


DeliFHery

Das Projekt.

DeliFHery ist ein neuer Paketdienst. Sie entwickeln nun die Software zur Paketverfolgung und behalten den Überblick über den Standort sämtlicher Pakete.

Der hier beschriebene Ablauf beschreibt die Anforderungen im Überblick. Die Punkte, die nur mit „API“ gekennzeichnet sind, sind ausschließlich im Backend als API umzusetzen. Für die Punkte, die als „Frontend“ markiert sind, ist auch eine Web-Oberfläche zu gestalten. Anforderungen, die mit dem Symbol  gekennzeichnet sind, müssen nur von Zweierteams ausgearbeitet werden. Alle anderen Anforderungen gelten für alle.

Funktion		API	Frontend
Registrierung			
1	Kunden registrieren sich	✓	✓
2	Kunden verwalten ihre Kontaktmöglichkeiten	✓	✓
Paketversand			
3	Absender berechnen den Preis für ein Paket	✓	✓
4	Absender melden ein Paket an und () drucken Etiketten	✓	✓
5	Das System startet die Zahlungsabwicklung	✓	
6	Die Zahlungsabwicklung wird abgeschlossen	✓	
7	Versanddienstleister pflegen Pakete direkt ohne Anmeldung ein ()	✓	
Statustracking			
8	Der Status eines Pakets wird abgefragt	✓	✓
9	Kunden aktivieren Benachrichtigungen für ein Paket	✓	✓
10	Das System informiert Empfänger über eine Statusänderung	✓	
11	Versanddienstleister aktualisieren den Status eines Pakets	✓	
12	Darstellung einer Zustellstatistik in Diagrammen für Absender		✓

Die Details.

Nummer	1
Titel	Kunden registrieren sich und melden sich an
Daten	<ul style="list-style-type: none">• Benutzername• Passwort
Hinweise	<ul style="list-style-type: none">• Kunden können sich kostenlos registrieren, um z. B. Pakete anmelden oder Benachrichtigungsfunktionen aktivieren zu können.
Web UI	Überlassen Sie diese Aufgaben dem Identity Provider. Für eine Benotung mit Sehr Gut (WEA) setzen Sie einen lokalen KeyCloak ein.
Lernfelder	<ul style="list-style-type: none">• OAuth & Open ID Connect

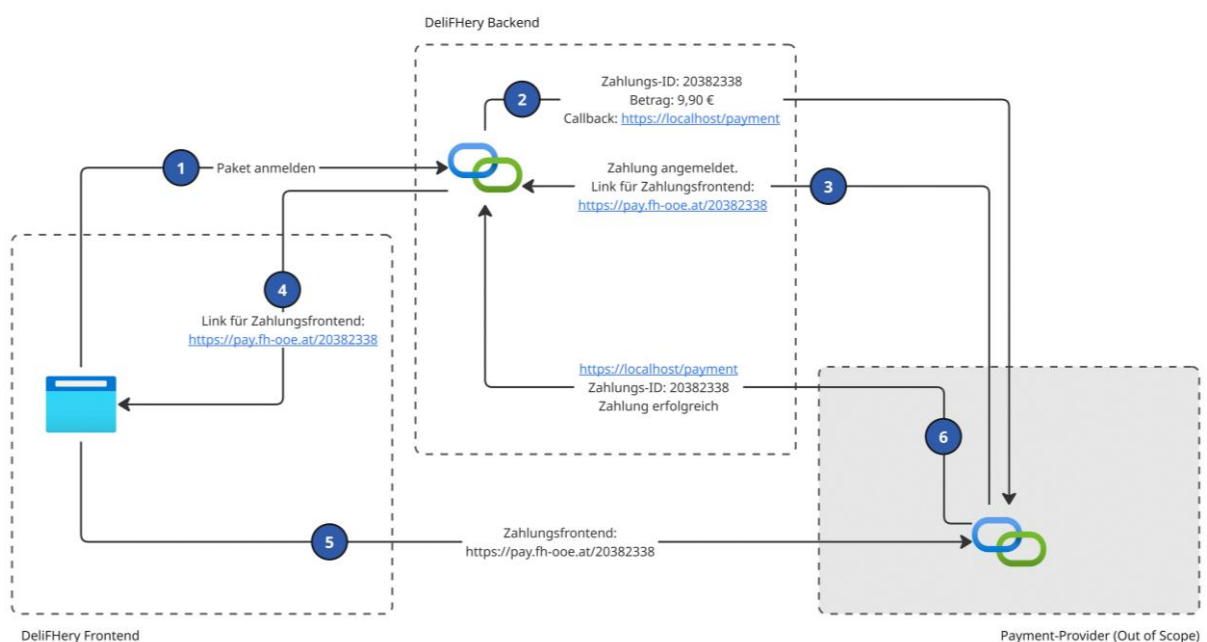
Nummer	2
Titel	Kunden verwalten ihre Kontaktmöglichkeiten
Daten	<ul style="list-style-type: none">• E-Mail-Adressen• Telefonnummern
Hinweise	<ul style="list-style-type: none">• Ein Aufruf darf nur authentifiziert erfolgen• Es sollen sowohl E-Mail-Adressen als auch Telefonnummern verwaltet werden• Kontaktmöglichkeiten sollen hinzugefügt, abgefragt, geändert und gelöscht werden können
Beispiele	<ul style="list-style-type: none">• max.mustermann@fhooe.at• fam.mueller@gmail.com• 05 0804 20
Web UI	Achten Sie auf eine übersichtliche Darstellung und setzen Sie geeignete Validatoren ein.
Lernfelder	<ul style="list-style-type: none">• Validierung von Eingabefeldern (bspw. in Formularen)• Design des Datenmodells• Einfache CRUD-Operationen als API

Nummer	3
Titel	Absender berechnen den Preis für ein Paket
Daten	<ul style="list-style-type: none"> • Absenderadresse • Empfängeradresse • Breite • Länge • Höhe • Gewicht
Hinweise	<ul style="list-style-type: none"> • Dieser Aufruf darf auch von anonymen Benutzern durchgeführt werden. • Dieser Aufruf liefert zurück, wie viel es kosten würde, ein Paket mit den übergebenen Daten zu versenden. Es wird aber noch kein Versand ausgelöst. • Die Preisberechnung erfolgt aufgrund verschiedener Regeln, die sich häufig ändern können. Es soll einfach und mit wenig Änderungen am Code möglich sein, neue Regeln hinzuzufügen oder alte zu entfernen. Es können auch mehrere Regeln kombiniert werden. • Implementieren Sie zumindest drei verschiedene Regeln, die jeweils von verschiedenen Parametern abhängig sind. • Stellen Sie sicher, dass Ihre Regeln korrekt funktionieren, indem Sie sie ausführlich automatisiert testen.
Beispiele	<ul style="list-style-type: none"> • Dreistufiges Preismodell mit Maximalwerten für Dimensionen und Gewicht • Versandkosten abhängig von der Anzahl der Bundesländer, die das Paket durchqueren muss • Saisonale Rabatte (-10% im Dezember)
Web UI	<ul style="list-style-type: none"> • Setzen Sie geeignete Komponenten ein, um Eingabefehler zu verhindern. • Beachten Sie, dass eine Berechnung unter Umständen mehr Zeit in Anspruch nimmt. Vermeiden Sie missverständliche Anzeigen.
Lernfelder	<ul style="list-style-type: none"> • Flexible Implementierung der Geschäftslogik • Abstraktion von Regeln • Automatisierte Tests

Nummer	4
Titel	Absender melden ein Paket an und (📦) drucken Etiketten
Daten	Siehe 3
Hinweise	<ul style="list-style-type: none"> • Dieser Aufruf darf nur authentifiziert durchgeführt werden, Absender müssen also registrierte Kunden sein. • Dieser Aufruf erledigt folgende Aufgaben: <ul style="list-style-type: none"> ○ Speichern der Sendung im System ○ Generierung einer eindeutigen Tracking-Nummer, die aus Sicherheitsgründen schwer zu erraten (und somit nicht fortlaufend), aber immer noch möglichst kurz und lesbar sein muss ○ Berechnung der Kosten (siehe 3) ○ Aufruf eines externen Services zum Start der Zahlungsabwicklung (siehe 5) ○ (📦) Generierung eines Etiketts mit Adresse des Empfängers und der Tracking-Nummer als Barcode in Form einer Grafik • Der Aufruf liefert die Tracking-Nummer, den Preis, einen Link zum Zahlungssystem (siehe 5) und (📦) das Etikett (z.B. Base64-codiert).
Beispiele	<ul style="list-style-type: none"> • Aufruf <ul style="list-style-type: none"> ○ Absender: FH Oberösterreich, Softwarepark 11, 4232 Hagenberg ○ Empfänger: Max Mustermann, Hauptplatz 13, 4020 Linz ○ Paket: 32cm x 20 cm x 14 cm (1,2 kg) • Antwort <ul style="list-style-type: none"> ○ Tracking-ID: 29502739293 ○ Preis: 9,90 € ○ Zahlungslink: https://payment.fh-ooe.at/af4d7d8a ○ Bild des Etiketts (📦)
Web UI	<ul style="list-style-type: none"> • Nutzen Sie zur Eingabe Komponenten, die Sie z.B. bereits bei der Berechnung verwendet haben.
Lernfelder	<ul style="list-style-type: none"> • Design der Geschäftslogik • Automatisierte Tests • Bild-Generierung (📦)

Nummer	5
Titel	Das System startet die Zahlungsabwicklung
Daten	<ul style="list-style-type: none"> • Zahlungsanbieter-API-Key • Eindeutige Identifikation der Zahlung • Betrag • Callback-URL • Redirect-URL
Hinweise	<ul style="list-style-type: none"> • Die Zahlungsabwicklung ist nicht Teil Ihres Systems, sondern erfolgt in einem externen Service in einem zweistufigen Prozess. Dies ist Schritt 1, für Schritt 2 siehe Punkt 6. DeliFHery schickt einen API-Key des Zahlungsanbieters zur Authentifizierung, eine frei wählbare, eindeutige ID der Zahlung, den gewünschten Betrag und eine Callback-URL an das Zahlungsservice. • Das Zahlungsservice antwortet mit einer URL für die Zahlungsabwicklung. DeliFHery leitet diesen an den Client (Web-UI) weiter. • Zu Testzwecken bekommen Sie eine lokal ausführbare Simulation des Zahlungsservices zur Verfügung gestellt.
Beispiele	<ul style="list-style-type: none"> • API-Key: OPA-12345-67890 • Zahlung: 35789829 • Betrag: € 9,90 • Callback-URL: https://localhost:5000/payment (Backend) • Redirect-URL: https://localhost:6000/payment-complete (Frontend) • Antwort: https://pay.fh-ooe.at/20382338
Lernfelder	<ul style="list-style-type: none"> • Abbilden komplexer Workflows und Aufruf eines externen Service • Abstraktion des Services zur besseren Testbarkeit Ihres Systems • Umgang mit API-Keys

Nummer	6
Titel	Die Zahlungsabwicklung wird abgeschlossen
Daten	<ul style="list-style-type: none"> • Zahlungsanbieter-API-Key • Eindeutige Identifikation der Zahlung • Status (erfolgreich/fehlgeschlagen)
Hinweise	<ul style="list-style-type: none"> • Nachdem der Client die Zahlung aus Schritt 5 durchgeführt oder abgebrochen hat, informiert das Zahlungsservice DeliFHery darüber durch einen Aufruf der übermittelten Callback-URL. • Zusätzlich benachrichtigt das Zahlungsservice auch das Frontend über den Abschluss der Zahlung, indem es den Browser auf die Redirect-URL weiterleitet. Diese enthält die Zahlungs-ID sowie den Zahlungsstatus. • Es dürfen nur Aufrufe mit dem korrekten API-Key angenommen werden, der auch in Schritt 5 übermittelt wurde. Damit ist sichergestellt, dass nur der Zahlungsanbieter eine Zahlung abschließen kann. • War die Zahlung erfolgreich, bekommt die zugehörige Sendung den Status „Angenommen“ und der Empfänger wird gegebenenfalls benachrichtigt (siehe 9). Ansonsten ändert sich der Status der Sendung nicht, der Absender kann den Zahlungsprozess aber erneut starten.
Beispiele	<ul style="list-style-type: none"> • API-Key: OPA-12345-67890 • Zahlung: 35789829 • Status: Zahlung erfolgreich
Lernfelder	<ul style="list-style-type: none"> • API-Design • Umgang mit API-Keys



Zusammenspiel der Anforderungen 5 und 6 mit dem Payment-Provider

Nummer	7
Titel	Versanddienstleister pflegen Pakete direkt ohne Anmeldung ein (👤)
Daten	Siehe 3, DeliFHery-API-Key
Hinweise	<ul style="list-style-type: none"> • Wird ein Paket nicht vorab vom Absender angemeldet, sondern direkt in einer Filiale des Versanddienstleisters abgegeben und bezahlt, wird die Sendung vom Versanddienstleister ins System eingepflegt. • Dieser Schritt erfolgt analog zu 4, allerdings mit folgenden Abweichungen: <ul style="list-style-type: none"> ○ Die Authentifizierung erfolgt über einen API-Key ○ Die Bezahlung über das externe Service fällt weg ○ Die Sendung beginnt direkt im Status „Erhalten“
Lernfelder	<ul style="list-style-type: none"> • Wiederverwendung von Code

Nummer	8
Titel	Der Status eines Pakets wird abgefragt
Daten	<ul style="list-style-type: none"> • Abfrage <ul style="list-style-type: none"> ○ Tracking-Nummer ○ Postleitzahl • Antwort: <ul style="list-style-type: none"> ○ Absender ○ Empfänger ○ Status-Historie
Hinweise	<ul style="list-style-type: none"> • Absender und Empfänger können die Status-Historie ihrer Pakete abfragen • Dieser Request ist ohne Authentifizierung möglich, die Angabe der Tracking-Nummer und der Postleitzahl genügt.
Beispiele	<ul style="list-style-type: none"> • 01.10.2025 16:45 Angemeldet • 02.10.2025 08:37 Erhalten (Linz) • 02.10.2025 09:18 In Verteilung (Linz) • 02.10.2025 15:12 In Verteilung (Pregarten) • 03.10.2025 07:55 In Zustellung • 03.10.2025 11:28 Zugestellt
Web UI	<ul style="list-style-type: none"> • Achten Sie auf eine gute Darstellung von Fehlern.
Lernfelder	<ul style="list-style-type: none"> • API-Design

Nummer	9
Titel	Kunden aktivieren Benachrichtigungen für ein Paket
Daten	<ul style="list-style-type: none"> • Tracking-Nummer • Postleitzahl
Hinweise	<ul style="list-style-type: none"> • Angemeldete Kunden können – unter Angabe der Tracking-Nummer und der Postleitzahl – Benachrichtigungen für ein Paket aktivieren. • Ab diesem Zeitpunkt erhalten sie bei jeder Statusänderung automatisiert eine E-Mail-Nachricht an die hinterlegten Kontaktmöglichkeiten (siehe Punkt 10).
Web UI	<ul style="list-style-type: none"> • Vermeiden Sie unnötige Benutzerinteraktionen. Stellen Sie den aktuellen Status klar dar.
Lernfelder	<ul style="list-style-type: none"> • API-Design

Nummer	10
Titel	Das System informiert Empfänger über eine Statusänderung
Daten	<ul style="list-style-type: none"> • Tracking-Nummer • Status
Hinweise	<ul style="list-style-type: none"> • Wenn sich der Status einer Sendung ändert, für die der Kunde eine Benachrichtigung aktiviert hat, soll dieser mit einer E-Mail über den neuen Status benachrichtigt werden. • Die E-Mail soll zumindest die Tracking-Nummer der Sendung, besser noch einen Link beinhalten, mit dem Details zur Sendung abgefragt werden können (siehe 8).
Lernfelder	<ul style="list-style-type: none"> • E-Mail-Versand

Nummer	11
Titel	Versanddienstleister aktualisieren den Status eines Pakets
Daten	<ul style="list-style-type: none"> • DeliFHery-API-Key • Eindeutige Identifikation der Sendung • Status • Zusatzinformation
Beispiele	<ul style="list-style-type: none"> • API-Key: XYZ-12345... Sendung: 123 Status: Angenommen • API-Key: XYZ-12345... Sendung: 456 Status: In Verteilung Zusatzinformation: Linz • API-Key: XYZ-12345... Sendung: 789 Status: Zugestellt Zusatzinformation: In Garage abgestellt
Hinweise	<ul style="list-style-type: none"> • Im Laufe ihrer Zustellung durchläuft eine Sendung verschiedene Stationen. Dies passiert zum Teil automatisch (z.B. Scan des Etiketts im Verteilungszentrum) und zum Teil manuell (Kennzeichnung der Sendung als „Zugestellt“ durch die Zustellerin). • In jedem Fall wird das System durch einen mittels API-Key authentifizierten Aufruf über die Statusänderung informiert. • Je nach Status ist das Feld „Zusatzinformation“ optional oder ein Pflichtfeld • Gegebenenfalls ist der Empfänger über die Statusänderung zu informieren (siehe 10).
Lernfelder	<ul style="list-style-type: none"> • Design des Datenmodells • Validierung und Plausibilitätsprüfung • Authentifizierung

Nummer	12 (nur für WEA)
Titel	Darstellung einer Zustellstatistik in Diagrammen für Absender <i>für „Sehr gut“ in WEA notwendig</i>
Daten	<ul style="list-style-type: none"> • Zeitraum, der ausgewertet werden soll • Siehe oben
Beispiele	Wie viele Pakete wurden erfolgreich bzw. nicht zugestellt, etc., welche Zustelladressen machen Probleme, etc.
Hinweise	<ul style="list-style-type: none"> • Dieser Aufruf darf nur authentifiziert von Absendern durchgeführt werden, Absender müssen also registrierte Kunden sein. • Eventuell sind aber Erweiterungen an den REST-Services notwendig • Stellen Sie die Daten im Frontend übersichtlich mit Hilfe von Diagrammen dar. • Erwartet wird zumindest eine Auswertung für Woche/Monat/Jahr pro Route
Lernfelder	<ul style="list-style-type: none"> • Aufbereitung der Daten im UI in Form von Diagrammen

Technologische Hinweise

Setzen Sie die beschriebenen Anforderungen mit den aus SWK5 und WEA5 bekannten Technologien und Konzepten um. Implementieren Sie die Datenzugriffsschicht auf Basis von ADO.NET. Der Einsatz eines OR-Mapping-Werkzeugs (NHibernate, Entity Framework etc.) ist nicht gestattet, weil diese Technologien erst im 6. Semester ausführlich behandelt werden und sie nicht Bestandteil des zu bewertenden Stoffumfangs sind.

Die Entwicklung der Projektarbeit für SWK5 ist auf GitHub durchzuführen. Alle Teams bekommen ein Repository zugewiesen, auf welches das Team und die Übungsleiter Zugriff haben. Für den WEA5-Teil darf das Repository weiterverwendet werden, die Entwicklung unter GitHub ist aber optional. Die finale Version der beiden Ausbaustufen in SWK5 sowie das Endergebnis des WEA5-Teils ist auf die E-Learning-Plattform hochzuladen. Bei Zweierteams müssen beide Teammitglieder den Upload durchführen.

Hinweis: GitHub Classroom stellt nur sehr eingeschränkt Ressourcen für GitHub Actions zur Verfügung, die sich alle Mitglieder eines Classrooms, also alle Studierenden, teilen müssen. Setzen Sie daher ein Werkzeug ein, mit dem Sie GitHub-Workflows auch lokal ausführen können.

Dokumentation

Dokumentation SWK5

Als Dokumentation zum Projekt bitten wir Sie um die Beantwortung der folgenden Fragen in Form einer README-Markdown-Datei im Stammverzeichnis Ihres Sourcecode-Repositories. Da wir – abgesehen von der Kommentierung des Quelltexts – keine weitere Dokumentation einfordern, erwarten wir, dass Sie sich mit diesen Fragen intensiv auseinandersetzen und diese ausführlich beantworten.

1. Für welches Datenmodell haben Sie sich entschieden? *ER-Diagramm, etwaige Besonderheiten erklären: Welche Entscheidungen mussten Sie treffen, wofür (und wogegen) haben Sie sich entschieden und warum?*
2. Dokumentieren Sie auf Request-Ebene den gesamten Workflow anhand eines durchgängigen Beispiels. Sie können ein Tool Ihrer Wahl einsetzen, z. B. Postman Workflows, VS Code, etc. *HTTP-Requests inkl. HTTP-Verb, URL, Parametern, Body und Headern*
3. Haben Sie im Zuge der Projektarbeit Erfahrungen mit dem Einsatz von KI-Agenten zur Generierung von Sourcecode gesammelt? Welche? *Was hat gut funktioniert, wo sind Sie gescheitert?*
4. Bei welchen Teilen Ihres Systems ist eine korrekte Funktionsweise aus Sicht von DeliFHery am wichtigsten? Welche Maßnahmen haben Sie getroffen, um sie zu gewährleisten?
5. Wie können Sie ihr System testen, ohne tatsächlich Zahlungen im externen Service auszulösen?
6. Wie haben Sie die Berechnung der Preise umgesetzt? Welche Teile Ihres Codes müssen Sie ändern, um eine andere oder neue Variante bereitzustellen?
7. Die Preisberechnung soll sich ab dem 1. Jänner, 0:00 auf ein neues Tarifmodell ändern. Wie würden Sie diese Anforderung lösen?
8. Welche Überlegungen haben Sie beim Generieren der eindeutigen Trackingnummer angestellt?

9. Wie haben Sie den E-Mail-Versand gelöst? Wie abhängig ist Ihre Implementierung von einem konkreten E-Mail-Versanddienst? Die Marketingabteilung möchte die Formatierung der E-Mails zukünftig selbst verändern können – wie würden Sie diese Anforderung lösen?
10. Denken Sie an die Skalierbarkeit Ihres Projekts: Die Österreichische Post möchte Ihr Produkt mit über 500 Millionen Paketen pro Jahr nutzen. Was macht Ihnen am meisten Kopfzerbrechen?
11. Wenn Sie das Projekt neu anfangen würden – was würden Sie anders machen?

Dokumentation WEA5

Die Dokumentation des Frontends (WEA5) muss eine **PDF**-Datei sein, welche folgende Kernbestandteile enthält:

1. Inhaltsverzeichnis
2. Dokumentation Ihrer Architektur: Beschreiben Sie die Architektur Ihres Projekts, so dass sich jemand Externer schnell in Ihrem Projekt zurechtfindet, ohne lange den Quelltext durchforsten zu müssen (Komponentenbaum, Zusammenhang Services mit Komponenten, wichtige Komponenten). Setzen Sie hierfür z.B. Compodoc ein: <https://compodoc.app/>
3. Stellen Sie mögliche Navigationswege in Ihrer Anwendung grafisch dar.
4. Kompletter bebildeter Testlauf Ihrer Anwendung. Für jedes Szenario muss zumindest ein Screenshot vorhanden sein, der die Funktionalität zeigt. Fügen Sie auch notwendige Erläuterungen hinzu.
5. Geben Sie an, welche KI-Werkzeuge Sie eingesetzt haben und welche Code-Abschnitte explizit mit KI-Werkzeugen erstellt wurden.
6. Beantworten Sie folgende Fragen:
 - a. Was ist zu tun, wenn sich URLs ändern? Wie invasiv ist der Eingriff in Ihre Anwendung, um diese zu ändern?
 - b. Wie stellen Sie sicher, dass bestimmte Seiten nur nach einem Login zugreifbar sind?
 - c. Wie stellen Sie eine korrekte Dateneingabe sicher?
 - d. Was passiert, wenn Aufrufe an das Backend Fehler produzieren?
7. Vor der Präsentation der Projektarbeit wird es einen Fragebogen in Moodle geben, Sie werden zeitgerecht per E-Mail informiert. Beantworten Sie diesen, ein Versäumnis wird mit einem schlechteren Notengrad bewertet.

Ausbaustufe 1 (SWK5)

In der ersten Ausbaustufe sind die folgenden Aufgaben zu erledigen:

Definieren Sie Ihren Entwicklungsprozess

- ☐ Sourcecodeverwaltung einrichten
- ☐ Ordnerstruktur überlegen und umsetzen
- ☐ Zusammenarbeit in 2er-Teams organisieren (Branching, Pull Requests, etc.)
- ☐ Issue-Tracking-System einrichten (👤)
- ☐ Build-Pipeline einrichten (👤)

Erstellen Sie die grundlegende Projektstruktur

- ☐ Visual Studio-Solution mit Projektstruktur anlegen
- ☐ Vorbereitung der Dokumentation (README.md)
- ☐ Projekt für Unit-Tests anlegen

Implementierung *(Hinweis: Aus vergangenen Jahren wissen wir, dass manche Entscheidungen in der Datenzugriffsschicht erst in der zweiten Ausbaustufe sinnvoll getroffen werden können. Arbeiten Sie also nicht zu weit vor und konzentrieren Sie sich vorerst auf die genannten Entitäten).*

- ☐ Datenbank mit den Entitäten für Kunden und Kontaktmöglichkeiten
- ☐ Implementierung der Datenzugriffsschicht für die ersten Entitäten
- ☐ Test der Datenzugriffsschicht (exemplarisch für zumindest eine Entität)

Ausbaustufe 2 (SWK5)

In der zweiten Ausbaustufe sind die folgenden Aufgaben zu erledigen:

Implementierung der geforderten Funktionalität

- ☐ Design der API-Endpunkte
- ☐ Design des restlichen Datenmodells
- ☐ Implementierung der API und Erweiterung der Datenzugriffsschicht
- ☐ Testen Sie automatisiert vor allem die Bereiche,
 - die komplexere Logik beinhalten,
 - wo manuelles Testen aufwändig wäre oder
 - wo Sie zahlreiche Sonderfälle behandeln und testen müssen.
- ☐ Code-Dokumentation (wo sinnvoll)

Beantwortung der Konzeptfragen

- ☐ Beantwortung sämtlicher Fragen

Ausfüllen des Feedbackfragebogens

Ausbaustufe 3 (WEA)

In der dritten Ausbaustufe sind die folgenden Aufgaben zu erledigen:

Implementierung der geforderten Funktionalität

- ☐ Ansprechendes Design des Web Frontends (hier geht es weniger um das Aussehen selbst, sondern um die übersichtliche Darstellung, Darstellung nur sinnvoller Daten, sowie logischer, intuitiver und einfacher Abläufe für Benutzer*innen; "Benutzbarkeit") Achtung: Die Benutzer*innen sind nicht die Entwickler*innen selber!
- ☐ Webanwendung mit Angular (Version ≥ 20), welche mit dem Backend über REST-Service-Schnittstellen kommuniziert. Die Services zum Backend dürfen NICHT generiert werden.
- ☐ Auswahl und Einsatz geeigneter CSS-Frameworks, bzw. Angular Komponenten Bibliotheken (Bootstrap, Fomantic, Angular Material, DevExtreme, etc.).
- ☐ Auswahl und Einsatz geeigneter Services für OAuth-Login (KeyCloak, IdentityServer etc.).
- ☐ (Sinnvoller) Einsatz von zusätzlichen Angular-Konzepten wie Pipes, Direktiven, Modulen, etc. (siehe z.B. Angular Dokumentation), die nicht in der Übung verwendet wurden.
- ☐ Code-Dokumentation (wo sinnvoll)
- ☐ Dokumentation wie oben in der WEA5-Sektion beschrieben

Ausfüllen des Feedbackfragebogens

Hinweise zur Präsentation

Am Semesterende findet sowohl für SWK5 als auch für WEA5 eine getrennte Präsentation Ihres Projekts statt.

Ablauf SWK5-Präsentation

1. Live-Demonstration des gesamten API-relevanten Funktionsumfangs auf Request-Ebene (z. B. in Postman, Visual Studio Code etc.) – unter Berücksichtigung möglicher Änderungswünsche durch die Vortragenden (andere Paketgrößen, fehlgeschlagene Zahlungen, etc.).
2. Detailliertere Besprechung von zwei Fragen (eine darf von Ihnen nominiert werden, eine vom Vortragenden).

Ablauf WEA5-Präsentation

Live-Demonstration der gesamten Funktionalität in einem logischen Ablauf. Die Präsentation ist eine Einzelpräsentation, die anderen Studierenden müssen nicht die ganze Zeit über anwesend sein.

Hinweise zur Bewertung

Folgende Kriterien werden in die Beurteilung der Projektarbeit einbezogen:

- ☐ Qualität des Quelltexts sowie der gewählten Architektur
- ☐ Qualität des Frontends (weniger Designskills, mehr durchdachtes und benutzerfreundliches User-Interface)
- ☐ Qualität der Dokumentation (Beantwortung der Fragen)
- ☐ Gesamteindruck der Präsentation (strukturierte, gut vorbereitete Demo)
- ☐ Vollständigkeit der Implementierung
- ☐ Korrektheit der Implementierung
- ☐ Verbale Erläuterung der Implementierung
- ☐ Termintreue