

Projektarbeit Ski-Service Management

(Sozialform: Einzelarbeit)



Modul 295

Backend für Applikationen realisieren

Partnerarbeit „Ski-Service Management“

1 Ausgangssituation

Die Firma Jetstream-Service führt als KMU in der Wintersaison Skiservicearbeiten durch und will im Zuge der Digitalisierung die interne Verwaltung der Ski-Service Aufträge komplett über eine Web- und Datenbank basierte Anwendung abwickeln. Die bereits existierende Online-Anmeldung soll bestehen bleiben und mit den erforderlichen Funktionen für das Auftragsmanagement erweitert werden.

In der Hauptsaison sind bis zu 10 Mitarbeiter mit der Durchführung der Serverarbeiten beschäftigt. Diese sollen einen autorisierten passwortgeschützten Zugang zu den anstehenden Aufträgen erhalten und diese zur Abarbeitung übernehmen und ändern können.

Das Teilprojekt umfasst ausschliesslich den Backendteil und umfasst folgende Aufträge, welche nach IPERKA durchzuführen sind:

- Web-API Projekt mit Authentifikation (Backend) inkl. OpenAPI Dokumentation
- Datenbankdesign und Implementierung (Code First, oder Database First)
- Testprojekt / Testplan (Unit-Test)
- Realisierung der kompletten Anwendung, gemäss den Anforderungen
- Durchführung der Tests (Postman)

Bemerkung:

- Bei datenlesenden Operationen (z.B. Auftragsliste usw.) ist **keine** Authentifikation erforderlich.
- Änderungen von Auftragsdaten ist nur über eine Authentifikation des Mitarbeiters möglich.

2 Allgemeine Anforderungen

Das Auftragsmanagement muss folgende Funktionen zur Verfügung stellen:

- Login mit Benutzername und Passwort
- Anstehende Serviceaufträge anzeigen (Liste)
- Bestehende Serviceaufträge mutieren. Dazu stehen folgende Stati zu Verfügung: *Offen*, *InArbeit* und *abgeschlossen*
- Aufträge löschen (ggf. bei Stornierung)

Die Informationen zur Online-Anmeldung, welche bereits realisiert wurde, müssen ggf. bei Bedarf wie folgt ergänzt werden.

- Kundenname
- E-Mail
- Telefon
- Priorität
- Dienstleistung (Angebot), siehe nachfolgende Auflistung. Pro Serviceauftrag kann immer nur eine Dienstleistung zugeordnet werden.

Die Firma bietet folgende Dienstleistungen (Angebot) an:

- Kleiner Service
- Grosser Service
- Rennski-Service
- Bindung montieren und einstellen
- Fell zuschneiden
- Heisswachsen

2.1 Zusammenfassung der Anforderungen

Nr.	Beschreibung
A1	Login Dialog mit Passwort für den autorisierten Zugang der Mitarbeiter (Datenänderungen).
A2	In der Datenbank müssen die Informationen des Serviceauftrags und die Login Daten der Mitarbeiter verwaltet sein.
A3	Erfasste Serviceaufträge abrufbar sein.
A4	Die erfassten Serviceaufträge müssen selektiv nach Priorität abrufbar sein.
A5	Mitarbeiter können eine Statusänderung eines Auftrages vornehmen.
A6	Mitarbeiter können Aufträge löschen (z.B. bei Stornierungen)
A7	Die aufgerufenen API-Endpoints müssen zwecks Fehlerlokalisierung protokolliert sein (DB oder Protokolldatei).
A8	Datenbankstruktur muss normalisiert in der 3.NF sein inkl. referenzieller Integrität
A9	Für die Web-API Applikation muss ein eigener Datenbankbenutzerzugang mit eingeschränkter Berechtigung (DML) zur Verfügung gestellt werden (Benutzer root bzw. sa ist verboten).
A10	Das Web-API muss vollständig nach Open-API (Swagger) dokumentiert sein.
A11	Das Softwareprojekt ist über ein Git-Repository zu verwalten.
A12	Ganzes Projektmanagement muss nach IPERKA dokumentiert sein

3 Zusätzliche Anforderungen

Zusatzpunkte für optionale Erweiterungen. Zur Erreichung der max. Punktzahl müssen zwei optionale Anforderungen umgesetzt werden. Es werden nur zwei zusätzliche Anforderungen bewertet.

Nr.	Beschreibung
AO1	Die Mitarbeiter können zu einem Auftrag einen Freitext bzw. Kommentar hinterlegen
AO2	Ein Auftrag kann mit sämtlichen Datenfeldern geändert werden
AO3	Das Login des Mitarbeiters wird nach drei nachfolgenden Falschanmeldungen automatisch gesperrt.
AO4	Personalisierte Auftragsliste des eingeloggten Mitarbeiters. Der Mitarbeiter kann sich zusätzlich zur gesamten Auftragsliste nur die von ihm übernommenen Aufträge ansehen.
AO6	Eingeloggte Mitarbeiter können ein gesperrtes Login zurücksetzen.
AO7	Gelöscht Aufträge werden nicht aus der Datenbank entfernt, sondern nur als gelöscht markiert.

4 Randbedingungen

Es müssen folgende Randbedingungen eingehalten werden:

- Als Datenbanksystem ist MS-SQL oder MySQL zu verwenden.
- Postman ist als Web-API Test-Tool zu verwenden.
- Der Datenbankzugriff hat über einen OR-Mapper zu erfolgen

5 Kurzpräsentation

Sie stellen Ihre Ergebnisse mittels einer Kurzpräsentation der Klasse vor, präsentieren Sie Ihre Lösung in einer Live-Demo und schliessen Sie Ihre Präsentation mit einem kurzen Fazit ab (lessons learned).

Dauer der Kurzpräsentation : ca. 10-15 min

6 Bewertung

Bewertung	Punkte
Entwicklungswerkzeug inkl. Verwaltungssystem (Repository) eingerichtet	2
Änderungen werden dokumentiert und festgehalten (commit)	2
Coderichtlinien wurden eingehalten (Naming, Konventionen etc.)	2
Funktionen Auftragsliste anfordern implementiert	2
Funktion Statusänderung eines Auftrages implementiert	2
Funktion Auftrag löschen implementiert	2
Datenbank korrekt und vollständig	2
API-Routing konsistent	2
Protokollierung implementiert	2
Datenvalidierung in Controller Funktionen korrekt	2
Authentifikation implementiert (Basic, JWT)	2
Statusänderung eines Auftrages möglich	2
API vollständig dokumentiert (OpenAPI, Swagger)	2
Postman Collection vorhanden	2
Testplan / Testing	2
Optional Anforderungen	
Anforderung 1	2
Anforderung 2	2
Dokumentation	
Projektdokumentation mit IPERKA	6
Präsentation / Fachgespräch	
Systematischer Aufbau der Präsentation / Inhalt / Medienvielfalt	2
Gestaltung und Lesbarkeit der Folien	2
Lösung vollständig erläutert	2
Live-Demo	2
Fazit	2
Total	50

Die Bewertung setzt sich aus der Arbeit und der Präsentation zusammen und ergibt eine Schlussnote.

Notenskala: $\frac{\text{Erreichte Punktzahl} \times 5}{\text{Max. Punktzahl}} + 1 = \text{Note (auf 1/10 Noten gerundet)}$