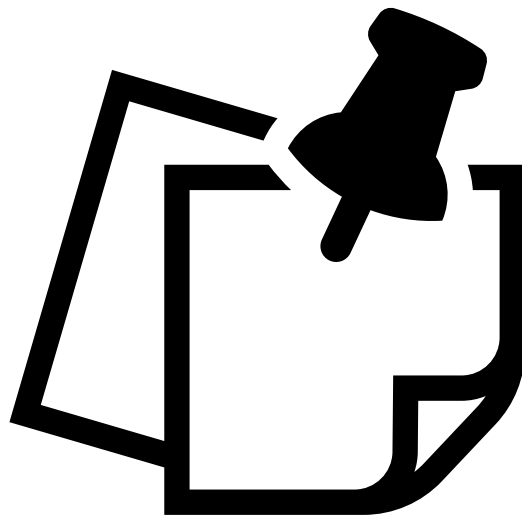


# Projektarbeit Ski-Service Auftragsverwaltung

---



## Modul 165

### NoSQL-Datenbanken einsetzen

# Einzelarbeit „Ski-Service Auftragsverwaltung“

## 1 Ausgangssituation

---

Die Firma Jetstream-Service führt als KMU in der Wintersaison Skiservicearbeiten durch und hat in den letzten Jahren grosse Investitionen in eine durchgängige digitale Auftragsanmeldung und Verwaltung, bestehend aus einer datenbankbasierender Web-Anmeldung und Auftragsverwaltung getätigt.

Aufgrund guter Auftragslage hat sich die Geschäftsführung für eine Diversifizierung mit Neueröffnungen an verschiedenen Standorten entschieden.

Die bis anhin eingesetzte relationale Datenbank genügt den damit verbundenen Ansprüchen an Datenverteilung und Skalierung nicht mehr. Um einerseits den neuen Anforderungen gerecht zu werden sowie andererseits Lizenzkosten einzusparen, soll im Backend der Anwendung die Datenbank auf ein NoSQL Datenbanksystem migriert werden.

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

- Datenbankdesign und Implementierung (NoSQL)
- Datenmigration (SQL → NoSQL)
- Migration WebAPI Projekt (siehe Modul 295)
- Testprojekt / Testplan
- Realisierung der kompletten Anwendung, gemäss den Anforderungen
- Durchführung Integrationstest mit bestehenden Frontend Lösung.

## 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	Datenbasis aus relationaler Datenbank vollständig nach NoSQL migriert
A2	Benutzerkonzept mit min. 2 Benutzeranmeldungen mit verschiedenen Berechtigungsstufen implementiert.
A3	Für die Web-API Applikation muss ein eigener Datenbankbenutzerzugang mit eingeschränkter Berechtigung (DML) zur Verfügung gestellt werden
A4	Schema für Datenkonsistenz implementiert
A5	Datenbank Indexe für schnelle Ausführung von Suchabfragen implementiert
A6	Backup und Restore Möglichkeiten umgesetzt (Skript-Dateien)
A7	Vollständige Datenbankmigration mittels Skript-Dateien realisiert
A8	Das Web-API Projekt (CRUD) komplett auf NoSQL Datenbanksystem migriert
A9	Datenmodell vollständig dokumentiert, inkl. Grafik zum Datenmodell
A10	Einfaches Testprojekt in Postman erstellt.
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	Automatisiertes Backup-Konzept durchgeführt u. implementiert.
AO2	Komplexe Schema Validierungen umgesetzt (Referenzen, enum, min, max. usw)
AO3	Datenmigrationsskripte zu den RDBMS nach NoSQL realisiert
AO4	Komplexes Datenmodell mit mehr als 6 Grundtypen (Collection / Labels) implementiert
AO5	Komplexe statistische Auswertungsabfragen realisiert

## 4 Randbedingungen

Es müssen folgende Randbedingungen eingehalten werden:

- Als NoSQL-Datenbanksystem ist MongoDB oder Neo4j zu verwenden.
- WebAPI ist Java, C# oder Python zu realisieren
- Postman ist als Web-API Test-Tool zu verwenden.

## 5 Kurzpräsentation

Sie stellen Ihre Ergebnisse mittels einer Kurzpräsentation der Klasse vor, präsentieren Sie Ihre Webseite 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
Datenbasis aus SQL komplett in NoSQL implementiert	2
Benutzerkonzept dokumentiert u. realisiert	2
Schema Erweiterungen realisiert	2
Index-Strukturen angelegt	2
Backup u. Restore Skript erstellt	2
Migration der Datenbasis von SQL nach NoSQL durchgeführt (Skript Dateien)	2
WebAPI auf NoSQL Datenbank migriert	2
Datenmodell komplett dokumentiert	2
Testprojekt zu WebAPI erstellt (z.B. Postman Collection)	2
Git Repository angelegt und Git Code Verwaltung	2
<b>Optional Anforderungen</b>	
Anforderung 1	2
Anforderung 2	2
<b>Dokumentation</b>	
Projektdokumentation mit IPERKA	6
<b>Präsentation / Fachgespräch</b>	
Systematischer Aufbau der Präsentation / Inhalt / Medienvielfalt	2
Gestaltung und Lesbarkeit der Folien	2
Lösung vollständig erläutert	2
Live-Demo	2
Fazit	2
<b>Total</b>	<b>44</b>

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

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