ist automatisch definiert? (Soll der Benutzer immer noch das Senden der Mail bestätigen?)**

Ja, der Benutzer soll eine Auswahl erhalten (Selektionsliste) an welche Personen die E-Mail versendet werden soll. Hierbei hat er ebenfalls die Möglichkeit alle zu deselektieren/selektieren, sodass er auch keine Mail versenden kann.

**Muss hierzu ein Mail-Tool auf demselben Rechner installiert sein?**

Ja hierzu muss ein Mailing-Tool auf dem Rechner installiert sein und dieses muss als Default-Mail-Tool im Betriebssystem hinterlegt sein.

**Sollen die E-Mails personalisiert sein?**

Ja, beim Versenden der Mails soll es möglich sein, den Namen des Empfängers an einer beliebigen Stelle zu setzen. Dann soll die E-Mail für jeden Empfänger mit seinem Namen an der gesetzten Stelle versendet werden. Es soll der hinterlegte Name der Person im System verwendet werden.

**Wenn vom Namen die Rede ist, Vor- und Nachname oder nur eins von beiden?**

Der Benutzer soll beim Senden der E-Mails entscheiden können, welcher Teil des Namens verwendet werden soll. Es soll auch möglich sein beide zu schreiben.

Jede Aktion kann an mehreren Orten stattfinden, mit Hilfsmitteln aus einer Liste durchgeführt werden. Anfallende Kosten (Rechnungen, Belege) sollen jeweils mit angegeben werden können.

**Wie ist ein Ort definiert?**

Ein Ort ist mit einer Lokalität gleichzusetzen. Diese wurde oben bereits definiert.

**Wie sind Hilfsmittel definiert?**

Diese wurden oben bereits definiert.

**Wie sieht die Liste aus? Soll diese Liste auch so auf der GUI erscheinen?**

Die Liste der Hilfsmittel wird als einfache Liste angezeigt. Hierbei steht der Titel des Hilfsmittels im Vordergrund. Die Liste soll nach dem Alphabet sortiert sein.

**Welche Eigenschaften umfassen Kosten?**

Die Kosten wurden oben bereits definiert. Diese besitzen einen Kostenposten (Text), Betrag (Zahl), welcher in der endgültigen Darstellung ein €-Zeichen angehängt bekommen, sowie eine Klassifizierung, welche Kostenart es ist (Auswahl aus Werteliste: geplant, bezahlt).

**Wie sind Rechnungen definiert?**

Rechnungen werden durch Dritte ausgestellt. Diese können im System bei entsprechenden Kosten hinterlegt werden. Rechnungen können weiterhin unterschiedliche Dateiformate besitzen wie PDF, JPG, PNG. Jedoch spielt das Dateiformat für das System keine entscheidende Rolle, da alle möglichen Formate hochgeladen werden können. Der Benutzer braucht entsprechende Programme um die Dateien auf dem Rechner zu öffnen. Deswegen empfehlen wir PDF, JPG oder PNG zu verwenden.

**Wie sind Belege definiert?**

Belege besitzen die gleiche Definition wie Rechnungen (siehe oben).

**Wie sollen die Rechnungen und Belege in System geladen werden?**

Rechnungen und Belege werden von Dritten geliefert. Z.B. als PDF in einer E-Mail oder als Brief per Post (Dieser muss eingescannt oder abfotografiert werden). Sobald die Rechnung/Beleg als virtuelle Datei vorliegt, kann diese im System hochgeladen werden. Diese werden dann als Bild in einer Liste dargestellt.

**Wie werden die Rechnungen und Belege mit den Kosten verknüpft?**

Rechnungen/Belege werden direkt einem Kostenposten zugewiesen. Dieser muss hierzu angelegt werden. Da Rechnungen/Belege jedoch nur im Bild-Format dargestellt werden, findet keine Überprüfung statt, die die Kosten mit den tatsächlichen Belegen vergleicht.

**Sollen auch Kosten eingetragen werden können, die noch keine Rechnung/Beleg besitzen?**

Ja, das soll möglich sein.

**Sollen auch geplante Kosten eingetragen werden können?**

Ja geplante Kosten werden über die Klassifizierung der Kosten eingetragen. Hierbei wird zwischen bereits bezahlten und geplanten Kosten unterschieden.

**Sollen Rechnungen und Belege auch nachträglich hinzugefügt werden können?**

Ja, Rechnungen und Belege sollen auch im Nachhinein hinzugefügt werden können.

Für den Benutzer soll es leicht möglich sein, die aktuellen Zustände aller Aktionen zu erkennen (geplant, in Arbeit, abgeschlossen, usw.) und zu ändern. Der Benutzer soll



geeignet dabei unterstützt werden, bestimmte vorgegebene Aktionsarten anlegen und durchführen zu können. Dabei soll der Benutzer einfach erkennen können, welche Aktionen bereits angelegt sind und welche noch nicht.

**Besitzt jede Aktion einen Zustand?**

Ja, jede Aktion besitzt die oben bereits genannten Zustände.

**Welche Zustände soll es genau geben?**

Eine Aktivität soll folgende Zustände besitzen: in Arbeit, geplant, aktiv, beendet, abgebrochen.

Eine Lokalität soll folgende Zustände besitzen: geplant, reserviert, gebucht, storniert.

**Sollen die Zustände farblich markiert werden? Wenn ja, wie?**

Ja, zur besseren Übersicht sollen die Zustände ähnlich einer Ampel farbcodiert werden: geplant (rot), in Arbeit/reserviert (gelb), abgeschlossen/gebucht (grün), abgebrochen/storniert (dunkelrot).

**Wie sind die unterschiedlichen Aktionsarten definiert?**

Es gibt unterschiedliche Aktionsarten: Zu einem gibt es als Aktion die Standesamtliche, sowie kirchliche Trauung, die den Mittelpunkt der Hochzeit bilden. Zusätzlich gibt es noch weitere Aktivitäten, die erledigt werden müssen, diese können zusätzlich als Unterhaltungsbeitrag und/oder als geheimer Beitrag gekennzeichnet werden.

**Wie soll der Benutzer hierbei unterstützt werden?**

In der Datenbasis soll es eine Sammlung von wichtigsten Aktivitäten geben, die dem Hochzeitsmanager vorgeschlagen werden.

Weiterhin soll ein Unterhaltungsmanager bei dem Anlegen einer Aktivität daran erinnert werden, dass die Aktivität als geheim eingestuft werden kann, sodass das Brautpaar (Hochzeitsmanager) davon nichts mitbekommt.

**Was versteht man unter der Durchführung einer Aktionsart?**

Das Anlegen, sowie Verwalten einer Aktivität.

**Wie soll erkenntlich gemacht werden welche Aktionen bereits angelegt wurden?**

Sobald man eine Aktivität anlegt und den Titel eingibt, soll eine Vorschlagsliste aufgebaut werden, die ähnliche Titel anzeigt, welche bereits in dieser Hochzeit angelegt wurden. Daraus kann der Nutzer schnell sehen, ob eine Aktivität doppelt angelegt wird.

Es soll darüber hinaus möglich sein, einer Aktion verschiedene Medien (Dokumente, Bilder, Videos, usw.) zuzuordnen.

**Wie sollen die einzelnen Medien ins System geladen werden?**

Die Medien müssen auf dem Rechner des Nutzers vorhanden sein. Auf der Detail-Seite einer Aktivität gibt es die Möglichkeit Medien der Aktivität hinzuzufügen. Hierbei öffnet sich ein Dialog, welcher den Nutzer dazu auffordert, das



entsprechende Medium auszuwählen. Dieses wird dann anschließend in einer Ordnerstruktur geladen (kopiert).

**Welche konkreten Medien werden unterstützt (Dateiformate?)**

Es werden alle Dateiformate unterstützt, da die Dateien nur in das System geladen werden und von da aus wieder auf den Computer des Nutzers gespeichert werden. Das System ist nicht für das Abspielen von Videos/ Musik oder zum Öffnen von Bildern/ Textdateien gedacht. Hierfür muss der Benutzer ein passendes Programm auf dem Computer haben, mit dem er das gewünschte Dateiformat öffnen kann.

**Welche weiteren Medien gibt es noch?**

Tonspuren sollen ebenfalls gespeichert werden können.

**Kann einer Aktion mehrere Medien des gleichen Typs hinzugefügt werden?**

Ja, das soll möglich sein.

**/LF50/**

Jede verantwortliche Person und jeder Teilnehmer kann bei den üblichen Kontaktdaten mehrere E-Mail-Adressen und mehrere Telefonnummern besitzen.

**Was sind übliche Kontaktdaten?**

Übliche Kontaktdaten sind: Vorname, Nachname, Geburtsdatum, E-Mail Adresse und Telefonnummer.

**Was sind mehrere? Unbegrenzt viele?**

Jede Person kann bis zu 5 E-Mail Adressen und 5 Telefonnummern haben.

**Müssen die Telefonnummern klassifiziert werden? (Festnetz, Handy, Arbeit)**

Die Telefonnummern müssen nicht, aber können klassifiziert werden. Werden sie nicht explizit klassifiziert, wird „Privat“ als Klassifizierung angegeben.

**Gilt dasselbe für E-Mails?**

Ja, dasselbe Schema gilt auch bei den E-Mail Adressen.

**Bei E-Mail Versand: Soll jeweils an jede eingetragene E-Mail Adresse eine Mail versendet werden?**

Nein, jede Person hat eine primäre E-Mail Adresse. Ist nur eine Adresse angelegt, so ist diese das. Sind mehrere E-Mail Adressen angegeben, so kann der Anleger die primäre Adresse angeben. Wird keine explizit als primäre Adresse angegeben, ist es die E-Mail Adresse, die zuerst ins System eingetragen wurde.

Die E-Mails werden nur an die primäre E-Mail Adresse verschickt. Der Hochzeitsmanager hat immer noch die Möglichkeit bei dem Versenden von E-Mails die anderen E-Mail Adressen auszuwählen.

**Soll die primäre E-Mail Adresse hervorgehoben werden?**

Ja, die primäre E-Mail Adresse soll in der Liste der E-Mail Adressen einer Person ganz oben stehen, mit einem entsprechenden Vermerk.

/LF60/ Das Catering kann entweder von einem kommerziellen Catering-Service als auch von ausgewählten Personen durchgeführt werden.

**Wie ist ein kommerzieller Catering –Service definiert?**

Bei einem kommerziellen Caterer handelt es sich um einen Caterer, der dies geschäftlich tut. Dies soll ebenfalls über die Software ersichtlich sein.

**Wie ist eine ausgewählte Person definiert?**

Bei einer ausgewählten Person handelt es sich um eine natürliche Person, die das Catering nicht geschäftlich unternimmt. Diese Person wird ebenfalls als Caterer eingetragen, jedoch soll ersichtlich sein, dass die Person das nicht geschäftlich betreibt.

**Inwiefern soll dies ersichtlich sein?**

Es soll entweder über einen Eintrag auf einer Detailseite des Caterers ersichtlich sein, aber auch auf der Übersichtsliste aller Caterer farblich markiert sein.

**Welche Eigenschaften umfasst der Cateringservice?**

Der Cateringservice ist mit dem Caterer gleichzusetzen und wurde oben bereits definiert.

**Welche Eigenschaften umfasst das Catering?**

Das Catering ist mit der Verpflegung gleichzusetzen und wurde oben bereits definiert.

/LF70/ Die oben erwähnte Liste der Hilfsmittel soll auf einfache Weise erweiterbar und zuweisbar sein. Sie sollen für sämtliche Hochzeitsveranstaltungen im System verfügbar sein.

**Wie sollen die Hilfsmittel zugewiesen werden?**

Hilfsmittel sollen über eine Liste ausgewählt und den Aktivitäten zugewiesen werden können. Bei der Auswahl der Hilfsmittel öffnet sich ein Dialog, in dem alle Hilfsmittel alphabetisch nach dem Titel sortiert aufgelistet sind.

**Wie soll die Liste erweitert werden?**

In dem Dialog, welches sich öffnet, soll es einen Button geben, mit dessen Hilfe man ein neues Hilfsmittel anlegen kann.

**Wer darf Hilfsmittel erstellen?**

Alle Benutzerrollen dürfen Hilfsmittel erstellen.

**Sollen Hilfsmittel auch nur für sich selbst erstellt werden können?**

Nein, erstellte Hilfsmittel sollen für alle anderen Nutzer auf der Systemlandschaft zugänglich sein.

**Dürfen private Nutzer auf das globale Hilfsmittel System zugreifen?**

Ja, private Nutzer dürfen auch auf globale Hilfsmittel zugreifen. Dabei muss allerdings eine aktive Internetverbindung verfügbar sein, da die Daten nicht vorher heruntergeladen werden.

**Gibt es noch andere Systeme aus denen Hilfsmittel geladen werden können?**

Es gibt keine anderen Systeme, aus denen Hilfsmittel direkt geladen werden können. Es können nur Hilfsmittel zusätzlich importiert werden, wenn die Import Datei dem Benutzer vorliegt.

/LF80/ Zur Kostenkontrolle soll es möglich sein, sämtliche bisher angefallenen Kosten auf einfache Weise addieren zu können. Zur Kostenabschätzung sollen auch geschätzte Kosten angegeben und addiert werden können.

**Soll die Addition vom System automatisch durchgeführt werden?**

Ja, sobald Kosten eingetragen sind, sollen diese automatisch von dem System berechnet werden. Hierbei werden die Kosten innerhalb einer Aktivität addiert. Weiterhin werden alle Kosten aller Aktivitäten addiert, welche die Gesamtkosten der Hochzeit widerspiegeln.

**Wenn ja, wann sollen die Kosten berechnet werden?**

Sobald an einer Stelle im Programm die Kosten geändert bzw. neu eingetragen werden, sollen die Kosten neu berechnet werden.

**Sollen auch Kosten ohne Belege/Rechnungen beachtet werden?**

Ja, sobald Kosten eingetragen werden, sollen diese beachtet werden. Dabei spielt es keine Rolle, ob ein Beleg verfügbar ist oder nicht.

**Soll eine Kostenübersicht am Ende der Addition erstellt werden?**

Ja, im Programm soll es eine Kostenübersicht geben. In dem Projekt der Hochzeitsplanung soll ganz am Ende diese Übersicht zu sehen sein.

**Soll es eine dauerhafte Kostenübersicht geben?**

Ja, am Ende des Projektes soll es eine Übersicht geben.

**Sollen die geplanten/geschätzten Kosten getrennt berechnet werden?**

Zunächst sollen alle bezahlten Kosten berechnet werden und der Gesamtbetrag festgehalten werden. Danach sollen die geplanten/ geschätzten Kosten addiert werden und der Gesamtbetrag notiert. Und am Ende sollen die beiden Gesamtbeträge zusammen addiert werden und ergeben damit die Kosten der Hochzeit. Hierbei wird für jede Aktivität alle drei Kennzahlen festgehalten und zusätzlich für die Hochzeit komplett.

/LF90/ Die Auswahl der Daten soll möglichst über (eventuell durchsuchbare) Auswahllisten erfolgen. Dies gilt vor allem für Zuordnungen von ausgewählten Personen zu den Aktionen usw.

**Welche Daten sollen in Listen angezeigt werden?**

Sämtliche Daten, die als Übersicht dargestellt werden. In dem Hochzeitsprojekt soll somit eine Übersichtsliste aller Aktivitäten verfügbar sein. Weiterhin soll es in

der Aktivität jeweils für Teilnehmer, verantwortliche Personen, Verpflegung, Hilfsmittel, sowie für Kosten Übersichtslisten generiert werden.

**Welche Daten sollen zuweisbar sein?**

Zuweisbare Daten sind sämtliche Datensätze die im System als unabhängiger Datensatz gespeichert werden und frei mit anderen Datensätzen verknüpft werden können. Das sind zu einem Personen (welche nur im Hochzeitsprojekt selbst gespeichert werden), sowie Hilfsmittel, Essen, Getränke, sowie kommerzielle Caterer, welche für alle Hochzeitsprojekte zugänglich sind.

**Sollen die Auswahllisten als Listen angezeigt werden?**

Ja die Auswahlliste sollen als Liste mit den wichtigsten Daten angezeigt werden.

**Wie sind wichtige Daten definiert?**

Wichtige Daten, sind jene Daten, die einen kurzen Einblick in den jeweiligen Listeneintrag ermöglichen. Darunter fallen Titel, mögliche Datumsangaben, eine kurze Beschreibung (Eintrag des Kommentarfeldes), sowie eine Navigationsmöglichkeit um sich mehr Informationen anzeigen zu lassen.

**Wie soll die Suche definiert werden?**

Die Suche soll über ein Eingabefeld am Anfang erreichbar sein. Weiterhin sollen die einzelnen Daten in der Listenübersicht nach einer Textsuche des Suchfeldes gefiltert werden. Dies soll als Live-Suche verfügbar sein.

**Nach welchen Eigenschaften soll gesucht werden?**

Sämtliche Daten, die in der Übersicht angezeigt werden sollen auf die Eingabe des Suchfeldes geprüft werden.

**Sollen die Listen sortiert angezeigt werden?**

Die Liste soll alphabetisch nach dem Titel des entsprechenden Datensatzes sortiert werden (Beispiel: Nachname Vorname bei einer Person, Titel einer Aktivität, Titel eines Hilfsmittel, Kostenposten bei einer Kostenliste)

/LF100/ Vor dem Hinzufügen von neuen Daten soll eine Überprüfung stattfinden, ob diese eventuell schon vorhanden sind. Das gilt in besonderem Maße für Personen und Aktionen.

**Wie soll die Überprüfung aussehen?**

Die Überprüfung prüft die eingegebenen Werte bei dem Anlegen einer Aktivität/Person/Hilfsmittel und prüft ob es im System bereits ähnlich angelegte Datensätze gibt. Falls dies der Fall ist, wird dem Benutzer einen Dialog angezeigt, welcher darauf hinweist, dass es diesen ähnlichen oder sogar identischen Datensatz bereits gibt und ob man nicht diesen auswählen wolle, anstatt einen neuen anzulegen. Die endgültige Entscheidung obliegt jedoch weiterhin dem Nutzer, ob ein Datensatz angelegt werden soll.

Geprüft werden alle Felder eines Datenobjektes, welche keine Listen sind.

**Welche Datentypen sollen konkret geprüft werden?**

Es sollen folgende Datentypen geprüft werden: Aktivitäten, Personen, Hilfsmittel, Caterer, Essen, Getränke

**Was soll passieren wenn ein Datensatz bereits existiert?**

Es öffnet sich der oben bereits erläuterte Dialog.

**Gibt es Aktionen die öfters angelegt werden können (Beispiel: Tanzen)?**

Ja, es soll möglich sein Aktivitäten öfters anlegen zu können. So soll es beispielsweise möglich sein, dass man während der Hochzeit mehrmals eine Aktivität Tanzen hat, oder dass es zwei unterschiedliche Personen mit demselben Namen existieren. Die endgültige Entscheidung obliegt immer beim Nutzer!

### 2.2.6. Produktdaten

**/LD10/** Die Daten sollen zentral verwaltet und in einer Datenbank abgespeichert werden.

**Sollen alle Daten zentral verwaltet werden? Oder kann ein privater Nutzer die Daten auch lokal verwalten?**

Daten werden in unserer Firma zentral in einer Datenbank verwaltet. Private Nutzer sollen die Möglichkeit haben Ihre Software mit unserer Datenbank zu verbinden, um auf die globale Sammlung der Hilfsmittel/kommerziellen Caterer/essen/Getränke und vorgeschlagene Aktivitäten zugreifen zu können. Dies ist jedoch optional. Der Nutzer soll auch die Möglichkeit sein eigenes Projekt gekapselt von unserer Datenbank zu erstellen. Dies ist oben bereits genauer erläutert.

**Besteht die Möglichkeit die Daten der privaten Nutzer ebenfalls in der Datenbank zu speichern?**

Ja, der private Nutzer hat die Möglichkeit, er muss jedoch seine Daten nicht teilen.

**Wie soll der private Nutzer sich mit der firmeninterne Datenbank verbinden?**

Bei der Softwareauslieferung werden administrative Zugangsdaten mitgeliefert. Beim Aufsetzen/Installieren der Software wird abgefragt ob eine Verbindung mit der Datenbank erstellt werden soll. Diese Verbindung wird über das Internet realisiert.

**Was passiert wenn er sich nicht mit der firmeninterne Datenbank verbindet?**

Hierbei muss der private Nutzer sein Projekt immer lokal auf seinem Rechner speichern. Er hat hierbei keinen Zugriff auf globale Hilfsmittel und kann diese auch nicht über unsere Datenbank teilen.

**Existiert bereits eine Datenbank?**

Für das aktuelle System eines Hochzeitsplaners existiert noch keine Datenbank.

**Existiert eine andere Datenbank?**

Wir besitzen Datenbanken für andere Eventplanungssysteme.

**Was für eine Datenbank nutzen Sie hierbei?**

Wir nutzen eine MySQL Datenbank.

**Werden die Daten der privaten Nutzer auch anderen Nutzer zugänglich gemacht?**

Daten von privaten Nutzer werden nur anderen Nutzer zugänglich gemacht, wenn diese sich zuvor mit unserer Datenbank verbunden haben.

**Wenn ja, welche Daten werden veröffentlicht?**

Veröffentlicht werden nur: Hilfsmittel, kommerzielle Caterer, Essen, Getränke.

**Muss Mehrbenutzerbetrieb möglich sein?**

Ja, Mehrbenutzerbetrieb soll ermöglicht werden.

### 2.2.7. Produktleistungen

/LL10/ Das Laden gewünschter Daten soll für eine sinnvolle Benutzung im Sekundenbereich erfolgen.

**Wie ist der Sekundenbereich definiert? Wie hoch darf die maximale Ladezeit sein?**

Die maximale Ladezeit soll bei 5 Sekunden liegen.

/LL20/ Die Anzahl der zu verwaltenden Elemente wird auf ca. 100.000 geschätzt.

**Welche Elemente sollen verwaltet werden?**

Sämtliche oben definierten Datentypen müssen verwaltet werden.

**Sind die 100.000 Elemente die Obergrenze oder soll die Zahl erweiterbar sein?**

Die 100.000 Elemente bilden nicht die Obergrenze. Diese Zahl wird je nach Erfolg der Software sich erhöhen. Jedoch rechnen wir in den ersten zwei Jahren mit einer konstanten Nutzerzahl, sodass 100.000 Elemente verwaltet werden müssen. In ferner Zukunft wird die Zahl steigen, somit muss die Software in dieser Hinsicht erweiterbar sein.

/LL30/ Die Daten müssen bei unserer eigenen Verwendung aus rechtlichen Gründen 10 Jahre online verfügbar sein.

**Sollen Daten komprimiert archiviert werden?**

Ja, die Daten sollen komprimiert archiviert werden. Wie sie komprimiert werden, spielt keinerlei Rolle.

**Wann sollen Daten archiviert werden?**

Daten werden automatisch 1 halbes Jahr nach Abschluss der letzten Aktivität archiviert. Sobald sich ein Nutzer in das System einloggt und archivierte Daten fordert, müssen diese erst dearchiviert werden.

**Muss man auf archivierte Daten immer noch schnell zugreifen können?**

Nein, auf archivierte Daten muss man nicht schnell zugreifen können. Das Laden der Daten beschränkt keiner zeitlichen Länge, da auf diese im Regelfall nicht zugegriffen wird und falls doch, spielt die zeitliche Dauer keine Rolle.

/LL40/ Um bei Anschaffungen und Neuerungen flexibel zu bleiben, ist auf Plattformunabhängigkeit besonders zu achten.

**Welche Plattformen nutzen Sie aktuell?**

Nur Windows 7

**Welche Betriebssystem-Versionen sollen unterstützt werden?**

Es sollen Windows ab Windows 7 und Ubuntu ab Ubuntu 14 unterstützt werden.

### 2.2.8. Qualitätsanforderung

Produktqualität	sehr gut	gut	normal	nicht relevant
Funktionalität	X			
Zuverlässigkeit		X		
Effizienz			X	
Benutzbarkeit	X			
Änderbarkeit			X	
Übertragbarkeit			X	
Gestaltung	X			

## 2.3. Vereinfachungen für den Programmentwurf

- Es muss nicht dafür gesorgt werden, dass auf dieselben Daten der Datenbank nicht gleichzeitig zugegriffen werden kann, d.h. es ist kein Locking-Mechanismus erforderlich.
- Eine Protokollierfunktion ist für die Anwendung nicht erforderlich (in der Realität natürlich schon!).
- Ein Loginvorgang und eine Benutzerverwaltung müssen in den Klassendiagrammen nicht modelliert und später auch nicht implementiert werden
- Die Synchronisation der Daten muss bei der Implementierung nicht berücksichtigt werden.

P.S.: Kopieren Sie den Aufgabentext (d.h. ohne Frontseite) vollständig als erstes Kapitel „Aufgabenstellung“ an den Anfang Ihrer PE-Dokumentation und verwenden Sie den Aufgabentext zusätzlich als Rahmen für Ihre Lastenheftanalyse („ausfüllen“ mit Fragen und Antworten)!

### 3. Analyseklassendiagramm

Das Analyseklassendiagramm umfasst sämtliche im Analyseheft festgehaltenen Entitäten. Es stellt die Beziehungen der verschiedenen Objekte in dem Hochzeitsprojekt dar. Die Objekte ergeben sich aus der vorherigen Analyse des Lastenheftes und der Betrachtung der verschiedenen Anwendungsfälle.

#### 3.1. Analyse der verschiedenen Objekte

Die aus der Analyse ergebenden Objekte werden im Folgenden betrachtet und in einen Gesamtzusammenhang gestellt. Dabei wird nicht auf jedes Attribut der Objekte eingegangen, sondern nur auf die wesentlichen und nicht eindeutigen Attribute. Jede Klasse im Analyseklassendiagramm besitzt eine ID zur eindeutigen Identifikation.

Der Hochzeitsplaner besitzt einige Objekte, die im Vorhinein schon festgelegt sind. Dazu gehören das Hochzeitsprojekt mit Lokalität, die standesamtliche Hochzeit mit Lokalität, einem Nutzer (dem Hochzeitsmanager) und drei Personen. Dabei handelt es sich um folgende Personen: der Braut, dem Bräutigam und dem Standesbeamten. Die drei Personen besitzen jeweils mindestens eine Telefonnummer. Diese Objekte werden angelegt, sobald ein neues Hochzeitsprojekt angelegt wird. Der Hochzeitsmanager muss im Laufe der Hochzeitsplanung die Daten für diese Objekte bereitstellen, da es sich ansonsten um keine gültige Hochzeit handelt.

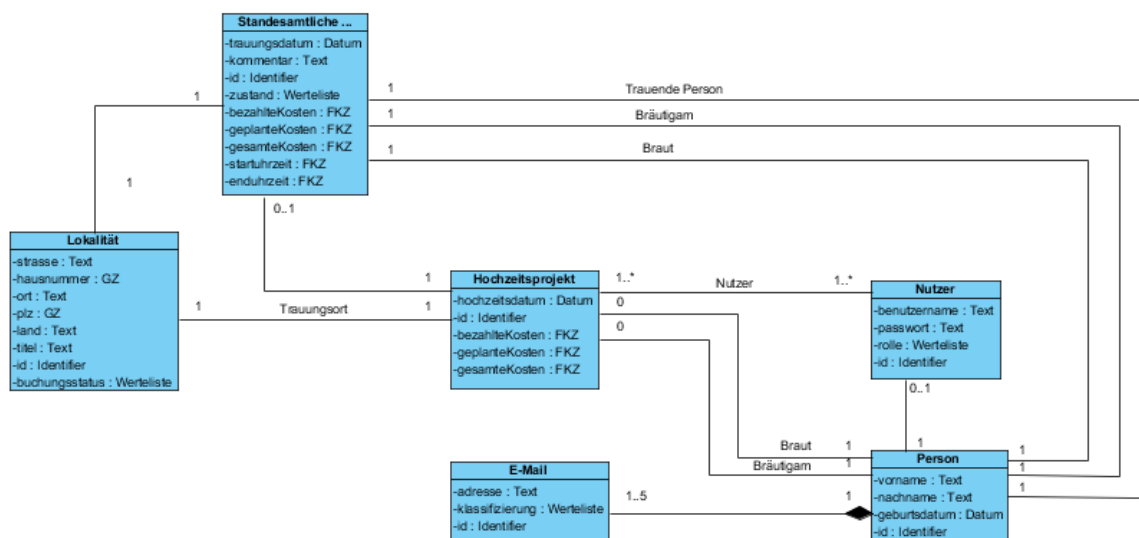


Figure 1: Hochzeitsprojekt benötigte Klassen



## Hochzeitsprojekt

Das Hochzeitsprojekt ist das Hauptobjekt der Hochzeitsplanung. Anhand dieser Klasse können alle Daten sichergestellt werden. Die benötigten Verbindungen zu anderen Klassen wurden oben bereits erwähnt und werden deshalb nicht noch einmal erläutert. Das Hochzeitsprojekt beinhaltet zwei Kostenkategorien. Bei bezahlten Kosten handelt es sich um Kosten, die bereits verbucht sind. Die geplanten Kosten sind Kosten, die voraussichtlich anfallen werden, sich jedoch jederzeit ändern beziehungsweise wegfallen können. Um die Kosten zu ermitteln, werden die einzelnen Aktivitäten durchgeschaut und die bezahlten bzw. geplanten Kosten aufaddiert. Die Gesamtkosten werden aus den zwei vorherigen Kostentypen berechnet. Ein Hochzeitsprojekt benötigt mindestens einen Benutzer (den Hochzeitsmanager) und kann beliebig viele weitere Benutzer haben.

## Nutzer

Bei einem Nutzer handelt es sich um eine Person, die Zugriff auf das System hat. Dieser Nutzer ist einer Person eindeutig zugewiesen und wird mit Benutzername und Passwort gesichert. Ein System kann beliebig viele Nutzer haben, jedoch ist mindestens einer vorausgesetzt. Jeder Nutzer besitzt eine Rolle, die vom Hochzeitsmanager bestimmt wird. Dabei sind folgende Rollen möglich:

- Hochzeitsmanager: Ist für die Hochzeit zuständig und kann sämtliche Daten lesen, schreiben und löschen.
- Unterhaltungsmanager: Kann sämtliche Daten lesen und nur neue Unterhaltungsbeiträge erstellen. Dabei hat er die Möglichkeit die Unterhaltungsbeiträge vor dem Hochzeitsmanager zu verstecken.
- Administrator: Ist für das ganze System verantwortlich. Backups, Dateien exportieren beziehungsweise importieren gehören ebenfalls zu seinen Aufgaben.

In 4.1 werden die Rollen genauer erläutert.

## *Person*

Eine Person besitzt einen Vornamen, einen Nachnamen, ein Geburtsdatum und eine eindeutige Identifikationsnummer, die für das System benötigt wird. Weiterhin werden Telefonnummern und E-Mail Adressen nicht nur mit einfachen Datentypen wie beispielsweise einem Text dargestellt, sondern als eigene Klasse, um eine Klassifizierung der einzelnen Nummern und Adressen innerhalb einer Person zu ermöglichen. Hierbei wird als Beziehungstyp eine Komposition verwendet, da eine Telefonnummer und/oder eine E-Mail Adresse nicht ohne die dazugehörige Person existieren kann. Jede Person besitzt 0 bis 5 E-Mail Adressen und 1 bis 5 Telefonnummern. Sobald eine Person angelegt wird, muss mindestens eine Telefonnummer bereitgestellt werden.

## *Telefonnummer*

Die Telefonnummer wird als eigenständige Klasse gelistet, um eine Klassifizierung zu ermöglichen. Neben der Nummer, die als Zahl gespeichert wird, wird die Klassifizierung gespeichert. Dabei sind folgende Klassifizierungsarten möglich: privat, geschäftlich, mobil und sonstige.

## *E-Mail*

Die E-Mail Adresse ist aufgebaut wie die Telefonnummer. Die Adresse wird als Text gespeichert. Als Klassifizierung gibt es folgende Arten: privat, geschäftlich und sonstige.

## *Standesamtliche Trauung*

Zu jeder Hochzeit wird eine obligatorische standesamtliche Trauung hinzugefügt, da es sich ansonsten nicht um eine Hochzeit handelt. Diese Aktivität ist im Analyseklassendiagramm nicht Bestandteil der zusätzlichen Aktivitätsliste, da diese Aktivität besonders wichtig ist und weniger Attribute besitzt als eine normale Aktivität. Jede standesamtliche Trauung findet an einer bestimmten Lokalität statt und gehört zu genau einem Hochzeitsprojekt. Es werden eine Braut, ein Bräutigam und eine trauende Person benötigt. Weitere Teilnehmer sind optional und werden vom Hochzeitsmanager hinzugefügt. Außerdem besteht eine Verbindung zur Verpflegung und zur Hilfsmittelplanung. Diese sind optional, da eine standesamtliche Trauung auch ohne diese stattfinden kann.

Der Zustand ist ein Attribut der standesamtlichen Trauung und kann folgende Werte annehmen:

- In Arbeit: Der Prozess der Planung ist noch am Laufen und es müssen noch Sachen erledigt werden.
- Geplant: Die Trauung ist fertig geplant und es werden keine weiteren Änderung benötigt.

- Aktiv: Während die standesamtliche Trauung stattfindet, wird der Zustand auf aktiv gesetzt. Dies wird vom System gemacht und der User hat dazu keine Möglichkeit.
- Beendet: Nachdem die standesamtliche Trauung beendet ist, wird der Zustand auf beendet gesetzt. Dies wird ebenfalls vom System gemacht und der User hat nicht die Möglichkeit diesen Zustand zu setzen.

### *Kirchliche Trauung*

Die kirchliche Trauung umfasst dieselben Attribute und Beziehungen wie die standesamtliche Trauung. Im Klassendiagramm wurde nur eine von beiden eingezeichnet, um die Übersichtlichkeit zu bewahren. Für eine detaillierte Beschreibung siehe „Standesamtliche Trauung“.

### *Lokalität*

Das Objekt Lokalität stellt eine reelle Lokalität dar. Dabei wird die komplette Adresse gespeichert. Die Lokalität besitzt ein optionales Attribut Buchungsstatus, welches folgende Werte annehmen kann:

- Geplant: Es ist geplant die Lokalität zu buchen, jedoch wurde noch keine Aktion vorgenommen, um die Lokalität zu reservieren.
- Reserviert: Die Lokalität wurde reserviert und es ist noch keine feste Zusage vorhanden.
- Gebucht: Die Lokalität wurde gebucht und die Kosten stehen auch schon fest.
- Storniert: Die Lokalität wurde bereits reserviert oder gebucht und wurde storniert. Dies ist vor allem sinnvoll, wenn bereits Kosten angefallen sind, um sie in den Gesamtkosten festzuhalten.

### *Aktivität*

Ein Hochzeitsprojekt besteht aus beliebig vielen Aktivitäten. Alle Tätigkeiten innerhalb eines Hochzeitsprojekts sind als Aktivitäten definiert. Jede Aktivität besitzt ein Attribut „Unterhaltsbeitrag“, welches ein Boolean ist und angibt, ob diese Aktivität von einem Unterhaltungsmanager angelegt wurde oder nicht. Wurde sie von einem Unterhaltungsmanager angelegt, so wird das Attribut auf „wahr“ gesetzt. Es gibt ein weiteres Attribut „Geheim“, welches ebenfalls ein Boolean ist und angibt, ob es sich um eine geheime Aktivität handelt. Geheime Aktivitäten können nur von Unterhaltungsmanager angelegt werden und werden dem Hochzeitsmanager nicht angezeigt. Soll die Aktivität dem Hochzeitsmanager nicht angezeigt werden, so wird der Wert des Attributes auf „wahr“ gesetzt. Außerdem ist es möglich den Zustand einer Aktivität festzulegen. Dabei kommen folgende Zustände in Frage:

- In Arbeit: Der Prozess der Planung ist noch am Laufen und es müssen noch Sachen erledigt werden.
- Geplant: Die Trauung ist fertig geplant und es werden keine weiteren Änderung benötigt.
- Aktiv: Während die standesamtliche Trauung stattfindet, wird der Zustand auf aktiv gesetzt. Dies wird vom System gemacht und der User hat dazu keine Möglichkeit.
- Beendet: Nachdem die standesamtliche Trauung beendet ist, wird der Zustand auf beendet gesetzt. Dies wird ebenfalls vom System gemacht und der User hat nicht die Möglichkeit diesen Zustand zu setzen.
- Abgebrochen: Wurde eine Aktivität angefangen und soll doch nicht zum Einsatz kommen, so kann sie auf abgebrochen gesetzt werden, wenn man die Aktivität nicht löschen will. Dies ist vor allem sinnvoll, wenn bereits Kosten angefallen sind. Damit werden die Kosten ebenfalls unter den Gesamtkosten gelistet.

Eine Aktivität kann an keiner oder einer Lokalität stattfinden. Diese Option besteht um bei kleinen Aktivitäten wie zum Beispiel „Einladungen schreiben“ nicht einen Ort angeben zu müssen. Auch bei weiteren Aktivitäten wie zum Beispiel „Einkaufen“ muss dadurch keine Lokalität definiert werden. Will man dennoch eine Aktivität an mehreren Lokalitäten stattfinden lassen, so muss für jede Lokalität eine eigene Aktivität angelegt werden.

Zu jeder Aktivität gehört eine verantwortliche Person. Es können zu jeder Aktivität beliebig viele Teilnehmer eingeladen werden. Damit ist es möglich, bei der standesamtlichen Trauung nur den engsten Kreis dabei zu haben und bei der Feier alle Freunde und Bekannte einzuladen.

Zu einer Aktivität können beliebig viele Medien hinzugefügt werden. Um eine Aktivität mit verschiedenen Hilfsmitteln auszustatten, wird die Hilfsmittelplanung gebraucht. Eine Aktivität kann beliebig viele Hilfsmittelplanungen besitzen.

### *Hilfsmittelplanung*

Die Hilfsmittelplanung wird benötigt um Hilfsmittel einer Aktivität hinzuzufügen. Die Menge bezieht sich auf die Anzahl der Hilfsmittel die geliefert werden. Pro Hilfsmittelplanung ist nur ein Hilfsmittel erlaubt. Jede Hilfsmittelplanung hat eine zuständige Person und kann Kosten haben. Kosten können anfallen, wenn es sich bei den Hilfsmitteln um externe Hilfsmittel handelt oder Hilfsmittel, die gekauft werden müssen.

## *Hilfsmittel*

Hilfsmittel sind Gegenstände, die einem bei der Ausführung einer Aktivität helfen. Hilfsmittel müssen über die Hilfsmittelplanung einer Aktivität hinzugefügt werden. Dies kommt daher, dass für jedes Hilfsmittel nur ein Objekt existiert. Die Anzahl der benötigten Hilfsmittel wird in der Hilfsmittelplanung gesteuert. Dadurch ist es möglich, eine Liste mit allen Hilfsmitteln anhand der Objekte anzuzeigen, ohne die Anzahl beachten zu müssen.

## *Kosten*

Die Kosten können verschiedenen Objekten zugewiesen werden. Dabei kann der Kostenposten angegeben werden. Jedes Objekt besitzt einen Status, der folgende Werte annehmen kann:

- Geplant: Die Kosten sind geplant und sollen demnächst bezahlt werden. Dabei können die tatsächlichen Kosten noch steigen oder fallen.
- Beahlt: Die Kosten wurden bereits bezahlt.

Die Werte wurden so benannt, um mit den Kostennamen der Aktivitäten übereinzustimmen. Damit können die geplanten und bezahlten Kosten einer Aktivität festgestellt werden. Zu den Kosten können auch Medien hochgeladen werden. Dabei handelt es sich vorzugsweise um den Beleg.

## *Medium*

Ein Medium kann einer Aktivität oder den Kosten zugewiesen werden. Dabei werden der Titel, der Dateipfad, der Datentyp und die Id als Attribute verwendet. Als Dateityp kann jeder beliebige Typ angegeben werden. Der Dateipfad gibt den Pfad zu der gespeicherten Ressource auf dem Server an.

## *Caterer*

Der Caterer besitzt ein Attribut "istKommerziell", mit dessen Hilfe angegeben wird, ob es sich bei dem Caterer um einen kommerziellen Caterer handelt. Jeder Caterer besitzt eine Telefonnummer, eine E-Mail Adresse und eine Kontaktperson. Außerdem kann jeder Caterer einen Standort besitzen. Ein Caterer kann beliebig viele Verpflegungen bereitstellen.

## Verpflegung

Mithilfe der Verpflegung werden das Essen und die Getränke pro Caterer gehandhabt. Über die Getränkeplanung werden die Getränke und über die Essensplanung wird das Essen organisiert. Dabei wird für jedes Getränk eine neue Getränkeplanung erstellt und für jedes Essen eine neue Essensplanung.

Jede Verpflegung kann beliebig viele Rechnungen besitzen. Eine Verpflegung gehört zu genau einer Aktivität und eine Aktivität kann eine oder mehrere Verpflegung besitzen.

## Essensplanung

In der Essensplanung werden die Menge, die Art und die Stückanzahl gespeichert. Die Menge gibt immer die Menge pro Essenseinheit an. Die Art kann zum Beispiel Gramm, Kilogramm und Stück entsprechen. Die Menge ist nicht erforderlich, wenn der Typ „Stück“ ist, ansonsten ist dieser stets anzugeben. Die Stückanzahl gibt an, wie viele Entitäten geliefert werden sollen. Um dies zu verdeutlichen dient die nachfolgende Tabelle mit Beispielen.

Essen	Art	Menge	Stückanzahl
Schnitzel	Gramm	400	250
Schnitzel	Kilogramm	0,4	250
Schnitzel	Stück	-	250

Die Essensplanung besitzt genau ein Essen und gehört zu genau einer Verpflegung.

## Essen

Das Essen enthält den Namen des Essens.

Es gehört zu keinem oder beliebig vielen Essensplanungen.

## Getränkeplanung

Die Getränkeplanung ist genauso aufgebaut wie die Essensplanung, nur mit Getränken. Siehe dazu „Essensplanung“.

## *Getränk*

Ein Getränk besitzt das Attribut “enthältAlkohol”. Dabei handelt es sich um einen Boolean und gibt an, ob in dem Getränk Alkohol enthalten ist. Ein Getränk kann zu keinem oder beliebig vielen Getränkeplanungen gehören.

## 3.2. Verwendete Analysemuster

Im Analyseklassendiagramm wurden einige Analysemuster verwendet um die Objekte der Lastenheftanalyse zu klassifizieren und zu modellieren. Analysemuster sind bewährte, schematische Lösungen für eine Menge verwandter Probleme und bieten sich bei der Modellierung an, um die Übersichtlichkeit des Diagramms zu fördern und die Semantik zu unterstützen.

### 3.2.1. Liste

Eine Liste wird dadurch definiert, das ein Ganzes aus gleichartigen Teilen besteht, die diesem Ganzen fest zugeordnet sind. Die Modellierung erfolgt durch die Komposition. Im Analyseklassendiagramm der Hochzeitsplanung findet man dies bei der Person und der Telefonnummer. Der Person sind die Telefonnummern fest zugewiesen und bilden eine Liste, da bis zu fünf Telefonnummern zugewiesen werden können.

### 3.2.2. Baugruppe

Bei einer Baugruppe wird das Ganze in viele Teile aufgeteilt. Es herrscht eine existenzabhängige Zugehörigkeit der Teile. Man ist nur über das Ganze in der Lage auf die Teile zugreifen zu können und die Teile können unterschiedlichen Typs sein. Bei uns tritt dies bei der Verpflegung auf. Die Verpflegung stellt das ganze dar und die Essensplanung und Getränkeplanung die einzelnen Teile. Durch die Komposition wird die Existenzabhängigkeit der Teile ausgedrückt.

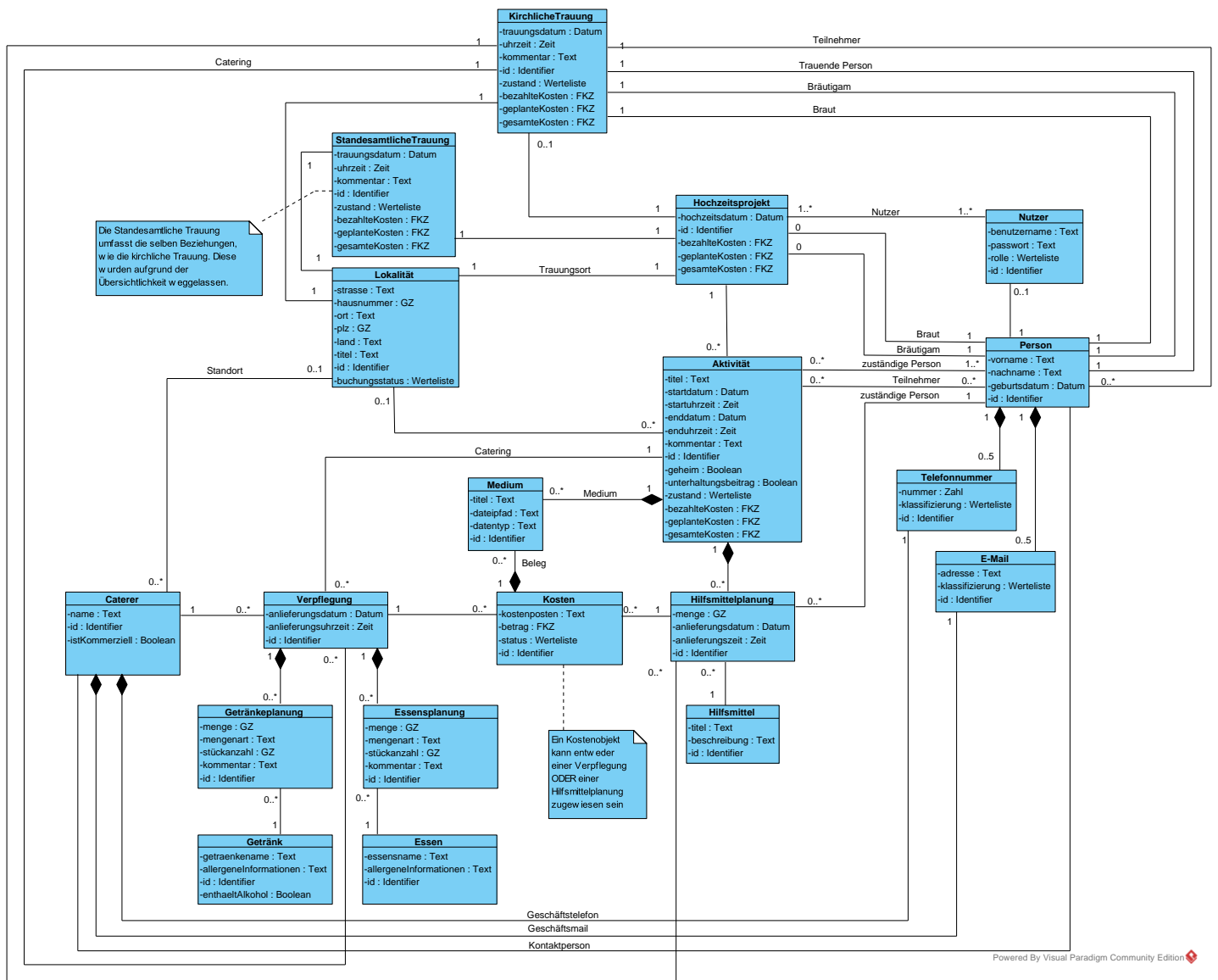
### 3.2.3. Rollen

Rollen werden verwendet, wenn eine Klasse in Bezug auf eine andere Klasse unterschiedliche Rollen einnehmen kann. Dies kann zur gleichen Zeit passieren. Bei uns gehört dies zu den am Häufigsten verwendeten Analysemustern. Zum Beispiel wurde dies bei der standesamtlichen Trauung verwendet. Zu der standesamtlichen Trauung gehören mehrere Personen. Jede dieser Personen hat eine andere Rolle. In unserem Fall sind das die Braut, der Bräutigam und die trauende Person. Es können auch optional noch Teilnehmer als Personen hinzugefügt werden.



### 3.2.4. Gruppe

Eine Gruppe stellt die Verbindung zwischen zwei Objekten dar. Dabei wird ausgedrückt, dass mehrere Einzelobjekte zu einem Zeitpunkt zum selben Gruppenobjekt gehören. Jedes Einzelobjekt kann maximal zu einer Gruppe gehören und eine Gruppe kann ggf. auch ohne Einzelobjekt existieren. Die Objektverbindungen können beliebig auf- und abgebaut werden. Diesen Fall haben wir bei der Verbindung zwischen Hochzeitsprojekt und Aktivität. Ein Hochzeitsprojekt kann keine oder mehrere Aktivitäten zu einem Zeitpunkt haben und eine Aktivität gehört zu einem Hochzeitsprojekt.



## 4. Use-Case-Diagramm

Die folgende Use-Case-Analyse visualisiert den kompletten Funktionsumfang der Hochzeitsplanungssoftware. Dabei werden alle Anwendungsfälle der Software und deren Zusammenhänge in unterschiedlichen Diagrammen übersichtlich dargestellt und mit den Akteuren der Hochzeitsverwaltungssoftware in Beziehung gesetzt. Zusätzlich werden die Besonderheiten aller Anwendungsfälle näher erläutert.

### 4.1. Rollenerläuterung der Anwendung

Die Hochzeitsverwaltungssoftware besitzt insgesamt vier unterschiedliche Rollen. Diese werden im Folgenden betitelt und näher erläutert.

#### 4.1.1. Standard-Nutzer (Pseudorolle)

Die Hochzeitsverwaltungssoftware besitzt einen Standard-Nutzer, welcher den Basisakteur jedes Anwenders repräsentiert. Dieser umfasst Funktionen, welche durch jeden anderen Nutzer ebenfalls ausgeführt werden können. Diese sind zum Beispiel sämtliche Standardaktionen, die das Anzeigen, Bearbeiten und Löschen von Aktivitäten umfassen. In der Regel wird einem Endnutzer jedoch nie diese Nutzerrolle zugewiesen. Daher wird sie ebenfalls als Pseudorolle betitelt.

#### 4.1.2. Hochzeitsmanager

Der Hochzeitsmanager spiegelt die Nutzerrolle der Person wider, die die gesamte Hochzeit organisiert. Dies kann zu einem das Brautpaar sein, oder eine dritte, durch das Brautpaar beauftragte Person. Weiterhin kann die Rolle des Hochzeitsmanager ebenfalls von mehreren Benutzern genutzt werden. Somit ist gewährleistet, dass der Bräutigam, sowie die Braut einen eigenen Zugang zum System besitzen und sie sich zusätzlich durch eine dritte Person beraten lassen können. Der Hochzeitsmanager besitzt in dem eigenen Hochzeitsprojekt, welches für ihn erstellt wurde administrative Rechte und kann daher sämtliche Funktionalitäten frei bedienen. Darunter fallen Aktionen wie das Bearbeiten und Löschen von Projektstammdaten und Aktivitäten, sowie der Im- und Export von Daten.

### 4.1.3. Unterhaltungsmanager

Der Unterhaltungsmanager umfasst sämtliche Funktionen, die sich um die Organisation von Aktivitäten handeln. Dazu gehört beispielsweise das Anzeigen, Bearbeiten und Löschen von Unterhaltungsaktivitäten, sowie deren Abhängigen Entitäten. Abhängige Entitäten sind Datenobjekte, die unmittelbar mit der aktuellen Aktivität verknüpft sind, wie zum Beispiel das Catering, Teilnehmer oder auch Hilfsmittel. Der Unterhaltungsmanager kann jedoch keine Funktionen bedienen, die auf Stammdaten der Anwendung zugreifen. Diese umfassen die Bearbeitung von der standesamtlichen und kirchlichen Trauung, Benutzerdaten und Autorisierungen, sowie das Im- oder Exportieren von Datensätzen. Diese Funktionen obliegen der Nutzerrolle Hochzeitsmanager. Dennoch bietet die Nutzerrolle Unterhaltungsmanager eine Funktion, auf die der Hochzeitsmanager keinen Zugriff hat. Durch die Aktion „Aktivität verstecken“ kann eine Aktivität, sowie deren abhängigen Entitäten für die Hochzeitsmanager versteckt werden. Dies ermöglicht das geheime Anlegen von Unterhaltungsbeiträgen (siehe 1.2.3).

### 4.1.4. Administrator

Als letztes gibt es die Nutzerrolle des Administrators der sämtliche Zugriffsrechte projektübergreifend hat. Dies bedeutet, dass er in allen Projekten lesenden und schreibenden Zugriff hat, sowie für die Verwaltung der zentralen Datenbank, sowie aller Nutzer verantwortlich ist. Besondere Aktionen des Administrators bilden die Datensicherung, sowie die Benutzerverwaltung.

## 4.2. Use-Case-Kompaktansicht

Das nachfolgende Use-Case Diagramm visualisiert die Gesamtübersicht der Hochzeitsplanungssoftware in einer kompakten Form. Hierbei werden die wichtigsten Use-Cases im Allgemeinen dargestellt. Diese splitten sich in weiteren Diagrammen zu detaillierteren Anwendungsfällen auf. Sämtliche Anwendungsfälle wurden farblich, anhand ihres Anwendungsgebiet innerhalb des Systems markiert. Hierbei wird zwischen Systemanmeldung (Login und Logout), Standardaktionen (Aktivitäten anzeigen, bearbeiten, suchen), Zusatzaktionen (Aktivität verstecken), Projektaktionen (Verwaltung der Projektstammdaten) und administrativen Aktionen (Datensicherung, Benutzerpflege, Import/Export) unterschieden. Folgend die Erläuterungen zu den einzelnen Aktionen.

### **Einloggen**

Anwendungsfall, der es dem Nutzer ermöglicht sich im System anzumelden. Hierbei werden nutzerspezifische Daten verlangt, wie beispielsweise Benutzername und Kennwort. Nach einer erfolgreichen Anmeldung im System kann der Nutzer weitere Anwendungsfälle nutzen, die für seine zugewiesene Rolle verfügbar sind.

### **Ausloggen**

Korrespondierendes Komplement zum Anwendungsfall Einloggen. Hierbei werden keine Nutzerspezifische Daten verlangt. Nach der Abmeldung kann der Nutzer keine weiteren Anwendungsfälle mehr nutzen. Ausnahme bildet der Anwendungsfall Einloggen.

### **Aktivität anzeigen**

Dieser Anwendungsfall dient der Visualisierung von einzelnen Aktivitätsdaten für den Nutzer. Hierbei kann sich der Nutzer detaillierte Informationen zu einer ausgewählten Aktivität anzeigen lassen.

### **Aktivitäten anzeigen**

Zeigt eine Liste oder einen Kalender, je nach präferierter Darstellungsform für den Nutzer an. Darin findet sich eine Sammlung von einzelnen Aktivitäten, die sortiert vorliegen. Hierbei werden nur wichtige Stammdaten einzelner Aktivitäten angezeigt. Bei der Auswahl einer Aktivität, tritt unmittelbar der Use-Case „Aktivität anzeigen“ in Kraft.

### **Aktivität bearbeiten**

Dieser Use-Case beschäftigt sich mit der Bearbeitung einer Aktivität. Hierdurch hat der Nutzer die Möglichkeit Änderungen innerhalb einer Aktivität vorzunehmen. Dies wird durch die Anzeige einer Formularseite ermöglicht. Für diesen Use-Case wird zusätzlich ein detailliertes Diagramm zur Verfügung gestellt.

### **Aktivität anlegen**

Korrespondierendes Äquivalent zu Aktivität bearbeiten. Hierbei sind lediglich sämtliche Formularfelder mit initialen Startwerten befüllt. Weiterhin wird in diesem Anwendungsfall eine neue Aktivität erstellt.

### **Unterobjekt anzeigen**

Äquivalent zu Aktivität anzeigen, hierbei bezieht sich das Tool jedoch lediglich auf Unterobjekte, wie beispielsweise Verpflegung, Hilfsmittel oder Person. Diese wurden zur Übersicht zu dem Stichwort Unterobjekt zusammengefasst.

### *Unterobjekte anzeigen*

Siehe Aktivität anzeigen. Als Visualisierungsoption steht hier eine Liste zur Auswahl.

### *Unterobjekt bearbeiten*

Siehe Aktivität bearbeiten.

### *Unterobjekt anlegen*

Siehe Aktivität anlegen.

### *Suchen*

Der Anwendungsfall Suchen beinhaltet einen komplexeren Anwendungsfall, der die allgemeine Suche und Darstellung von Aktivitäten und Unterobjekten repräsentiert. Dieser Anwendungsfall ist in einem weiteren Diagramm näher dargestellt.

### *Aktivität verstecken*

Anwendungsfall, der es einem Nutzer mit der Rolle Unterhaltungsmanager ermöglicht eine Aktivität zu verstecken. Dies bedeutet, dass eine Aktivität für die Rolle Hochzeitsmanager unsichtbar wird. Hierbei muss die Aktivität bearbeitet werden.

### *Hochzeit anlegen*

Durch den Use-Case Hochzeit anlegen hat der Nutzer die Möglichkeit ein neues Hochzeitsprojekt anzulegen. Dies ist in der Regel der initiale Schritt, den der Nutzer nach der ersten erfolgreichen Anmeldung im System unternimmt. Diese Funktion ist nur für den Hochzeitsmanager oder für einen Administrator verfügbar.

### *Hochzeit bearbeiten*

Der Anwendungsfall beinhaltet die Bearbeitung von Hochzeitsstammdaten, die unmittelbar mit dem Projekt in Verbindung stehen. Dies sind zum Beispiel die Angabe des Bräutigam und der Braut, das Trauungsdatum oder andere Hochzeitsdaten, deren Bearbeitung lediglich dem Hochzeitsmanager vorbehalten ist.

### *Hochzeit anzeigen*

Die Anzeige der Hochzeit visualisiert sämtliche wichtigen Stammdaten auf einen Blick. Die Anzeige darf auch durch den Standardnutzer durchgeführt werden, wie im Lastenheft gefordert (1.2.3).

### ***Daten importieren***

Daten sollen durch den Hochzeitsmanager oder durch einen Administrator in das bestehende Hochzeitsprojekt importiert werden. Der Anwendungsfall ermöglicht den Import.

### ***Daten exportieren***

Korrespondierendes Komplement zum Anwendungsfall Daten importieren.

### ***Rollen verwalten***

Der Anwendungsfall ermöglicht es dem Nutzer mit der Rolle Hochzeitsmanager anderen Nutzern spezielle Rollen zu verteilen. Diese können Unterhaltungsmanager oder ebenfalls Hochzeitsmanager sein. Der Administrator hat ebenfalls die Möglichkeit sämtliche Rollen an Nutzer zu vergeben. Die eigene mit eingeschlossen.

### ***Nutzer pflegen***

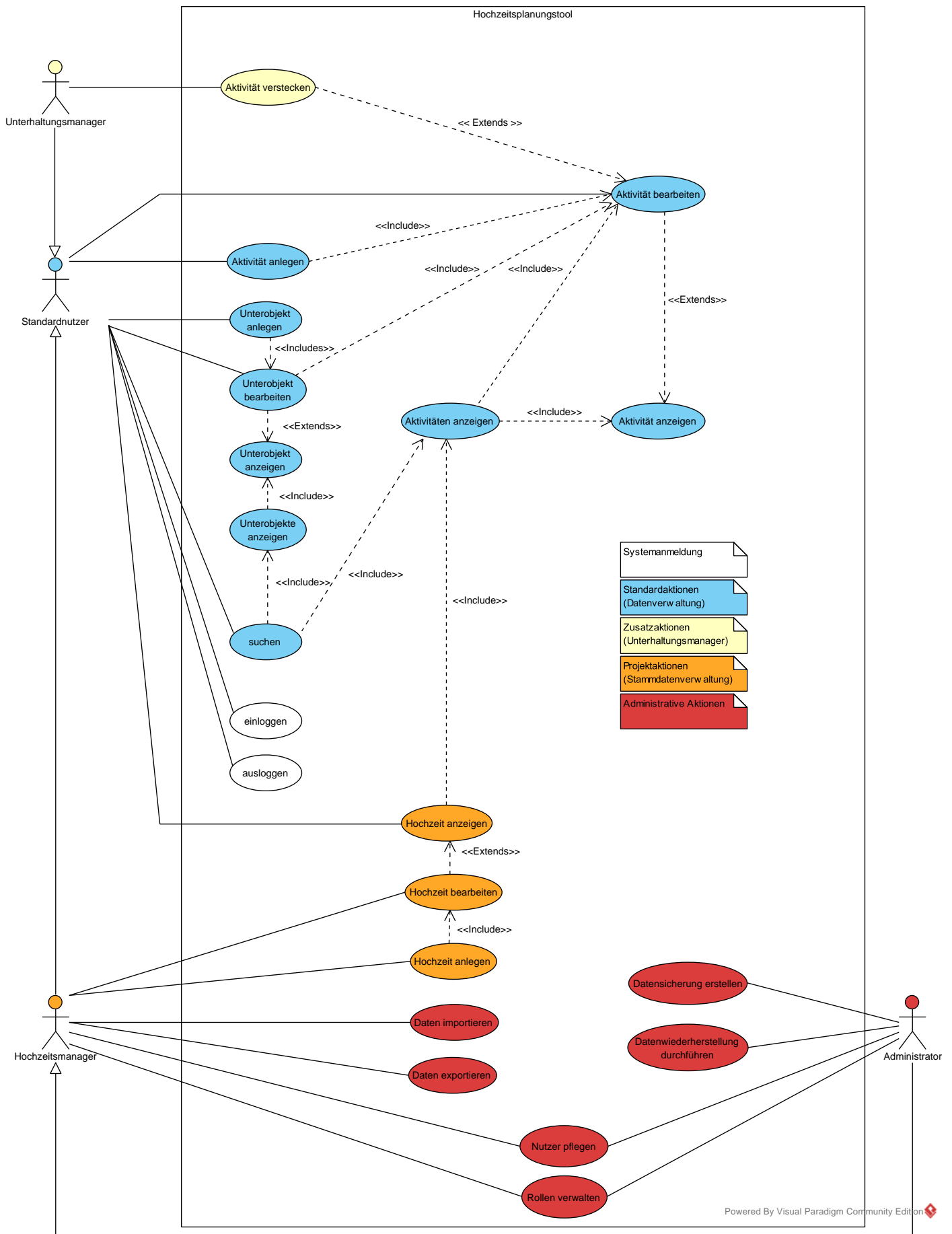
Sämtliche Nutzer innerhalb eines Systems müssen durch eine bestimmte Person gepflegt werden. Dies übernimmt die Rolle Administrator in diesem Anwendungsfall. Hierbei werden weitere Anwendungsfälle abgedeckt, wie beispielsweise das Zurücksetzen eines Passwortes oder das Ändern von Systemnutzerdaten.

### ***Datensicherung erstellen***

Der Administrator hat die Möglichkeit eine globale Datensicherung der Datenbank durchzuführen. Dies wird durch diesen Anwendungsfall abgedeckt.

### ***Datenwiederherstellung durchführen***

Korrespondierendes Komplement zu Datensicherung erstellen. Hierbei wird die Datensicherung im System eingespielt.





### 4.3. Use-Case Diagramm: Aktivität bearbeiten

Das Use-Case Diagramm „Aktivität bearbeiten“ stellt den dazugehörigen Use-Case und dessen Abhängigkeiten näher dar. Sämtliche Aktionen können durch die Rolle des Standardnutzers durchgeführt werden. Eine Ausnahme bildet der Anwendungsfall „Aktivität verstecken“, der zuvor bereits erläutert wurde.

#### *Aktivität anzeigen*

Um eine Aktivität zu bearbeiten, muss diese zuvor angezeigt werden. Dies bildet dieser Anwendungsfall ab.

#### *Detailseite anzeigen*

Der Anwendungsfall Detailseite anzeigen, beinhaltet die Visualisierung von einzelnen Stammdaten für eine entsprechende Entität. Hierbei wird eine Übersichtsseite für beispielsweise eine Aktivität erzeugt, die alle wesentlichen informativen Inhalte, sowie Listeninhalte besitzt.

#### *Liste anzeigen*

Liste anzeigen ist ein Anwendungsfall, der die Visualisierung einer Liste für eine Menge von Entitäten des gleichen Typs übernimmt. Hierbei besitzt ein übergeordnetes Objekt, wie zum Beispiel Aktivität eine Sammlung von Unterobjekten, wie zum Beispiel eine Liste an benötigten Hilfsmittel für die Aktivität. Dieser Use-Case stellt einen allgemeinen Anwendungsfall dar, da jede Visualisierung einer Liste ähnliche Elemente, wie beispielsweise einen Listenkopf, Listeneinträge oder ein Vertikales Layout besitzt.

#### *Hilfsmittel (plural) anzeigen*

Die Erweiterung des Anwendungsfalls Liste anzeigen. Hierbei werden spezielle Listen für Hilfsmittel angezeigt.

#### *Teilnehmer (plural) anzeigen*

Die Erweiterung des Anwendungsfalls Liste anzeigen. Hierbei werden spezielle Listen für Teilnehmer angezeigt.

#### *Verantwortliche (plural) anzeigen*

Die Erweiterung des Anwendungsfalls Liste anzeigen. Hierbei werden spezielle Listen für Verantwortliche angezeigt.

### ***Verpflegungen anzeigen***

Die Erweiterung des Anwendungsfalls Liste anzeigen. Hierbei werden spezielle Listen für Verpflegungen angezeigt.

### ***Hilfsmittel (singular) anzeigen***

Die Erweiterung des Anwendungsfalls Detailseite anzeigen. Hierbei wird eine spezielle Detailseite für ein konkretes Objekt des Typs Hilfsmittel angezeigt.

### ***Teilnehmer (singular) anzeigen***

Die Erweiterung des Anwendungsfalls Detailseite anzeigen. Hierbei wird eine spezielle Detailseite für ein konkretes Objekt des Typs Person angezeigt, die als Teilnehmer agiert.

### ***Verantwortlicher (singular) anzeigen***

Die Erweiterung des Anwendungsfalls Detailseite anzeigen. Hierbei wird eine spezielle Detailseite für ein konkretes Objekt des Typs Person angezeigt, die als Verantwortlicher agiert.

### ***Verpflegung anzeigen***

Die Erweiterung des Anwendungsfalls Detailseite anzeigen. Hierbei wird eine spezielle Detailseite für ein konkretes Objekt des Typs Verpflegung angezeigt.

### ***Aktivitätsdaten ändern***

Der Anwendungsfall Aktivitätsdaten ändern umfasst die Änderung einzelner Stammdateneinträge innerhalb einer Aktivität. Hierbei können einfache Attribute der Aktivität (keine Listen) bearbeitet werden.

### ***Listendaten löschen***

Listendaten löschen bildet einen allgemeinen Anwendungsfall ab, der dem Nutzer die Möglichkeit gibt, zugewiesene Einträge innerhalb einer Aktivität zu löschen. Hierbei wird jedoch nur die Referenz auf die Aktivität gelöscht, nicht jedoch das Listenobjekt an sich. Auf dieses kann über die Suche weiterhin zugegriffen werden. Beispiel: Aktivität besitzt Hilfsmittel Luftballon. Nach Löschen des Listenelements, gibt es in der Sammlung aller globalen Hilfsmittel nach wie vor den Luftballon, jedoch gibt es diesen nicht mehr in der Aktivität.

### *Listendaten hinzufügen*

Allgemeiner Anwendungsfall der das Hinzufügen von Listenelementen innerhalb einer Aktivität ermöglicht. Hierbei hat der Nutzer die Möglichkeit Elemente aus einer bestehenden globalen Liste zu referenzieren oder ein neues Unterobjekt anzulegen, welches ebenfalls referenziert wird.

### *Listendaten bearbeiten*

Der Anwendungsfall Listendaten bearbeiten bildet ebenfalls einen allgemeinen Anwendungsfall ab, der die Bearbeitung von Listen innerhalb einer Entität ermöglicht.

## **4.4. Use-Case Diagramm: Suchen**

Das Use-Case Diagramm „suchen“ stellt den Anwendungsfall einer Suche innerhalb des Systems dar. Hierbei wird insbesondere auf die Art der möglichen Suche, sowie deren Bearbeitung eingegangen. Hierbei kommt es vor allem darauf an, welches Objekt der Nutzer finden möchte. So können Aktivitäten innerhalb einer Kalenderansicht gesucht werden. Für alle anderen Objekte, aber auch ebenfalls für Aktivitäten, steht eine Listenübersicht zur Verfügung.

### *Listensuche auswählen*

In diesem Anwendungsfall wird explizit eine Liste als präferierte Darstellungsform des Ergebnisses gewählt. Dadurch wird dem Nutzer eine Suche für alle Entitäten gewährleistet, die in die globale Sammlung mitaufgenommen werden können.

### *Kalendersuche auswählen*

Hierbei wird explizit eine Kalenderansicht als präferierte Darstellungsform gewählt. Dadurch hat der Nutzer die Restriktion innerhalb dieser Suche nur nach Aktivitäten suchen zu können. Diese werden innerhalb eines Kalenders dargestellt.

### *Freitextsuche ausfüllen*

Dieser Use-Case repräsentiert die Eingabe von gesuchten Stichwörtern, um bestimmte Objekte zu finden. Die Freitextsuche ist in beiden Ansichten (Liste und Kalender) verfügbar. Entsprechende Treffer werden in der Liste oder im Kalender angezeigt. Der Rest wird herausgestrichen.

### ***Kategoriefilter setzen***

In der Sicht der Listensuche können Filteroptionen für alle Entitäten gesetzt werden, die in die globale Objektsammlung mitaufgenommen werden können, sowie Aktivitäten. Hierbei repräsentiert der Anwendungsfall Kategoriefilter setzen das Speichern der ausgewählten Filteroption.

### ***Hilfsmittelfilter setzen***

Der Anwendungsfall Hilfsmittelfilter setzen ist ein konkreter Fall für das Setzen der Filteroption Hilfsmittel.

### ***Catererfilter setzen***

Der Anwendungsfall Catererfilter setzen ist ein konkreter Fall für das Setzen der Filteroption Caterer.

### ***Aktivitätsfilter setzen***

Der Anwendungsfall Aktivitätsfilter setzen ist ein konkreter Fall für das Setzen der Filteroption Aktivität.

### ***Datumssuche setzen***

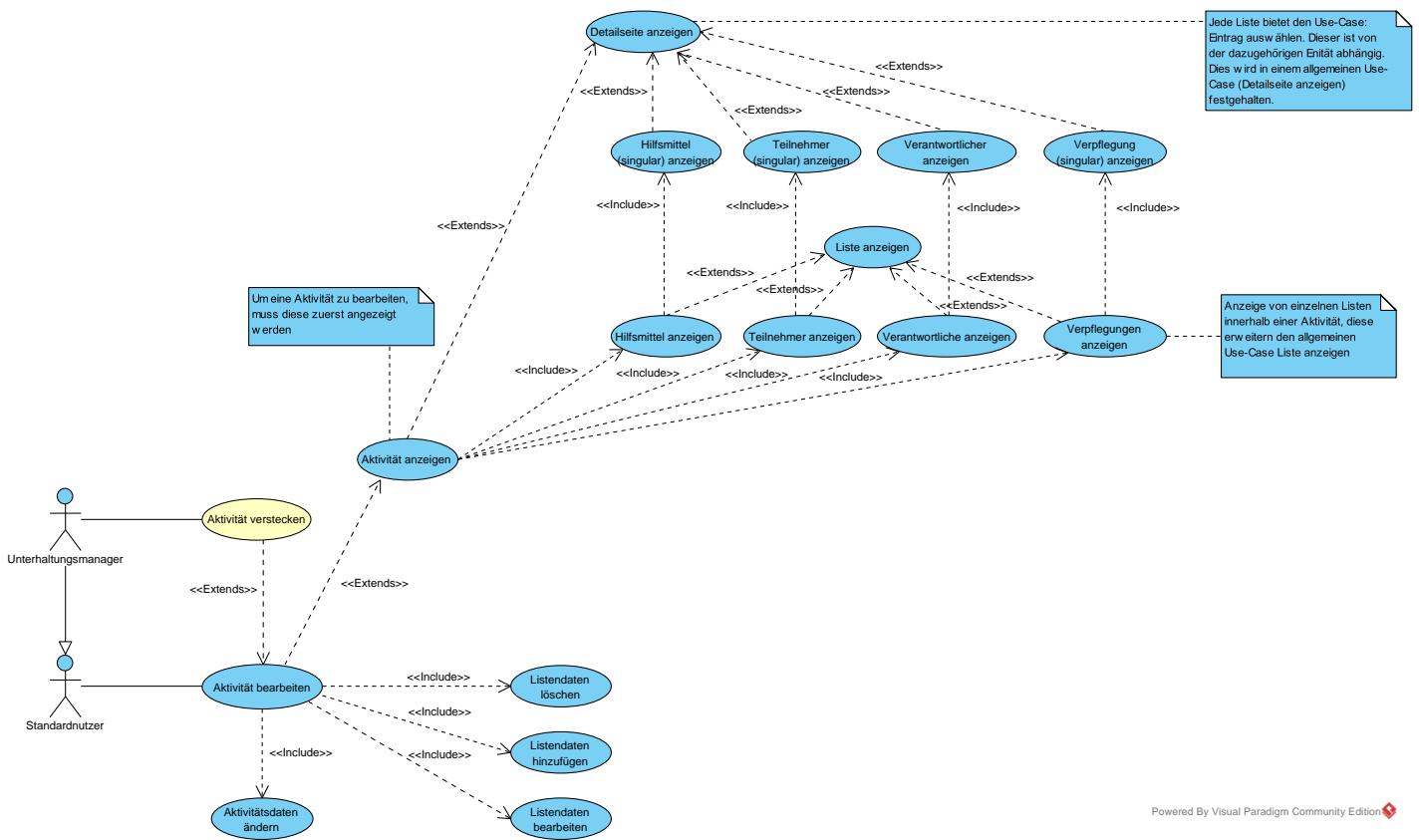
In der Ansicht der Kalendersuche kann zusätzlich eine Datumssuche aktiviert werden. Hierbei hat der Nutzer die Möglichkeit Aktivitäten innerhalb einer zeitlichen Spanne hervorzuheben.

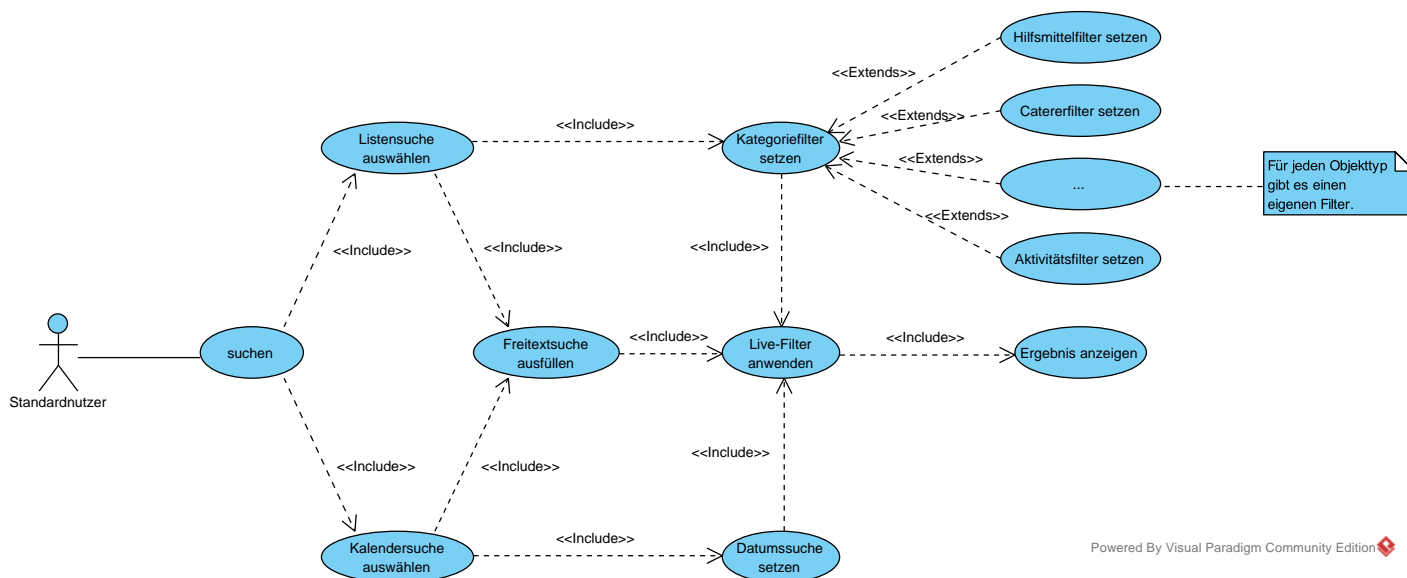
### ***Live-Filter anwenden***

Bei jeder Eingabe des Users innerhalb des Suchfeldes, der Datumssuche oder beim Setzen eines Filters wird direkt die Suche gestartet. Somit sprechen wir von einer Live-Suche, die durch den Nutzer indirekt getriggert werden kann.

### ***Ergebnis anzeigen***

Die Anwendung eines Live-Filters inkludiert nach einer erfolgreichen Suche den Anwendungsfall Ergebnis anzeigen. Hierbei wird je nach gewählter Suchansicht das Ergebnis in der Liste oder dem Kalender angezeigt.





## 5. Sequenzdiagramm

Im folgenden Abschnitt wird das Szenario „Catering anlegen“, sowie das exemplarische Unterszenario „Essen anlegen“ erläutert und anschließend durch ein Sequenzdiagramm visualisiert. Die Beschreibung des Diagramms dient der Auflistung von Besonderheiten und dient ebenfalls zur Bezugsherstellung. Die Darstellung des Sequenzdiagramms findet in der Phase der Analyse statt, sodass einige Funktionen oder Klassennamen allgemeingültig ausgedrückt werden. Dies ist bewusst so gemacht, um bei dem späteren Entwurf des Klassendiagramms an keine Restriktionen gebunden sein zu müssen. Weiterhin wurde auf die exakte Angabe von Parameterwerten, sowie auf die Rückgabe typkonformer Daten bewusst verzichtet. Diese wurden im vollen Umfang allgemeingültig dargestellt. Die Verlauf der Nachrichten, sowie deren beteiligten Objekte stehen im Vordergrund der Szenariobetrachtung. Dies dient als Grundlage zum weiteren Entwurf.

### 5.1. Szenariobetrachtung: Catering anlegen

In diesem Szenario wird für eine bestimmte Aktivität ein Catering angelegt. Die korrespondierende Klasse im Klassendiagramm ist Verpflegung. Daher wird im späteren Ablauf des Sequenzdiagramms auch von dem Anlegen einer Verpflegung gesprochen. Dies ist mit Catering gleichzusetzen.

Das Szenario betrachtet das Anlegen eines Verpflegungsobjekts, welches für eine spätere Aktivität genutzt werden kann. Hierbei werden unter anderem Speisen und Getränke hinzugefügt, um den Teilnehmern der Aktivität eine Mahlzeit oder etwas zu trinken zu bieten. Das neue Anlegen einer Verpflegungseinheit benötigt hierzu wichtige obligatorische Fakten. Unter anderem muss ein eindeutiger Caterer für das Catering verantwortlich sein. Daher muss diese obligatorisch eingetragen werden. Hierbei kann jedoch ebenfalls auf eine private Person zurückgegriffen werden. Weiterhin hat man die optionale Möglichkeit Datum und Uhrzeit für das Eintreffen des Caterings zu hinterlegen. Nachdem der Nutzer eine Liste an diversen Speisen und eine Liste an Getränken hinzugefügt hat, müssen anschließend noch die Kosten für das Catering gepflegt werden. Nach der Speicherung durch den Nutzer kann das neu angelegte Catering Objekt auf der Datenbank gespeichert werden.

## 5.2. Diagrammbetrachtung: Catering anlegen

Das oben beschriebene Szenario wird durch das nachfolgende Sequenzdiagramm nach UML Standard visuell dargestellt. Hierbei wurde auf die explizite Darstellung von Fehlerbenachrichtigungen, wie zum Beispiel Datenbankfehler, fehlende Berechtigungen oder falsche Nutzereingaben, zum Gunsten der Übersichtlichkeit verzichtet. Exemplarisch wurde eine Fehlerbehandlung beim Sichern einer Verpflegung eingefügt.

Weiterhin beinhaltet das Szenario viele weitere Unterszenarien, wie beispielsweise das neue Anlegen von Caterern, Speisen, Getränken oder Kosten. Diese wurden an der entsprechenden Stelle als eigenes Sequenzdiagramm referenziert. Exemplarisch wurde hierfür ein Sequenzdiagramm für das Objekt Essen angelegt. Dieses visualisiert das Unterszenario „Essen anlegen“ und dient als Vergleichsdiagramm für die anderen referenzierten Szenarien, da diese sich stark ähneln.

Viele in dem Diagramm dargestellte Funktionen befinden sich in einem allumfassenden Parallelitätsblock, welcher dem Nutzer die freie Pflege der Attribute ermöglicht. Dadurch hat der Nutzer die Möglichkeit die Eingabe für spezielle Felder und Listen, wie beispielsweise Datum, Uhrzeit oder Speisen in einer beliebigen Reihenfolge auf der Formularseite abzuarbeiten. Weiterhin wurden optionale Szenarien innerhalb des Sequenzdiagramms durch entsprechende Optional-Blöcke dargestellt. Diese müssen vom Nutzer nicht betreten werden, um einen erfolgreichen Abschluss des Szenarios zu gewährleisten.

Alternative Blöcke geben dem Nutzer in diesem Diagramm eine Wahl zwischen zwei unterschiedlichen Aktionen. So kann er in der Regel zwischen dem neuen Anlegen eines Unterobjekts oder der Auswahl eines bestehenden Objekts entscheiden, wenn er beispielsweise die Getränkliste des Caterings pflegt. Dies kann der Nutzer so lange wiederholen, solange er noch weitere Getränke auf seiner Liste wünscht. Daher wird die Bedingung in den dazugehörigen Schleifenblöcken ebenso abstrakt gehalten, da dies eine vom Nutzer intuitive Entscheidung ist.

Generell besitzt das Sequenzdiagramm insgesamt elf unterschiedliche Entitätstypen, deren Namen aus dem Analyseklassendiagramm resultieren und daher noch abstrakt gehalten werden. Das Szenario beginnt durch eine vom Nutzer ausgelöste Aktion, indem er eine neue Verpflegung anlegen möchte. Hierbei muss als Vorbedingung bereits eine bestehende Aktivität existieren, in welcher der Nutzer im Bearbeitungsmodus agiert. Erst durch diese Bedingung, kann es dem Nutzer ermöglicht werden, der Aktivität ein Catering hinzuzufügen. Durch diese Aktion wird eine neue Instanz eines Verpflegungsobjektes erstellt, welches das Catering repräsentiert. Hierbei wird ebenfalls eine



eindeutige ID für das entsprechende Objekt gesetzt. Auf einen Zugriff auf die Datenbank um die nächste, noch nicht vergebene ID zu finden, wird hier aus Gründen der Übersichtlichkeit verzichtet. Nach der erfolgreichen Erstellung des Objekts werden die Formulardaten des Objekts abgefragt, um eine entsprechende Formularseite für den Nutzer anzuzeigen. Hierbei wird flexibel auf die bestehenden Attribute des Verpflegungsobjekts reagiert und eine passende Seite erzeugt.

Nachdem der Nutzer nun eine Möglichkeit besitzt sämtliche benötigten Attribute innerhalb der Seite zu pflegen, steht es ihm frei, welche er zuerst bearbeiten möchte. Bei der Pflege von einfachen Feldvariablen, wie beispielsweise Datum oder Uhrzeit des eintreffenden Caterings, werden die eingegebenen Werte in die entsprechenden Attribute des Verpflegungsobjektes geschrieben. Eine Validierung der Eingabewerte findet statt, jedoch wurde hierbei auf die Darstellung im Sequenzdiagramm verzichtet.

Bei der Pflege von komplexeren Datenobjekten, wie beispielsweise Caterer, Essen oder Getränk, erhält der Nutzer eine jeweilige Vorschlagsliste, basierend auf den bestehenden Datensätzen in der Datenbank. Hierbei hat der Nutzer die freie Möglichkeit einen bestehenden Datensatz auszuwählen und diesen mit einer entsprechenden Planung zu verbinden. Diese Essens- oder Getränkeplanung wird anschließend auf das Catering referenziert. Bei der Auswahl von Speisen und Getränken muss der Nutzer nach der Auswahl noch eine zu bestellende Menge hinterlegen, sodass die entsprechende Getränke- oder Essensplanung vollständig ist. Optional kann der Nutzer noch ein Kommentar bei der Planung einfügen. Bei den Kosten entfällt die Zwischenstufe der Planung, da hierbei auf keine in der Datenbank stehenden Objektdaten zurückgegriffen wird. Hierbei gibt es keine Vorschlagsliste und jedes Kostenobjekt muss neu erzeugt werden. Dieses wird direkt als Listenelement auf das Catering referenziert.

Nachdem der Nutzer sämtliche Eintragungen zu seinem gewünschten Catering vorgenommen hat, hat dieser die Möglichkeit das aktuelle Catering zu sichern oder zu verlassen. Tätigt der Nutzer die Aktion „sichern“, so werden alle Daten des Objekts „Verpflegung“ gesammelt und an den Datenbankkonnektor gesendet, der die Verbindung zur Datenbank aufbaut und die entsprechenden Datensätze sichert. Exemplarisch werden hier Fehlerbehandlungen, sowie die Interaktion zwischen Benutzeroberfläche und Nutzer näher dargestellt. Da das Speichern auf der Datenbank als asynchrone Nachricht ausgeführt wird, wird dem Nutzer danach direkt eine Art Sanduhr angezeigt, die diesem eine kurze Wartezeit signalisiert. Nach einer Antwort durch den Datenbankkonnektor wird die Sanduhr wieder entfernt und ein Dialog geöffnet, abhängig von der Statusmeldung, die die Datenbank zurückliefert. Sofern alles in Ordnung ist, wird ein Dialog angezeigt, der dem User das erfolgreiche Speichern signalisiert. Andernfalls wird ein Fehlerdialog geöffnet, der den Nutzer auf eventuelle Fehler

hinweist. Nach Schließen des Dialogfensters, egal ob erfolgreich oder erfolglos, kehrt der Nutzer zur Detailseite seiner eben angelegten Verpflegung zurück und kann diese nun verlassen. Hierbei wird er nochmals gefragt, ob er die Seite verlassen möchte, denn andernfalls würden nicht gespeicherte Änderungen verloren gehen. Nach positiver Bestätigung des Dialogs kehrt der Nutzer zur allgemeinen Hochzeitsprojektübersichtsseite zurück und das Verpflegungsobjekt wird zerstört.

### 5.3. Pseudocode: Catering anlegen

Bei dem Pseudocode ist zu beachten, dass an zwei Stellen der Datenbankkonnektor aufgegriffen und vertieft wurden. An diesen Stellen wurden Hinweise hinterlegt. Die Vertiefungen der Programmabfolgen sind unter der Hauptprogrammabfolge zu finden.

```
-----
- PSEUDOCODE SEQUENZDIAGRAMM      -
- CATERING ANLEGEN                 -
-----
```

Pseudocode der Klasse Main,  
EVENT "ClickVerpflAnlegen" wurde ausgelöst:

```
ERSTELLE VERPFLEGUNGSOBJEKT der Klasse VERPFLEGUNG;
ERMITTLE nötige FORMULARDATEN des VERPFLEGUNGSOBJEKTS;
SENDE FORMULARDATENLISTE an UI;
```

```
UI: ERSTELLE FORMULARSEITE;
UI: ZEIGE FORMULARSEITE;
WARTe auf NUTZEREINGABE;
EVALUIERE NUTZEREINGABE;
```

```
WENN NUTZEREINGABE = DATUM;
    DANN
```

```
        EMPFANGE DATUM von UI;
//Auf die Validierung der Nutzereingaben wird hier zu Gunsten
//der Übersichtlichkeit verzichtet. Diese muss jedoch stattfinden!
```

```
        SCHREIBE DATUM in VERPFLEGUNGSOBJEKT;
```

```
    SONST WENN NUTZEREINGABE = UHRZEIT;
```

```
        DANN
```

```
            EMPFANGE UHRZEIT von UI;
```

```
            SCHREIBE UHRZEIT in VERPFLEGUNGSOBJEKT;
```

```
    SONST WENN NUTZEREINGABE = CATERER hinzufügen;
```

```
        DANN
```

```
            SUCHE CATERER auf DATENBANK;
```

```
//Datenbankkonnektor nimmt die Verbindung zur Datenbank auf
```

```
            EMPFANGE CATERERLISTE von DATENBANKKONNEKTOR;
```

```
            SENDE CATERERLISTE an UI;
```

```
            UI: ZEIGE CATERERLISTE;
```

```
            WARTe auf NUTZEREINGABE;
```

```
            WENN NUTZEREINGABE = CATERER auswählen;
```

```
                DANN
```

```
                    EMPFANGE ausgewählten CATERER;
```

```

        SCHREIBE CATERER in VERPFLEGUNGSOBJEKT;
    SONST WENN NUTZEREINGABE = CATERER anlegen;
        DANN
            ERSTELLE CATEREROBJEKT der Klasse CATERER;
//Dieser Anwendungsfall wird im Pseudocode nicht
//näher verfeinert.
        SCHREIBE neuen CATERER in VERPFLEGUNGSOBJEKT;
    SONST WENN NUTZEREINGABE = ESSEN hinzufügen;
        DANN
            SENDE SUCHANFRAGE an DATENBANKKONNEKTOR;
// Exemplarischer Pseudocode siehe DATENBANKKONNEKTOR

    EMPFANGE ESSENSLISTE von DATENBANKKONNEKTOR;
    SENDE ESSENSLISTE an UI;
    UI: ZEIGE ESSENSLISTE;
    WARTEN auf NUTZEREINGABE;
    WENN NUTZEREINGABE = ESSEN auswählen;
        DANN
            EMPFANGE ausgewähltes ESSEN;
            SONST WENN NUTZEREINGABE = ESSEN anlegen;
                DANN
                    ERSTELLE ESSENSOBJEKT der Klasse ESSEN;
                    ERSTELLE ESSENSPLANUNGSOBJEKT der Klasse ESSENSPLANUNG;
                    SCHREIBE neue ESSENSPLANUNG in VERPFLEGUNGSOBJEKT;
//Hier wird eine Referenz erstellt
                    SCHREIBE ESSENSOBJEKT in ESSENSPLANUNGSOBJEKT;
                    ERMITTLE nötige FORMULARDATEN des ESSENSPLANUNGSOBJEKTS;
                    SENDE FORMULARLISTE an UI;

    UI: ERSTELLE FORMULARDIALOG;
    UI: ZEIGE FORMULARDIALOG;

    WARTEN auf NUTZEREINGABE;
    EVALUIERE NUTZEREINGABE;
    WENN NUTZEREINGABE = MENGE;
        DANN
            EMPFANGE MENGE von UI;
            SCHREIBE MENGE in ESSENSPLANUNGSOBJEKT;
            SONST WENN NUTZEREINGABE = MENGENTYP;
                DANN
                    EMPFANGE MENGENTYP von UI;
                    SCHREIBE MENGENTYP in ESSENSPLANUNGSOBJEKT;
            SONST WENN NUTZEREINGABE = STÜCKANZAHL;
                DANN
                    EMPFANGE STÜCKANZAHL von UI;
                    SCHREIBE STÜCKANZAHL in ESSENSPLANUNGSOBJEKT;
            SONST WENN NUTZEREINGABE = KOMMENTAR;
                DANN
                    EMPFANGE KOMMENTAR von UI;
                    SCHREIBE KOMMENTAR in ESSENSPLANUNGSOBJEKT;
            SONST WENN NUTZEREINGABE = GETRÄNK hinzufügen;
                DANN
                    SUCHE GETRÄNK auf DATENBANK;
                    EMPFANGE GETRÄNKELISTE von DATENBANKKONNEKTOR;
                    SENDE GETRÄNKELISTE an UI;
                    UI: ZEIGE GETRÄNKELISTE;
                    WARTEN auf NUTZEREINGABE;
                    WENN NUTZEREINGABE = GETRÄNK auswählen;
                        DANN
                            EMPFANGE ausgewähltes GETRÄNK;
                            SONST WENN NUTZEREINGABE = GETRÄNK anlegen;
                                DANN

```

```

        ERSTELLE GETRÄNKEOBJEKT der Klasse GETRÄNK;
        ERSTELLE GETRÄNKEPLANUNGSOBJEKT der Klasse GETRÄNKEPLANUNG;
        SCHREIBE neue GETRÄNKEPLANUNG in VERPFLEGUNGSOBJEKT;
//Hier wird eine Referenz erstellt
        SCHREIBE GETRÄNKEOBJEKT in GETRÄNKEPLANUNGSOBJEKT;
        ERMITTLE nötige FORMULARDATEN des GETRÄNKEPLANUNGSOBJEKTS;
        SENDE FORMULARLISTE an UI;

        UI: ERSTELLE FORMULARDIALOG;
        UI: ZEIGE FORMULARDIALOG;

        WARTEN auf NUTZEREINGABE;
        EVALUIERE NUTZEREINGABE;
        WENN NUTZEREINGABE = MENGE;
            DANN
                EMPFANGE MENGE von UI;
                SCHREIBE MENGE in GETRÄNKEPLANUNGSOBJEKTS;
            SONST WENN NUTZEREINGABE = MENGENTYP;
                DANN
                    EMPFANGE MENGENTYP von UI;
                    SCHREIBE MENGENTYP in GETRÄNKEPLANUNGSOBJEKTS;
            SONST WENN NUTZEREINGABE = STÜCKANZAHL;
                DANN
                    EMPFANGE STÜCKANZAHL von UI;
                    SCHREIBE STÜCKANZAHL in GETRÄNKEPLANUNGSOBJEKTS;
            SONST WENN NUTZEREINGABE = KOMMENTAR;
                DANN
                    EMPFANGE KOMMENTAR von UI;
                    SCHREIBE KOMMENTAR in GETRÄNKEPLANUNGSOBJEKTS;
        SONST WENN NUTZEREINGABE = KOSTEN hinzufügen;
            DANN
                ERSTELLE neues KOSTENOBJEKT der Klasse KOSTEN;
//Das Anlegen eines Kostenobjekts wird hier nicht näher
//betrachtet.
                FÜGE KOSTENOBJEKT zu VERPFLEGUNGSOBJEKT hinzu.
            SONST WENN NUTZEREINGABE = SPEICHERN;
                DANN
                    ERMITTLE vollständiges VERPFLEGUNGSOBJEKT;
//Hier findet ebenfalls die Prüfung auf Vollständigkeit statt
                    ERMITTLE die ESSENSPLANUNG;
                    ERMITTLE die GETRÄNKEPLANUNG;
                    SENDE zu speichernde Daten an DATENBANKKONNEKTOR um Daten auf
DATENBANK zu speichern;
// Exemplarischer Pseudocode siehe DATENBANKKONNEKTOR

        ZEIGE eine WARTEANZEIGE
//Bsp: Sanduhr
        WARTEN auf STATUSMELDUNG des DATENBANKKONNEKTOR als Antwort;
        EMPFANGE STATUSMELDUNG;
        ENTFERNE die WARTEANZEIGE;
        WENN STATUSMELDUNG = 200
//Status 200 = OK
            DANN
                ZEIGE DIALOG für erfolgreiches speichern;
            SONST
                ZEIGE DIALOG für Fehler;
        WARTEN bis NUTZER DIALOG bestätigt;
        ERMITTLE VERPFLEGUNGSOBJEKT;
        SENDE VERPFLEGUNGSOBJEKT an GUI zur Anzeige der DETAILSEITE;
        UI: ZEIGE DETAILSEITE;
        SONST WENN NUTZEREINGABE = ABBRECHEN;
            DANN

```

```
UI: ZEIGE BESTÄTIGUNGSDIALOG;  
WARTE AUF NUTZEREINGABE;  
WENN NUTZEREINGABE = BESTÄTIGT;  
    DANN  
        ZERSTÖRE VERPFLEGUNGSOBJEKT;  
        ZERSTÖRE ESSENSPLANUNGSOBJEKT;  
        ZERSTÖRE GETRÄNKEPLANUNGSOBJEKT;  
        UI: ZEIGE ÜBERSICHTSSEITE;  
    SONST  
        UI: SCHLIESSE DIALOG;
```

```
-----  
-      PSEUDOCODE DATENBANKKONNENTOR      -  
-----
```

```
- SENDE SUCHANFRAGE an DATENBANKKONNEKTOR -  
-----
```

```
EMPFANGE SUCHANFRAGE;  
ÜBERPRÜFE SUCHANFRAGE;  
FÜHRE SUCHANFRAGE aus;  
WARTE auf ERGEBNIS;  
EVALUIERE STATUSMELDUNG der DATENBANK;  
GEBE ERGEBNIS zurück;
```

```
-SENDE Daten an DATENBANKKONNEKTOR um sie auf der DATENBANK zu speichern-  
-----
```

```
EMPFANGE DATEN;  
STELLE die zu SICHERNDEN DATEN fest;  
SOLANGE noch zu SICHERNDE OBJEKTE ÜBRIG;  
    EVALUIERE das OBJEKT;  
    WENN OBJEKT ist KORREKT;  
        SOLANGE DATEN im OBJEKT;  
            SCHICKE DATEN AN DATENBANK;  
    SONST  
        GEBE STATUSMELDUNG 500 zurück;  
GEBE STATUSMELDUNG 200 zurück;
```



## 5.4. Szenariobetrachtung: Essen anlegen

Wie oben bereits näher erläutert, existieren im Sequenzdiagramm „Catering anlegen“ Referenzblöcke, die wiederum auf weitere Unterszenarien verweisen. Da diese sich vom Ablauf her stark ähneln, wird hier exemplarisch das Szenario „Essen anlegen“ näher betrachtet. Der Anwendungsfall kann genau zwei unterschiedliche Vorbedingungen haben: Entweder der Nutzer hat gerade eine Aktivität angelegt und möchte hierbei eine Speise zu seiner Essensliste hinzufügen, findet jedoch keine passende Speise in der Vorschlagsliste oder der Nutzer möchte einfach so eine Speise im System anlegen, ohne dass er diese einer Aktivität, bzw. einer Essensplanung zuweisen möchte.

Um seine Speise anlegen zu können, werden ähnlich wie bei der Entität Catering zuvor auch, obligatorische und optionale Attributwerte benötigt. Bei der Entität „Essen“ handelt es sich um den obligatorischen Titel der Speise, sowie die optionale Angabe von Allergenen Informationen.

## 5.5. Diagrammbetrachtung: Essen anlegen

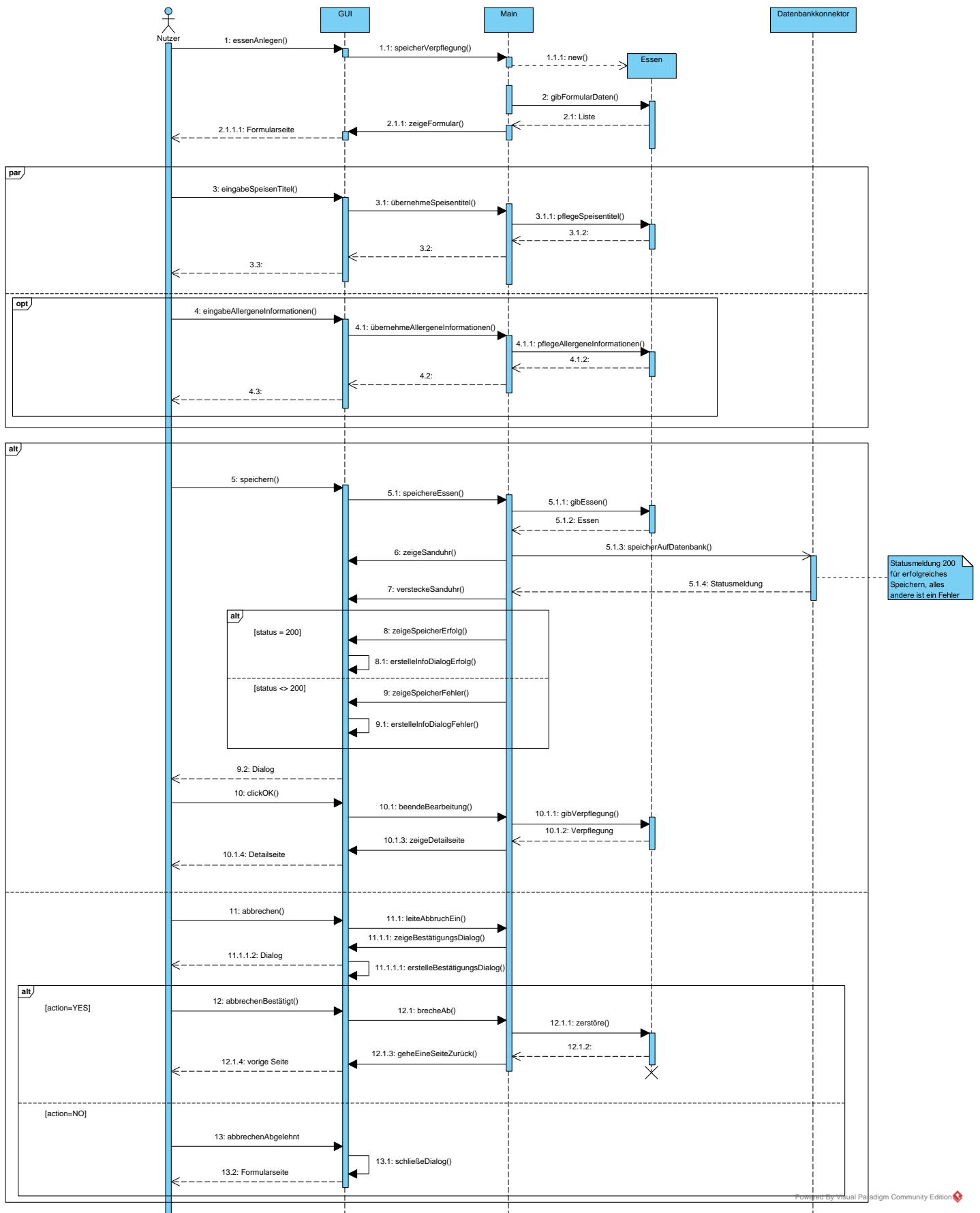
Das oben beschriebene Unterszenario wird durch das nachfolgende zusätzliche Sequenzdiagramm nach UML Standard visualisiert. Wie in dem ersten Diagramm wird hier auf die explizite Fehlerbehandlung aus Gründen der Übersichtlichkeit verzichtet. Dieses Diagramm dient lediglich der Visualisierung der Referenzen aus dem ersten Diagramm. Hierdurch wird ein Einblick in den Ablauf beim expliziten Anlegen einer Entität, wie beispielsweise einer Speise, gewährt. Im Fokus stehen hierbei der Nachrichtenaustausch, sowie das Erstellen, Löschen und Speichern einzelner Elemente.

Generell besitzt das Sequenzdiagramm insgesamt fünf unterschiedliche Entitätstypen, die miteinander interagieren. Die Typdefinitionen der einzelnen Entitäten sind dem Analyseklassendiagramm zu entnehmen und werden bewusst abstrakt gehalten, um beim späteren Entwurf an keine Restriktionen durch das Sequenzdiagramm gebunden zu sein.

Durch die initiale Aktion „Essen Anlegen“ wird das Szenario durch den Nutzer eingeleitet. Hierbei wird durch das System eine neue Instanz des Objekts Essen erstellt. Nach einer Anforderung der benötigten Attribute für die Entität Essen wird dem Nutzer eine Formularseite, basierend auf der Art des Objekts angezeigt. Nachdem der Nutzer im bearbeitenden Modus ist, hat er die Möglichkeit die entsprechenden Felder der Objekte zu pflegen. Im Beispiel der Speise ist das das obligatorische Feld für den Titel, sowie das optionale Feld für weitere Allergene Informationen. Nach einer Werteingabe in das Feld, werden die dazugehörigen Attribute der Objektinstanz abgeändert.

Nach der Bearbeitung aller nötigen Felder hat der Nutzer, ähnlich wie im vorigen Sequenzdiagramm, die Möglichkeit seine Aktion abubrechen oder zu speichern. Bei der Speicherung wird derselbe Prozessablauf, wie der oben beschriebene genutzt. Lediglich die Datenabfrage der korrespondierenden Entität ändert sich. Weiterhin hat der Nutzer die Möglichkeit das Anlegen der Speise abubrechen und zu seiner vorigen Seite zurückzukehren.





## 6. Aktivitätsdiagramm

Das nachfolgende Aktivitätsdiagramm visualisiert das Szenario „Unterhaltungsbeitrag anlegen“ und stellt sämtliche Relationen UML konform dar. Das Aktivitätsdiagramm basiert auf der Analyse aller zur Verfügung stehenden Datenklassen und bildet Entitätsnamen dementsprechend verallgemeinert ab. Spätere Entwurfsentscheidungen im Entwurfsklassendiagramm, insbesondere Namensgebungen, können vom analysebasierten Aktivitätsdiagramm abweichen. Im Fokus stehen das allgemeine Verhalten zwischen einzelnen Objekten, sowie deren Kontrollflüsse und das generelle Systemverhalten.

### 6.1. Szenariobetrachtung: Unterhaltungsbeitrag anlegen

Innerhalb eines Hochzeitsprojekts können Hochzeitsmanager, sowie Unterhaltungsmanager Beiträge anlegen, die zur Unterhaltung der Gäste während der Hochzeit beitragen. Diese können explizit als besondere Aktivitätsart angelegt werden. Als Hochzeitsmanager bekommt man beim Anlegen einer Aktivität so die Möglichkeit zwischen einer normalen Aktivität und einem Unterhaltungsbeitrag zu wählen. Der Unterhaltungsmanager hat hingegen lediglich die Möglichkeit den Unterhaltungsbeitrag anzulegen. Weiterhin wird nach dem Erstellen des Beitrags eine leere Formularseite angezeigt, in die der Nutzer sämtliche, für die Aktivität wichtige Daten eintragen kann. Unter anderem fallen hierbei obligatorische Daten wie zum Beispiel Titel, Start- und Endzeit, sowie Start- und Enddatum an, aber auch optionale Daten wie beispielsweise ein Kommentar, Lokalität, Hilfsmittelplanungen, Medien, Teilnehmer oder andere Daten, die aus dem Analyseklassendiagramm entnommen werden können.

Daten wie Teilnehmer oder Hilfsmittelplanungen sind wiederum Listenelemente, die eine Auswahlmöglichkeit innerhalb von Listen bieten oder ebenfalls die Möglichkeit weitere Unterobjekte selbst zu erstellen. Weiterhin kann der aktuelle Zustand der Aktivität gesetzt werden. Hierbei hat der Nutzer die Wahl zwischen den definierten Zuständen: geplant, in Arbeit, abgeschlossen und abgebrochen.

Nach der erfolgreichen Eingabe von mindestens allen obligatorischen Daten hat der Nutzer die Möglichkeit sämtliche Änderungen konsistent zu speichern. Zusätzlich kann der Nutzer das Anlegen des Unterhaltungsbeitrags jederzeit abbrechen. Dadurch wird der bestehende Unterhaltungsbeitrag gelöscht und der Nutzer kehrt zur vorigen Übersichtsseite zurück.

## 6.2. Diagrammbetrachtung: Unterhaltungsbeitrag anlegen

Das Aktivitätsdiagramm des oben beschriebenen Szenarios umfasst das Anlegen, sowie die damit verknüpfte Bearbeitung und Pflege einer Aktivität, die wahlweise ebenfalls als Unterhaltungsbeitrag angelegt werden kann. Aktionen, die näher aufgesplittet wurden, wurden dementsprechend grau markiert und verfeinert. Eine Ausnahme bilden die Aktivitäten Zustand ändern, Medium auswählen, Hilfsmittelplanung auswählen, Teilnehmer auswählen und Lokalität auswählen. Diese wurden zu Gunsten der Übersichtlichkeit nicht näher verfeinert, da diese Aktivitätsverläufe ähnlich wie die, der Aktivität „Verantwortliche Person auswählen“ sind. Hierbei müssen nur die Namen der einzelnen Entitäten geändert werden. Insgesamt wurden vier Unteraktivitäten näher verfeinert.

Der Nutzer startet den Aktionsverlauf, indem er eine neue Aktivität erstellen möchte. Hierbei wird anfangs eine initiale Aktivität erzeugt, die mit den wichtigsten Werten, wie beispielsweise eine eindeutige ID, der verantwortlichen Person, sowie einem Status befüllt wird. Weiterhin wird zu Beginn die Nutzerrolle geprüft. Ist der Nutzer in der Rolle eines Unterhaltungsmanagers hat er gar keine Wahl und es wird direkt ein Unterhaltungsbeitrag generiert, indem das entsprechende boolesche Attribut gesetzt wird. Der Hochzeitsmanager hingegen bekommt einen Dialog zur Auswahl angezeigt. Hierbei kann er sich entscheiden ob er einen Unterhaltungsbeitrag oder eine normale Aktivität erstellen möchte. Nach der Wahl eines Unterhaltungsbeitrags ist dieser fertig initialisiert.

Anschließend wird dem Nutzer eine Formularseite angezeigt, die den entsprechenden Beitrag als Eingabe benötigt. Nun können sämtliche obligatorischen und optionalen Daten durch den Nutzer gepflegt werden. Die Reihenfolge zur Eingabe spielt hierbei keine Rolle. Die Eingaben einzelner Felder werden auf ihren Wert geprüft und bei Erfolg gesichert. Der Zustand muss nicht extra geprüft werden, da hier nur eine Auswahl aus einer definierten Statusliste möglich ist. Komplexere Eingaben, wie beispielsweise Medium, Lokalität, Hilfsmittelplanung, Teilnehmer oder weitere verantwortliche Personen werden durch Unteraktivitäten näher behandelt. Hierbei wird zuerst eine Liste der dazugehörigen Entität angezeigt, die sämtliche Einträge enthält, die bereits in der Datenbank sind. Falls es keine Einträge gibt oder der Nutzer keinen dieser Einträge bevorzugt, hat er die Möglichkeit diesen neu anzulegen. Die resultierende Entität, wie beispielsweise eine Person, wird der entsprechenden Liste innerhalb der Aktivität hinzugefügt. Sämtliche getätigten Änderungen werden im Beitrag erfasst und gesichert. Somit entsteht ein Objekt, welches einen geänderten Beitrag verkörpert.

Erst nach der expliziten Sicherung durch den Nutzer werden sämtliche Pflichteingaben auf Vollständigkeit geprüft. Falls bei der vorigen Eingabe ein Fehler oder nun eine Unvollständigkeit auftaucht, so wird die entsprechende Fehleingabe als Objekt übermittelt und dem Nutzer ein

Fehlerdialog angezeigt. Nach schließen des Dialogs löst das akzeptierende Event „Nutzer schließt Dialog“ die Aktion „Formularseite anzeigen“ aus, sodass der Nutzer wieder die Formularseite angezeigt bekommt. Hierbei wird zuvor jedoch das entsprechende Feld auf der Formularseite als fehlerhaft markiert. Nun hat der Nutzer nochmals die Möglichkeit seine Eingaben zu korrigieren. Falls jedoch kein Fehler auftritt, so wird der Nutzer gefragt, ob er den Beitrag, bevor er ihn publiziert, gegenüber der Hochzeitsmanager verstecken möchte. Stimmt er dem zu, so wird das entsprechende boolesche Attribut innerhalb der Aktivität gesetzt. Anschließend wird der vollständige Beitrag auf der Datenbank gespeichert. Dieses Szenario wird hier nicht näher erläutert. Es resultiert eine Statusmeldung, die entweder zum Erfolg oder zum Fehler führen kann. Bei Erfolg wird die Aktivität erfolgreich verlassen und auf die Detailseite des Beitrags zurückgekehrt.

Während des gesamten Prozess hat der Nutzer die Möglichkeit das Anlegen des Beitrags abubrechen. Hierbei wird ein erneuter Bestätigungsdialog angezeigt, um den Nutzer auf die Folgen eines Abbruchs hinzuweisen. Hierbei gehen sämtliche nicht getätigten Speicherungen verloren. Der aktuell erstellte, nicht gespeicherte Betrag wird gelöscht. Weiterhin kehrt der Nutzer zur Übersichtsseite zurück. Verneint er den Bestätigungsdialog, so werden keine Änderungen vorgenommen und der Nutzer kehrt zur Formularseite zurück und kann weitere Änderungen tätigen.

## 6.3. Pseudocode: Unterhaltungsbeitrag anlegen

```
-----
- PSEUDOCODE AKTIVITÄTSDIAGRAMM -
- UNTERHALTUNGSBEITRAG ANLEGEN -
-----
```

EVENT "ClickErstelleBeitrag" wurde ausgelöst:

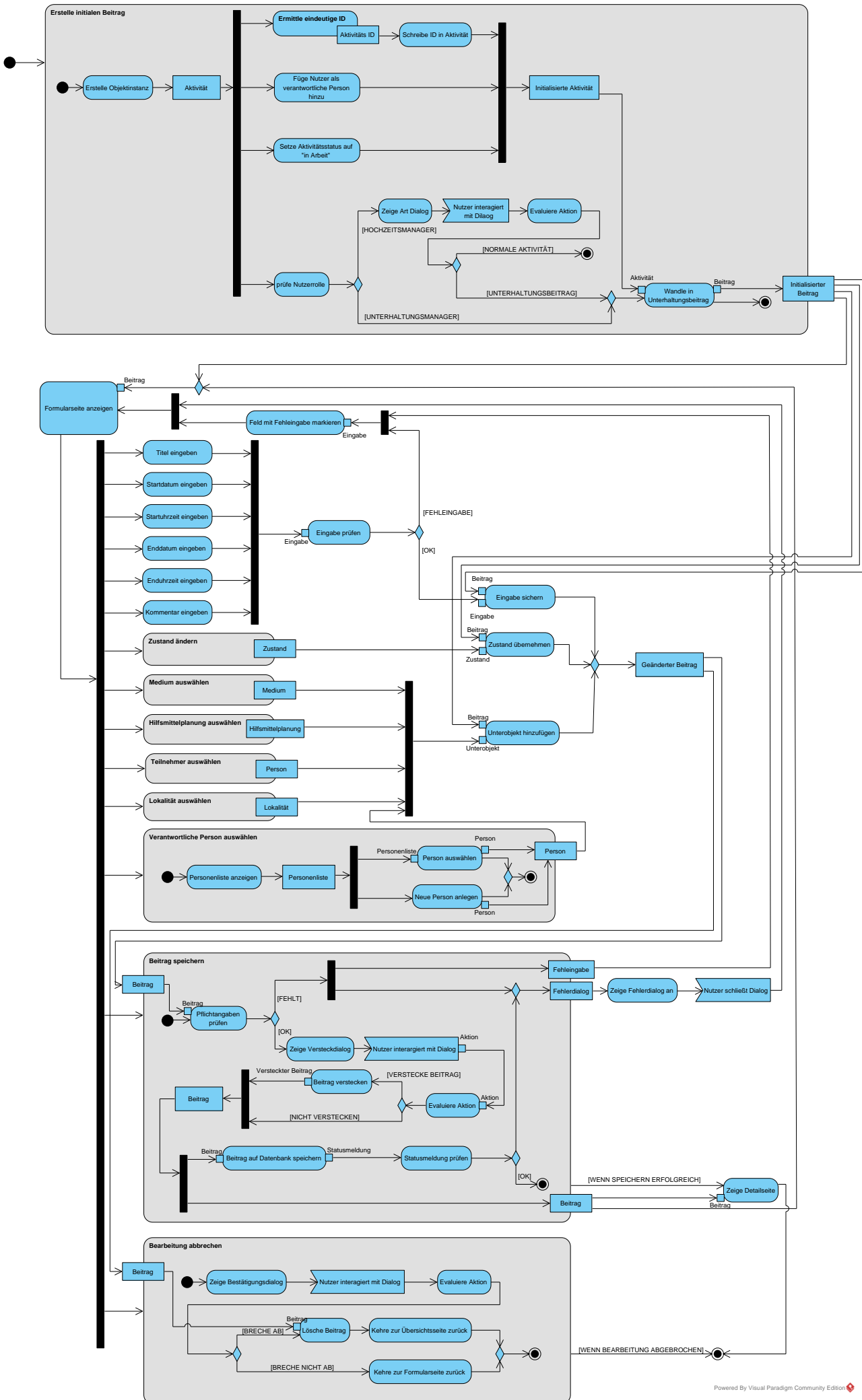
```
ERSTELLE OBJEKT einer Aktivität;
ERMITTLE eindeutige ID;
//Zugriff auf Datenbank
SCHREIBE ID in das OBJEKT der Aktivität;
ERMITTLE NUTZERPERSON;
//Person, die mit dem aktuellen Nutzer verknüpft ist
FÜGE PERSON in LISTE der verantwortlichen Personen hinzu;
SETZE STATUS des OBJEKTS auf IN_ARBEIT;
ERMITTLE NUTZERROLLE;
WENN NUTZERROLLE = HOCHZEITSMANAGER;
    DANN
        ZEIGE DIALOG um Aktivitätsart zu bestimmen;
//Normale Aktivität oder Unterhaltungsbeitrag?
        WARTE auf NUTZEREINGABE;
        EVALUIERE NUTZERAKTION;
        WENN NUTZERAKTION = UNTERHALTUNGSBEITRAG;
            WANDLE aktivitäts-OBJEKT in unterhaltungsbeitrags-OBJEKT;
//Entsprechendes Flag wird gesetzt
        SONST
            WANDLE aktivitäts-OBJEKT in unterhaltungsbeitrags-OBJEKT;

ZEIGE FORMULARSEITE für OBJEKT an;
//Wichtige Sprungadresse für erneute Eingaben
WARTE AUF NUTZEREINGABE;
EVALUIERE NUTZEREINGABE;
WENN NUTZEREINGABE = TITEL oder STARTDATUM oder STARTUHRZEIT oder ENDDATUM
oder ENDUHRZEIT oder KOMMENTAR;
    DANN
        PRÜFE NUTZEREINGABE;
        WENN PRÜFUNG = FEHLEINGABE;
            DANN
                MARKIERE fehlerhaftes EINGABEFELD;
                ZEIGE wieder FORMULARSEITE;
//Rücksprung auf erneute Nutzereingabe
            SONST
                SICHERE EINGABE in OBJEKT;
                ZEIGE wieder FORMULARSEITE;
        SONST WENN NUTZEREINGABE = ZUSTAND;
            DANN
                SICHERE ZUSTAND in OBJEKT;
                ZEIGE wieder FORMULARSEITE;
        SONST WENN NUTZEREINGABE = MEDIUM oder HILFSMITTELPLANUNG oder
TEILNEHMER oder LOKALITÄT oder VERANTWORTLICHE PERSON
            WENN NUTZEREINGABE = MEDIUM auswählen;
//Exemplarisch wird nur die Auswahl von
//verantwortlichen Personen näher behandelt
//da die Aktivitätsabläufe sich stark ähneln
            DANN
                WÄHLE MEDIUM als UNTEROBJEKT aus;
            SONST WENN NUTZEREINGABE = HILFSMITTELPLANUNG auswählen;
            DANN
                WÄHLE HILFSMITTELPLANUNG als UNTEROBJEKT aus;
            SONST WENN NUTZEREINGABE = TEILNEHMER auswählen;
```

```

DANN
    WÄHLE TEILNEHMER als UNTEROBJEKT aus;
SONST WENN NUTZEREINGABE = LOKALITÄT auswählen;
DANN
    WÄHLE LOKALITÄT als UNTEROBJEKT aus;
SONST WENN NUTZEREINGABE = VERANTWORTLICHE PERSON auswählen;
//Hier der nähere Ablauf einer Auswahl
DANN
    ZEIGE LISTE der verantwortlichen PERSONEN an;
    WARTEN auf NUTZEREINGABE;
    WENN NUTZEREINGABE = PERSON AUSWÄHLEN;
        DANN
            WÄHLE PERSON aus PERSONENLISTE aus;
            NEHME PERSON als UNTEROBJEKT;
            SONST WENN NUTZEREINGABE = PERSON ANLEGEN;
                LEGE neue PERSON an;
//Neuer Unterprozess bei dem eine weitere Person
//neu angelegt wird. Hierbei öffnet sich eine
//neue Formularseite.
            FÜGE PERSON zur PERSONENLISTE hinzu;
            NEHME PERSON als UNTEROBJEKT;
        SICHERE UNTEROBJEKT in OBJEKT;
        ZEIGE wieder FORMULARSEITE;
    SONST WENN NUTZEREINGABE = SPEICHERN;
        DANN
            PRÜFE PFLICHTANGABEN von OBJEKT;
            WENN nicht alle PFLICHTANGABEN ausgefüllt;
                DANN
                    ERSTELLE FEHLERDIALOG anhand fehlender EINGABE;
                    ZEIGE FEHLERDIALOG;
                    WARTEN bis NUTZER den DIALOG schließt;
                    MARKIERE fehlerhaftes EINGABEFELD;
                    ZEIGE wieder FORMULARSEITE;
                SONST
                    ZEIGE VERSTECKDIALOG;
//Dialog, welcher abfragt, ob der Nutzer den
//Beitrag verstecken will
                    WARTEN AUF NUTZEREINGABE;
                    WENN NUTZEREINGABE = JA;
                        DANN
                            VERSTECKE OBJEKT;
//Entsprechendes Flag wird gesetzt
                    SPEICHERE OBJEKT auf DATENBANK;
                    EMPFANGE STATUSMELDUNG der DATENBANK als antwort;
                    WENN STATUSMELDUNG = FEHLER;
                        DANN
                            ZEIGE FEHLERDIALOG;
                            WARTEN bis NUTZER den DIALOG schließt;
                            ZEIGE wieder FORMULARSEITE;
                        SONST
                            ZEIGE DETAILSEITE des OBJEKTS;
    SONST WENN NUTZEREINGABE = ABBRECHEN;
        DANN
            ZEIGE BESTÄTIGUNGSDIALOG;
            WARTEN auf NUTZEREINGABE;
            WENN NUTZEREINGABE = JA, breche ab;
                DANN
                    LÖSCHE OBJEKT;
                    ZEIGE vorige ÜBERSICHTSSEITE;
                SONST
                    ZEIGE wieder FORMULARSEITE;

```



## 7. Entwurfsklassendiagramm

Im Folgenden wird das Entwurfsklassendiagramm erläutert. Es beschreibt das Datenmodell des Hochzeitsplaners und ist wesentlicher Bestandteil der Konzeption. Das Entwurfsklassendiagramm basiert im Wesentlichen auf dem Analyseklassendiagramm und baut darauf auf. Neben den im Analyseklassendiagramm eingeführten Klassen, werden noch weitere Klassen als Hilfe für die Umsetzung benötigt. Diese Klassen wurden im Entwurfsklassendiagramm identifiziert und dargestellt. Die Beschreibung des Entwurfsklassendiagramms erfolgt Paketweise. Es wird nicht auf die Methoden und Attribute der einzelnen Klassen eingegangen.

### 7.1. Pakete

In diesem Abschnitt wird auf die einzelnen Pakete des Entwurfsklassendiagramms eingegangen.

#### 7.1.1. Paket Modellklassen

Das Paket Modellklassen ist das Hauptpaket des Entwurfsklassendiagramms. Das Paket ist aus dem Analyseklassendiagramm entstanden. Dabei wurden weitere Klassen identifiziert und hinzugefügt. Das Hochzeitsprojekt ist direkt mit der Main Klasse verbunden. Durch diese Verbindung kann die Main auf alle Daten innerhalb eines Hochzeitprojektes zugreifen. Das Hochzeitsprojekt besitzt weiterhin dieselben Verbindungen wie vorher. Bei der weiteren Analyse des Paketes werden nur die Unterschiede hervorgehoben, Gleichheiten mit dem Analyseklassendiagramm werden nicht weiter betrachtet.

Im Analyseklassendiagramm gab es Aktivitäten und die kirchliche als auch standesamtliche Trauung. Die Attribute waren fast identisch und bei den Trauungen handelt es sich auch um Aktivitäten. Um dies sinnvoll nutzen zu können, wurde eine Oberklasse „Aktivität“ erstellt. Diese ist Abstract, damit kein Objekt von dieser angelegt werden kann.

Die kirchliche und standesamtliche Trauung besitzen dieselben Attribute und können dadurch ebenfalls zusammengefasst werden. Jedoch können sie nicht direkt von Aktivität erben, da in der Aktivität der Zustand nicht erfasst wird. Um nicht beiden Klassen einen Zustand zuweisen zu müssen, wurde eine weitere Oberklasse implementiert, die Trauung. Die Trauung ist ebenfalls Abstract, so wie die Aktivität. Die Trauung besitzt einen Trauungszustand und dieser wurde als Enum umgesetzt. Der Zustand wurde nicht bei der Superklasse Aktivität hinzugefügt, weil die Klasse Aktion andere Zustände besitzt als die Trauung. Die Klasse Aktion erbt ebenfalls von Aktivität und beinhaltet die Attribute geheim und Unterhaltsbeitrag. Für die Erklärung der Attribute siehe Analyseklassendiagramm. Die



Aktion besitzt einen Aktionszustand, welcher sich, wie oben bereits erwähnt, von den Zuständen der Trauung unterscheidet.

Bei der Lokalität wird der Buchungsstatus als Enum abgespeichert. Die Rolle des Benutzers wird ebenfalls so gesichert.

Die Werteliste der Klassifizierung der Telefonnummer und E-Mail Adresse wurde im Entwurfsklassendiagramm als Enum umgesetzt. Die Werteliste des Bezahlstatus der Kosten wurde ebenfalls als Enum implementiert.

Die Essens- und Getränkeplanung beinhalten dieselben Attribute und Methoden. Aufgrund dessen wurde die abstrakte Oberklasse Lebensmittelplanung eingeführt. Die beiden Klassen erben von der Oberklasse.

Die Attribute und Methoden des Essens und der Getränke besitzen eine große Schnittmenge. Daher wurden diese zusammengefasst und eine Oberklasse gebildet. Das Getränk beinhaltet noch ein weiteres Attribut enthältAlkohol.

### 7.1.2. GUI

Das Paket GUI bildet in dem Entwurfsklassendiagramm das zweitgrößte Paket. In diesem werden die Funktionalitäten bereitgestellt, um die grafische Oberfläche der Software zu entwerfen. Die Klasse GUIController übernimmt dabei die Hauptaufgabe. Diese besitzt ebenfalls eine Verbindung zu Main. Der GUIController besitzt eine Shell, in dem sich der ganze Inhalt der Anwendung befindet. Die Shell ist die erste sichtbare Komponente. Dieser erbt von der Java Klasse javax.swing.JFrame. Die Shell besitzt eine Icontabbar, um die Bedienung möglichst übersichtlich zu gestalten. Um sich dies besser vorstellen zu können, siehe GUI Entwurf. Die Icontabbar besitzt die Tabart als Enum. Die im Enum enthaltenen Werte werden in der Icontabbar angezeigt.

Um die intuitive Bedienung zu erleichtern, wird eine Statusbar implementiert. Die Klasse Dialog dient dazu, Dialoge innerhalb der Anwendung anzeigen zu können. Der Klasse Dialog erbt von der javax.swing.JDialog Klasse.

Die verschiedenen Seiten der GUI wurden als einzelne Klassen implementiert. Dabei spielen folgende fünf Klassen eine Rolle:

- **Aktivitätsübersicht:** Hier wird eine Übersicht der Aktivitäten bereitgestellt. Dafür wird die Planung in einem Kalender angezeigt und zum ausgewählten Tag wird eine Liste der Aktivitäten angezeigt.
- **Datenverwaltungsübersicht:** Hier können Daten zur Hochzeit gepflegt werden. Dabei können z.B. neue Getränke, Essen, Personen und Lokalitäten angelegt werden. Diese werden in einer Liste angezeigt. Innerhalb der Datenverwaltung kann in einem Suchfeld nach bereits vorhandenen Daten gesucht werden.
- **Benutzerübersicht:** Dient zur Benutzerverwaltung. Alle Benutzer werden in eine Liste angezeigt.
- **Hochzeitsübersicht:** Alle Stammdaten der Hochzeit werden in dieser Übersicht angezeigt. Diese Übersicht ist als Liste dargestellt.
- **Login:** Der Loginscreen dient zur Anmeldung ins System. Die Daten werden in einer Liste angezeigt.

Die Liste wird mithilfe der Klasse `javax.swing.JList` umgesetzt. Um Listeneinträge anzeigen zu können, benötigt die Liste Listenelemente. Außerdem besitzt die Liste noch eine Toolbar, in der eine Suche ermöglicht wird.

Der Kalender erbt von der Java `javax.swing.JCalendar` Klasse und wird durch weitere Methoden ergänzt.

Die Klassen Kalender, Suchfeld, Toolbar, Listenelement und Bildkarussell implementieren den `EventProvider`. Dieser ermöglicht es, Events zu senden und zu empfangen. Der `EventProvider` implementiert die Klassen im `UIEventInterfaces` Paket.

Eine weitere wichtige Seite der Shell ist die Detail Seite. Diese erbt von `javax.swing.JComponent`. Die Aktivitäts-, Personen-, Lokalitäts-, Verpflegungs- und Hilfsmitteldetails erben von der Details Klasse. Die Aktivitätsdetails Klasse ist mit dem Bildkarussell verbunden. Damit ist es möglich, die Medien innerhalb einer Aktivität durchzuschauen.

### 7.1.3. UIEventInterfaces

Dieses Paket dient zur Bereitstellung von Events. Ohne diese wäre eine Kommunikation zwischen den einzelnen Komponenten nicht möglich. Alle Komponenten, die untereinander kommunizieren wollen, müssen diese implementieren.

#### 7.1.4. ExceptionHandling

Das Paket ExceptionHandling kümmert sich um die Exceptions. Die Main implementiert die Klassen AuthorizationException und ConnectionException. Die Exceptions werden geworfen, sobald ein User versucht eine Aktion auszuführen, die nicht in seinem Aufgabenbereich liegt oder keine Verbindung zum Server aufgebaut werden kann. Diese erben wiederum von der java.lang.Exception Klasse.

#### 7.1.5. Util

Das Paket Util ist ein Hilfspaket für die Main. Die Main Klasse verwendet das Util Paket zum Erledigen von verschiedenen Aufgaben. Der E-MailSender ist für das Senden der E-Mails zuständig. Dieser erbt vom E-MailHelper, der die Grundfunktionalitäten zum Senden von E-Mails bereitstellt. Auch implementiert er den PDFExporter, damit Texte als PDF Anhänge mitgeschickt werden können. Der LanguageHelper hilft der Main Klasse die verschiedenen Sprachen bereitzustellen. Er implementiert verschiedene Sprachen und kann dann die von der Main angefragt Sprache zurückgeben. Ist die Sprache nicht definiert, wird Englisch als Default Sprache zurückgegeben. Die Klasse Synchronisation ist für die Synchronisation zwischen dem Server und den lokalen Änderungen verantwortlich. Sobald eine Internetverbindung besteht, werden die lokal gespeicherten Daten, die sie nicht auf dem Server befinden, auf den Server geschrieben.

Die Klasse KalenderEvent ermöglicht es Kalendereinträge als ICS zu exportieren. Dafür wird die Klasse KalenderICSExporter importiert.

Um Medien hochladen zu können, wird die Klasse Medienupload benötigt. Durch diese ist es möglich, Dateien auszuwählen und durch eine Verbindung zum Server, auf den Server zu laden. Um Dateien auf dem Rechner des Benutzers auszuwählen, wird die Klasse javax.swing.JFileChooser implementiert. Der ArchiviererController ist für das Archivieren der Daten zuständig. Es können auch Daten dearchiviert werden. Die Daten werden archiviert, sobald die Zeit für die Archivierung erreicht ist. Die Klasse EventZeitFuerArchivierung gibt an, wann die Archivierung gestartet werden soll. Um Daten zu archivieren wird das Interface Archivierer benötigt. Der Dearchivierer ist für das dearchivieren der Daten zuständig. Diese Funktion wird nur ausgeführt, wenn der Benutzer dies explizit angibt.

### 7.1.6. Kern

Die in diesem Paket enthaltenen Klassen beinhalten die Funktionalitäten, die den zentralen Ablauf der Anwendung gewährleisten. Hierzu gehört unter anderem die Bereitstellung der Benutzeroberfläche. Diese Funktionalität ist in dem Paket GUI zusammengefasst. Dabei wird für jeden Client der Applikation eine separate Oberfläche zur Verfügung gestellt. Der Client erhält dabei nur die grafische Darstellung der Oberfläche und dessen Eingaben werden zurück an die Applikation geschickt, die die Eingaben für die entsprechende Oberfläche weiterverarbeitet und den Ergebnisbildschirm zurück an den Client schickt.

Die Systemverwaltung, die Datenkonvertierung, die Datenbankverbindung, das Exceptions Handling und alle weiteren Aufgaben werden von der Main übernommen.

Über die Login Klasse wird der Login gesteuert und der Logout über die Logout Klasse. Die Beiden Klassen erben von der Autorisierung, die sich Paket Systemverwaltung befindet.

### 7.1.7. Systemverwaltung

Für die Benutzerverwaltung als auch für das Backup ist das Paket Systemverwaltung zuständig. Die Klasse Autorisierung überprüft beim Login und Logout, ob der Benutzer bestimmte Rechte hat. Diese hat eine Verbindung zur Benutzerverwaltung, in der die angelegten Benutzer gespeichert und verwaltet werden können. Für das Backup ist die Klasse Backup zuständig. Dafür besitzt sie eine Verbindung zur Klasse Datenbankanbieter. Durch diese wird ein Zugriff auf die aktuellen Daten auf der Datenbank ermöglicht. Um das Backup auf ein externes Medium abspeichern zu können, erbt die Klasse Backup von der Java Klasse jUSB. Damit kann das fertige Backup durch USB auf ein anderes Medium übertragen werden. Auch ist es möglich, das Backup in seiner Cloud zu sichern.

### 7.1.8. Datenkonvertierung

Das Paket Datenkonvertierung ist für die Konvertierung und den Im- und Export der Daten zuständig. Um die Daten importieren bzw. exportieren zu können, müssen sie zuerst in ein einheitliches Format gebracht werden. In unserem Fall ist das Format CSV. Das Interface FileConverter stellt Methoden bereit, um aus einem File die Einträge zu konvertieren und andersrum. Die Klassen Import und Export können von jeder Klasse aufgerufen werden, um den Import bzw. Export der Daten zu starten. Diese Klassen erben jeweils von dem FileConverter und zusätzlich noch von dem Interface Importierbar oder Exportierbar. Diese stellen Methoden zur Verfügung, um den Import bzw. Export durchführen zu können. Jede Modellklasse implementiert diese Interfaces und kann damit den Import bzw. Export starten.

### 7.1.9. Datenbankverbindung

Um auf die Daten der Datenbank zugreifen zu können, wird eine Datenbankverbindung benötigt. Dies wird durch das Paket Datenbankverbindung erledigt. Der Datenbankanbieter stellt dabei die Hauptklasse dar und steuert den Ablauf der Anfragen. Dieser ist mit der Klasse Entity verbunden. Jeder Eintrag im Hochzeitsplaner ist ein Entity. Um ein Entity eindeutig bestimmen zu können, benötigt dieser eine ID. Auch wird über den Datenbankanbieter das Backup gesteuert und dieser besitzt eine Verbindung zur Main.

Der Datenbankanbieter besitzt einen Datenpuffer, um oft geladene Daten schneller bereitstellen zu können. Ebenfalls besitzt der Datenbankanbieter eine Liste, mit deren Hilfe er die angefragten Daten zurückgeben kann. Die Liste erbt von der Java Klasse java.util.HashMap. Mithilfe dieser Java Klasse, kann ein Objekt durch seine ID eindeutig identifiziert werden.

Um eine Suche innerhalb des Hochzeitsprojekts durchführen zu können, wird die Klasse Suche benötigt. Diese greift auf den Datenbankanbieter zurück und lässt diesen die Suchanfrage ausführen.

## 7.2. Verwendete Entwurfsmuster

Entwurfsmuster sind Lösungsvorlagen für häufig auftretende Probleme beim Softwareentwurf. Sie werden häufig in der Softwarearchitektur verwendet. Ein Entwurfsmuster löst ein oder mehrere Probleme und bietet ein erprobtes Konzept. Entwurfsmuster lassen sich meist in eine der drei Kategorien Erzeugungsmuster, Strukturmuster oder Verhaltensmuster einordnen. Für den Hochzeitsplaner finden einige solcher Muster Anwendung, welche nachfolgend aufgezeigt werden.

### 7.2.1. Singleton

Das Entwurfsmuster Singleton gehört zu den sogenannten Erzeugungsmustern, welche der Erzeugung von Objekten dienen. Das Singleton sorgt dafür, dass von einer Klasse maximal ein Objekt erzeugt werden kann. Meist findet eine statische Implementierung des Musters statt, wodurch das einzige Objekt der Klasse in diesem Fall global verfügbar ist.

Die Implementierung eines Singletons wird über ein statisches Klassenattribut, welches das einzige Objekt der Klasse hält, realisiert. Dies ist nur für die Klasse selbst sichtbar. Der Zugriff auf die Instanz erfolgt über eine Klassenmethode (z.B. „getInstance“), welche das hinterlegte Objekt liefert. Es ist denkbar, erst ein Objekt erzeugen, wenn diese Methode aufgerufen wird.

Klassen in dem Hochzeitsmanager, die dieses Entwurfsmuster umsetzen, sind mit dem Stereotyp „<<Singleton>>“ gekennzeichnet. Die Main Klasse ist die einzige Klasse im Entwurfsklassendiagramm, die dieses Entwurfsmuster verwendet.

### 7.2.2. Lazy Loading

Lazy loading ist ein Entwurfsmuster, welches bewirkt, dass die Erzeugung eines Objektes erst dann stattfindet, wenn dieses benötigt wird. Vorteile bietet dieses Muster besonders bei einer großen Anzahl von Objekten, welche jeweils einige Zeit benötigen, um geladen zu werden. Indem man verhindert, dass nicht benötigte Objekte bereits geladen werden, kann eine Verbesserung der Programmgeschwindigkeit erreicht werden.

In dem Hochzeitsmanager wird dieses Entwurfsmuster für die Klasse „Entity“ verwendet. Dies macht sich besonders bei der Verwendung der Listenklassen bemerkbar, weil diese eine große Anzahl von Objekten der Klassen „Entity“ enthalten können. Somit steht eine Listenklasse zeitnah zur Verfügung und kann verwendet werden.

### 7.2.3. Fassade

Die Fassade ist ein Entwurfsmuster, welche eine einheitliche Schnittstelle zu einem Subsystem bietet. Auf die Klassen des Subsystems kann trotzdem noch zugegriffen werden. Dies bietet sich vor allem an, wenn ein System viele technische orientierte Klassen enthält, die selten von außen verwendet werden. Der Hochzeitsmanager implementiert dies gleich an zwei Stellen. Zu einem wird dies bei dem Paket GUI verwendet und zum anderen bei dem Paket Modelklassen. Der GUIController ist bei der GUI verantwortlich für die technisch orientierten Klassen und bei den Modellklassen bildet das Hochzeitsprojekt die Fassade.



## 8. GUI-Entwurf

Die nachfolgenden Screenshots visualisieren Teilausschnitte des Softwareprodukts. Hierbei wurden insgesamt drei unterschiedliche Gesamtansichten ausgewählt. Diese werden im Folgenden näher erläutert und mit den entsprechenden Entitäten in Verbindung gebracht. Hierbei stehen in erster Linie die Navigation, das Design und das Zusammenspiel einzelner Komponenten im Vordergrund. Die komplette grafische Benutzerschnittstelle sieht in allen Distributionsvarianten ähnlich aus, da die Entwicklung JAVA basiert ist. Die komplette Oberfläche ist mit einem eigenen Design ausgestattet worden. Sämtlichen UI-Elemente wurden dafür überarbeitet. In den Qualitätsanforderungen wurde weiterhin ein hoher Wert auf Gestaltung, Funktionalität und Benutzbarkeit gelegt. Dadurch wurden sämtliche Elemente mit besonderer Sorgfalt, unter Einhaltung moderner Designrichtlinien entworfen. Hierbei stehen die einfache Nutzbarkeit, die Übersichtlichkeit und der schnelle Zugriff auf alle wichtigen Funktionalitäten im Fokus.

Generell bietet eine Statusleiste einen einfachen Einstieg in das komplette Tool. Hierüber kann sich der Nutzer in das System einloggen, ausloggen oder sein Nutzerprofil verwalten, sofern er denn eingeloggt ist. Weiterhin wird darauf geachtet, dass die gesamte GUI die einzelnen Elemente dynamisch, anhand der Größe des aktuellen Fensters ausrichtet. Daher gibt es ein zusätzliches Kontextmenü in der Statusleiste, welches für kleine Bildschirme optimiert wurde.

Unter der Statusleiste bietet die Anwendung die wichtigsten Zugriffsoptionen für die aktuellen Nutzer, die anhand der Nutzerrolle ermittelt werden. So kann der Unterhaltungsmanager auf Funktionen wie das Anlegen von Unterhaltungsbeiträgen oder das generelle Einsehen der Hochzeit zugreifen. Der Hochzeitsmanager kann diese jedoch zusätzlich bearbeiten, sowie weitere Benutzer des Systems und seine eigenen Daten verwalten.



## 8.1. Hochzeitsübersicht


Der erste Screenshot visualisiert die Übersichtsseite des kompletten Hochzeitsprojekts. Hier hat der Nutzer die Möglichkeit die wichtigsten Daten des Projekts einzusehen. An erster Stelle stehen hierbei das Hochzeitsdatum, sowie der trauende Standesbeamte. Dieser, sowie die Aktivitäten standesamtliche Trauung und kirchliche Trauung (sofern angelegt) können über einen Link näher eingesehen werden. Über den Stift-Button kann die ganze Seite in den Bearbeitungsmodus wechseln. Dadurch wird dem Nutzer die Möglichkeit geboten, die eingegebenen Daten nochmals zu ändern und zu speichern.

Unter den wichtigsten Stammdaten des Hochzeitsprojekts stehen die zwei Hauptrollen der Hochzeit im Fokus. Hierbei werden die persönlichen Kontaktdaten der Braut, sowie des Bräutigam übersichtlich dargestellt. Links ermöglichen den direkten Zugriff auf die entsprechende Detailseite der Person (bei Klick auf Namen), das direkte Senden einer E-Mail, sofern ein Mail Programm auf dem Rechner installiert ist (bei Klick auf die E-Mail), sowie der direkte Verbindungsaufbau für einen Anruf, sofern ein Programm zur Telefonie, wie beispielsweise Skype, auf dem Rechner installiert ist (bei Klick auf Telefonnummer).

Unter dem Hochzeitspaar findet der Nutzer eine zusätzliche Gesamtkostenübersicht für die komplette Hochzeit. Hierbei werden die Beträge eines Kostenobjekts anhand des dazugehörigen Status gruppiert und aufsummiert. Dadurch gibt es eine Unterscheidung in bereits bezahlte Kosten, sowie geplante Kosten. Die Gesamtkosten werden durch die Summe aller Kosten ermittelt.

Ihre Hochzeitsinformationen

Hochzeitsdatum	09.09.2016
Standesbeamter/in	Günter Rudolf
Standesamtliche Trauung	Ja, hier klicken für nähere Infos
Kirchliche Trauung	Nein, hier klicken um hinzuzufügen



Erika Mustermann

Braut


Persönliche Daten

Name:  
Erika Mustermann

Geburtsdatum:  
21.07.1995

E-Mail:  
erika.mustermann@mail.de

Telefonnummer:  
0800-123-123-6



Max Mustermann

Bräutigam

Persönliche Daten

Name:  
Max Mustermann

Geburtsdatum:  
02.05.1995

E-Mail:  
max.mustermann@mail.de

Telefonnummer:  
0800-678-123-4

Kostenübersicht

35,00€
Bezahlte Kosten
145,00€
Geplante Kosten
180,00€
Gesamte Kosten

## 8.2. Aktivitätensicht

Der nächste Screenshot repräsentiert die allgemeine Sicht auf alle geplanten Aktivitäten. Hierbei steht ein Kalender im Fokus, der sämtliche Tage markiert, an denen eine oder mehr Aktivitäten stattfinden. Zusätzlich befindet sich unter dem Kalender eine Liste mit allen Aktivitäten. Wenn der Nutzer im Kalender einen Tag selektiert, so werden ebenfalls die Einträge in der Liste gefiltert, sodass nur die Aktivitäten angezeigt werden, die ebenfalls an dem gewählten Tag stattfinden. So kann der Nutzer auf einfache Arte und Weise Aktivitäten anhand ihres Datums finden. Zusätzlich hat der Nutzer die Möglichkeit innerhalb der Liste über das Suchfeld eine Freitextsuche zu starten, die die Listeneinträge anhand der Eingabe filtert. Hierbei können beide Suchen ebenfalls kombiniert werden um das maximal beste Suchergebnis zu liefern. Dadurch wird die Benutzerfreundlichkeit stark erhöht.

Die Listeneinträge repräsentieren die wichtigsten Daten einer Aktivität. Hierbei bildet der Icon am Anfang eines Eintrages die Art der Aktivität ab. Ein Blatt Papier steht für eine normale Aktivität, wie beispielsweise Tischkarten schreiben, Salate richten oder andere Aufgaben. Ein Puzzle hingegen verkörpert einen Unterhaltungsbeitrag. In der Mitte des Listeneintrags findet man den Titel der entsprechenden Aktivität, sowie deren Anfangs- und Endzeit. Am Ende gibt ein definierter Wert den Überblick über den aktuellen Status des Eintrags. Dieser wurde zusätzlich farblich markiert um den aktuellen Prozessstatus stärker hervorzuheben. Bei einem Klick auf einen entsprechenden Listeneintrag kann dieser näher im Detail betrachtet werden. Dadurch gelangt man auf die Detailseite des entsprechenden Eintrags (siehe nächster Screenshot).

<

August2016

>

	Sun	Mon	Tue	Wed	Thu	Fri	Sat
31	31	1	2	3	4	5	6
32	7	8	9	10	11	12	13
33	14	15	16	17	18	19	20
34	21	22	23	24	25	26	27
35	28	29	30	31	1	2	3

Today

Selected

Working Day

Non-Working Day

Geplante Aktivitäten

Aktivitäten am 26. August

Search

	Salat richten (für abends) 09.00 Uhr - 10.30 Uhr	Erledigt
	Blumen holen 16.30 Uhr - 17.30 Uhr	Geplant
	Polterabend 20.30 Uhr - 01.30 Uhr	Geplant

### 8.3. Detailansicht

Der letzte Screenshot visualisiert die Detailansicht einer Aktivität. Für andere Entitätstypen ist die Ansicht ähnlich, entsprechend derer Attribute. Hierbei werden alle Attribute und Relationen zu anderen Objekten auf einer Seite angezeigt. Diese ist mit einer Scrollbar ausgestattet um sämtlichen Inhalt zu erreichen.

Die Detailseite wird mit den wichtigsten Stammdaten der Aktivität eingeleitet. Diese bilden zu einem der Titel als Überschrift, sowie der Aktivitätszeitraum, eine Verlinkung auf die Lokalität, sowie ein zusätzlich mit angegebenes Kommentarfeld. Da diese Daten wichtige Stammdaten sind, nehmen sie die komplette Bildschirmbreite ein, um den Fokus des Nutzers zuerst darauf zu richten. Anschließend folgen weitere Unterobjekte, die mit der Aktivität referenziert sind. Über die Werkzeugleiste können alle Stammdatenfelder bearbeitet werden, sowie die komplette Aktivität gelöscht werden. Zusätzlich hat der Nutzer die Möglichkeit den aktuellen Status der Aktivität per Auswahlliste zu setzen, oder die Aktivität zu einem Unterhaltungsbeitrag zu wandeln.

Zu einem bietet die Anwendung die Möglichkeit angehängte Medien mit Hilfe eines Karussells wiederzugeben. Hierbei kann der Nutzer sich durch alle angehängten Medien klicken, sofern diese die Dateiformate JPG, PNG oder GIF besitzen. Bei anderen Dateiformaten kann keine Vorschau angezeigt werden und es erscheinen lediglich ein Bildplatzhalter, sowie der Dateiname der Datei. Die Datei kann über den Download-Button in der Werkzeugleiste heruntergeladen werden. Mit einem Klick auf den Mülltonnen-Button, kann das derzeit angezeigt Medium gelöscht werden. Rechts neben der Medienanzeige befindet sich eine Kurzübersicht zur Kostenüberwachung. Hierin werden sämtliche Kosten der Hilfsmittel und des Caterings für eine spezielle Aktivität aufsummiert und anhand ihres Status dem Feld „geplant“ oder „bezahlt“ zugewiesen. Dadurch sieht der Nutzer jederzeit, ob eine geplante Aktivität noch im Kostenrahmen liegt.

Unter der Medienanzeige und den Kosten finden sich zwei Personenlisten wieder. Hier können Teilnehmer und verantwortliche Personen definiert werden. Bei Klick auf eine Person gelangt der Nutzer auf die Detailseite der entsprechenden Person. Weitere Personen können über den Plus-Button hinzugefügt werden. Die Mülltonne erlaubt das Löschen von Personen. Weiterhin kann unter den Personen mit einer Freitextsuche gesucht werden. Nach den Teilnehmern folgen die Hilfsmittel und das Catering, welche jeweils den dazugehörigen Caterer/Hilfsmittelobjekt darstellen, sowie die Menge. Zusätzlich wird dem Nutzer noch die aktuellen Kosten angezeigt, farblich codiert nach dem aktuellen Kostenstatus. Nach Doppelklick auf das entsprechende Item, erscheint ein Dialog für die Verpflegungs-/Hilfsmittelplanung.





## 9. Relationales Datenbank Mapping

Im folgenden Abschnitt wird der komplette Datenbankentwurf der Hochzeitsplanungsanwendung näher betrachtet. Hierbei stehen die Übersetzung der typischen JAVA-Datentypen, sowie die Relation der einzelnen Datenbanktabellen und deren Schlüssel im Vordergrund. Weiterhin werden lediglich die Datenklassen in den Datenbankentwurf mit aufgenommen. Der Datenbankentwurf orientiert sich am Entwurfsklassendiagramm. Hierbei können bestehende Objekte und Beziehungen übernommen und innerhalb des relationalen Datenbankmodells angepasst werden. Neue Tabellen wurden nur an obligatorischen Stellen eingefügt, wie beispielsweise bei Many-To-Many Beziehungen, um die entsprechenden Normalformen zu erfüllen.

### *Identifizier*

Jeder Tupel innerhalb einer Tabelle besitzt im Datenbankentwurf ein systemweites eindeutiges künstliches Attribut, welches als Schlüsselattribut dient. Dieses wird in der Regel durch den Anfangsbuchstaben der entsprechenden Tabelle, sowie der anschließenden Kennung ID benannt. Weiterhin repräsentiert eine ID einen 32-bit Integer. Hierdurch können genügend Entitäten in der Datenbank abgebildet werden.

### *Hochzeitsprojekt*

Das Hochzeitsprojekt ist das zentrale Element in der Datenbank. Hierfür gibt es konkrete Nutzer, die sich auf darauf einloggen können. Alle Aktivitäten, sowie die unterschiedlichen Trauungen besitzen eine Relation zum Hochzeitsprojekt. Durch weitere Relationen können so Beziehungen zwischen mehreren Tabellen hergestellt werden.

### *Datenklassen*

Der Inhalt der Datenklassen wird hier nicht näher beschrieben. Informationen hierzu können aus dem Analyseklassen-, sowie aus dem Entwurfsklassendiagramm entnommen werden. Die Datenfelder der einzelnen Tabellen wurden soweit wie möglich aus den vorigen Entwürfen entnommen. Hierbei wurde lediglich der JAVA Typ auf äquivalente Datenbanktypen angepasst.

### *Entfernte Felder*

Im direkten Vergleich zum Analyseklassendiagramm fällt auf, dass manche Felder nicht mit in die Datenbank übernommen wurden. Dies liegt daran, dass beispielsweise die Kalkulation von Kosten direkt in der Anwendung stattfindet. Hierbei werden die einzelnen Kosten über die Beziehungen

ermittelt und aufaddiert. Eine Speicherung auf der Datenbank aufgrund des höheren Wartungsaufwands findet nicht statt (Auflösung von Redundanzen).

### *Referenzbildung*

Referenzen zwischen den typischen Datentabellen wurden immer über die künstlichen ID-Felder realisiert. Hierbei wurde, je nach Art der Beziehung, der Primärschlüssel einer Tabelle als Fremdschlüssel in einer anderen Tabelle aufgenommen.

### *Planungstabellen*

Das Relationale Datenbankmodell beinhaltet diverse Planungstabellen. Diese wurden eingefügt, um sogenannte Many-To-Many-Beziehungen aufzulösen. Hierbei werden die jeweiligen Primärschlüssel beider in Relation stehenden Tabellen in einer weiteren Tabelle zusammengetragen. Hierbei mussten im Vergleich zum Analyseklassendiagramm zwei neue Tabellen eingefügt werden.

### *Zusatztabellen*

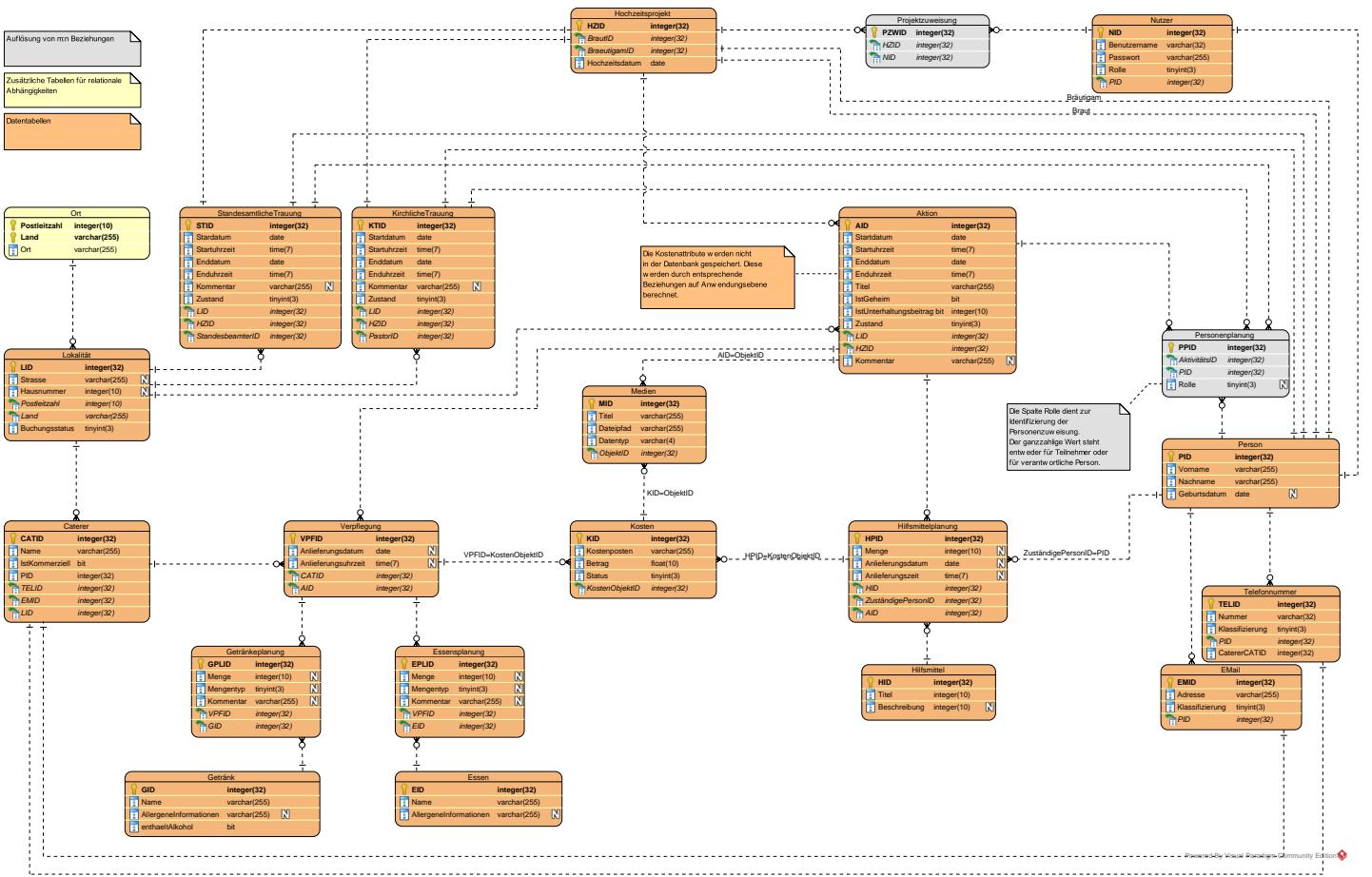
Im Diagramm wurden weiterhin Zusatztabellen (gelb markiert) eingefügt. Diese werden benötigt um weitere Normalformen innerhalb der Datenbank zu erfüllen. So sorgt die Tabelle Ort dafür, dass das Feld mit der Ortsangabe nur von den Primärschlüsseln abhängig ist.



Auflösung von m:n Beziehungen

Zusätzliche Tabellen für relationale Abhängigkeiten

Entitätstabellen



## 10. Besonderheiten

Um die Software im Vergleich zu anderen Produkten hervorzuheben und dem Benutzer bei der Planung der Hochzeit alles zu erleichtern, wurden einige zusätzliche Funktionen implementiert. Die implementierten Funktionen werden im Folgenden kurz aufgezählt und erläutert.

### 10.1. Backups

Der Benutzer hat die Möglichkeit direkt aus der Anwendung Backups zu erstellen. Dafür wird keine externe Software benötigt. Die ist vor allem für Firmen wichtig, damit sie die Backups nichts „künstlich“ erzeugen müssen. So können sie die Backups direkt aus der Software herstellen lassen und wenn nötig, wieder einlesen.

### 10.2. Import & Export

Durch den Import und Export von Daten wird es den Benutzern ermöglicht, Daten untereinander auszutauschen. Die Funktionalität ist vor allem wichtig, wenn nicht der globale Server verwendet wird.

### 10.3. Übersicht der Aktivitäten im Kalender

Um unseren Benutzern eine einfache und übersichtliche Möglichkeit bieten zu können, alle Aktivitäten auf einen Blick zu sehen, wurde ein Kalender implementiert. Durch nur einen kurzen Blick auf den Kalender weiß er, ob an dem Tag etwas geplant ist oder nicht.

### 10.4. E-Mail Versand über die Software

Sollen E-Mails versendet werden, macht dies meistens viel Arbeit, da jede E-Mail einzeln geschrieben werden muss. Unsere Software bietet den Benutzern an, E-Mails automatisiert zu verschicken. Dabei kann der Benutzer einstellen, dass der Name der Person als Ansprache genommen wird und spart einiges an Arbeit.

### 10.5. Globaler Server für alle Benutzer

Bei der Planung der Hochzeit stehen viele Aktionen an und es werden viele Hilfsmittel benötigt. Um den Benutzern das Planen zu erleichtert, gibt es den globalen Server. Auf diesem sind schon Hilfsmittel und andere benötigte Daten angelegt und der Benutzer kann diese einfach verwenden.