

Projektdokumentation: WueKaStL

Uni-Würzburger Kaffeestrichliste

1. Projektübersicht

1.1 Projektziel und Beschreibung

WueKaStL (Würzburger Kaffeestrichliste) ist eine Java-basierte Webanwendung zur digitalen Verwaltung der Kaffeestrichliste und des Guthabens für Mitarbeiter. Das System ersetzt die traditionelle Papier-Strichliste und bietet eine moderne, benutzerfreundliche Lösung zur Erfassung von Getränkekonsument und zur Verwaltung von Benutzerguthaben.

1.2 Hauptfunktionalitäten

Das System bietet folgende Kernfunktionen:

- **Benutzerverwaltung:** Registrierung, Authentifizierung und Profilverwaltung
- **Getränkeverwaltung:** Verwaltung von Artikeln mit Preisen und Icons
- **Warenkorb-System:** Auswahl mehrerer Getränke vor der Buchung
- **Buchungssystem:** Automatische Erfassung von Käufen und Guthabenänderungen
- **Guthabenverwaltung:** Echtzeit-Anzeige des Benutzerguthabens
- **Verlaufsansicht:** Übersicht aller getätigten Buchungen
- **Admin-Bereich:** Umfassende Verwaltungsfunktionen für Administratoren
- **PIN-Authentifizierung:** Sichere Anmeldung mit persönlicher PIN

1.3 Technologie-Stack

Backend:

- Java 1.8 (Servlets, JSP)
- MariaDB-Datenbank 10.11 mit JDBC
- Apache Tomcat 9.0
- Apache Httpd 2.4

Frontend:

- HTML, CSS
- JavaScript mit jQuery (3.7.1)
- Responsive Design

2. Systemarchitektur

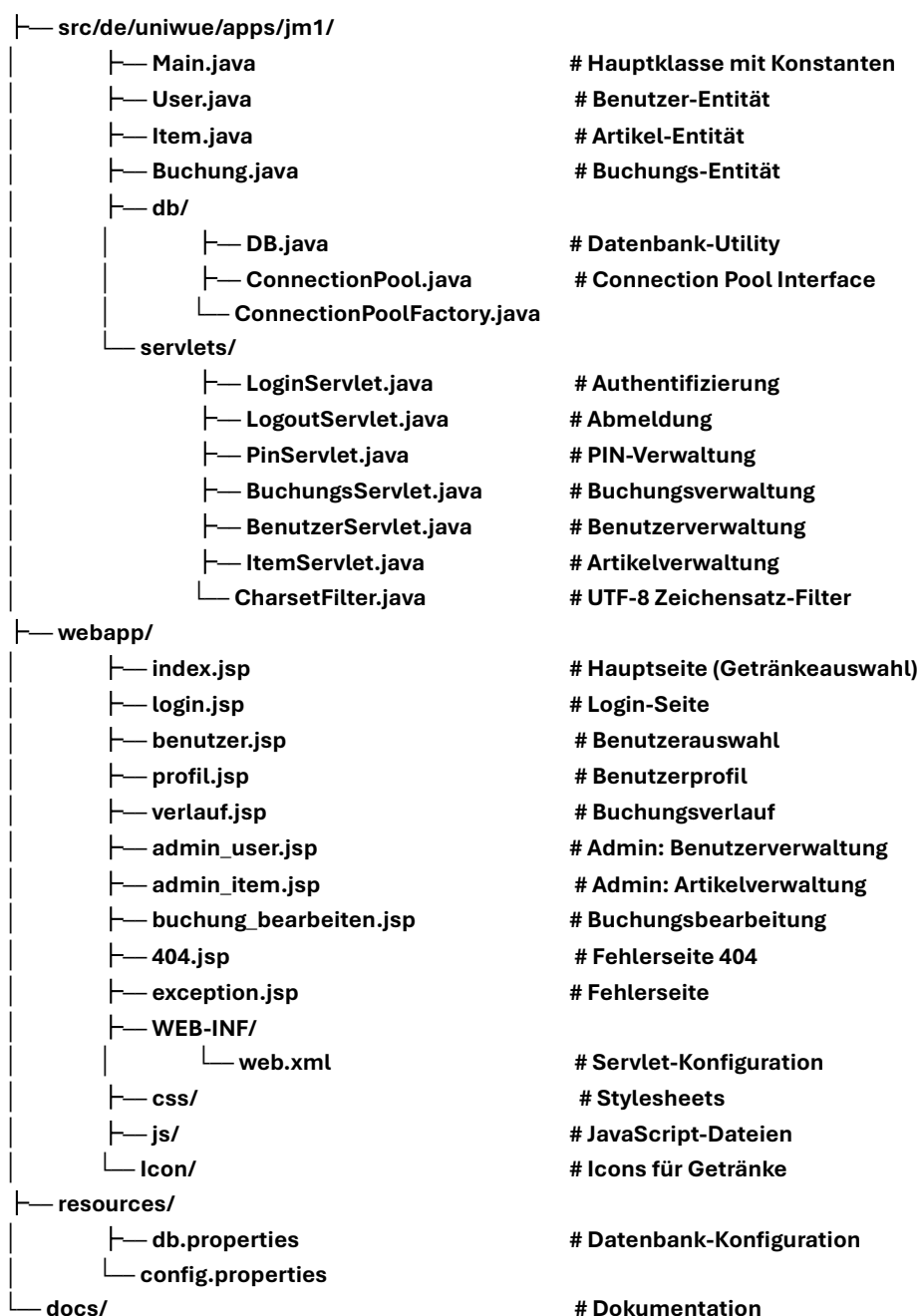
2.1 Architekturübersicht

Das Projekt folgt einer 3-Schichten-Architektur:

1. **Präsentationsschicht** (webapp/): JSP-Seiten und statische Ressourcen
2. **Geschäftslogikschicht** (src/de/uniwue/apps/jm1/servlets): Servlets
3. **Datenzugriffsschicht** (src/de/uniwue/apps/jm1/): Java Klassen

2.2 Projektstruktur

AzubiProjekt_JM-1/



2.3 Datenmodell

Haupttabellen:

1. **user:** Benutzerinformationen
 - UID (Primärschlüssel)
 - Vorname, Nachname
 - Guthaben (in Cent)
 - Passwort (SHA-256 Hash)
 - Status (z.B. "Azubi", "Mitarbeiter")
 - Admin (Boolean)
 - Aktiv (Boolean)
2. **item:** Artikel/Getränke
 - UID (Primärschlüssel)
 - Name
 - IconUrl
 - Aktiv (Boolean)
3. **itempreis:** Preisverlauf für Artikel
 - ID (Primärschlüssel)
 - ItemID (Fremdschlüssel → item)
 - Preis (in Cent)
 - Datum (Gültigkeitsdatum)
4. **buchung:** Transaktionen
 - UID (Primärschlüssel)
 - UserID (Fremdschlüssel → user)
 - Summe (in Cent)
 - Kommentar
 - Datum
 - Typ (STRICHE, AUFBUCHUNG, ABBUCHUNG)
5. **strich:** Einzelne Artikel in Buchungen
 - Item-ID (Fremdschlüssel → item)
 - Buchung-ID (Fremdschlüssel → buchung)
 - Itempreis-ID (Fremdschlüssel → itempreis)

Beziehungen:

- User 1:N Buchung (Ein Benutzer kann viele Buchungen haben)
- Buchung N:M Item (Eine Buchung kann mehrere Items enthalten)
- Item 1:N Itempreis (Ein Item kann mehrere Preise im Zeitverlauf haben)

3. Funktionsbeschreibung

3.1 Benutzer-Funktionen

3.1.1 Anmeldung und Authentifizierung

Anmeldeprozess:

1. Benutzer wählt seinen Namen aus der Liste (benutzer.jsp)
2. Bei Benutzern mit PIN: Eingabe der 4-stelligen PIN (login.jsp)
3. Bei Benutzern ohne PIN: Direkter Zugang
4. System erstellt eine Session mit User-Objekt

Sicherheit:

- Passwörter werden mit SHA-256 gehasht
- Session-basierte Authentifizierung
- Automatische Weiterleitung bei fehlender Anmeldung (bei jeder Seite)

3.1.2 Getränkeauswahl und Warenkorb

Hauptseite (index.jsp):

- Grid-Ansicht aller aktiven Getränke
- Anzeige von Name, Icon und Preis
- Klick auf Getränk fügt es zum Warenkorb hinzu
- Echtzeit-Berechnung der Gesamtsumme
- Anzeige des aktuellen Guthabens im Header

Warenkorb-Funktionalität (js/script.js):

- Hinzufügen/Entfernen von Artikeln
- Mengenänderung pro Artikel
- Dynamische Preisberechnung
- Bestätigungs-Button wird erst bei gefülltem Warenkorb aktiv

3.1.3 Buchungsvorgang

Ablauf:

1. Benutzer füllt Warenkorb
2. Klick auf "Bestätigen"
3. BuchungServlet verarbeitet die Anfrage:
 - Erstellt Buchung vom Typ "STRICHE"
 - Erstellt pro item ein strich mit der buchungs-UID
 - Speichert itempreis-UID für historische Korrektheit
 - Aktualisiert das Benutzerguthaben
4. Weiterleitung zur Hauptseite mit Erfolgsbestätigung

3.1.4 Verlauf und Profil

Verlaufsansicht (verlauf.jsp):

- Chronologische Liste aller Buchungen
- Anzeige von Datum, Summe, Typ und Artikeln
- Löschfunktion für Buchungen (bis 2 Tage alt)

Farbcodierung nach Buchungstyp:

- Grün: AUFBUCHUNG
- Rot: STRICHE/ABBUCHUNG

Profil (profil.jsp):

- Anzeige persönlicher Daten
- PIN-Verwaltung (Ändern/Entfernen)
- Guthabenübersicht

3.2 Administrator-Funktionen

3.2.1 Benutzerverwaltung (admin_user.jsp)

Funktionen:

- **Benutzer erstellen:** Neuen Benutzer mit Startguthaben anlegen
- **Benutzer bearbeiten:** Vorname, Nachname, Status ändern
- **Benutzer aktivieren/deaktivieren:** Zugang steuern
- **Guthaben aufladen:** Buchung vom Typ AUFBUCHUNG erstellen
- **Guthaben abbuchen:** Buchung vom Typ ABBUCHUNG erstellen
- **Buchungsverlauf einsehen:** Alle Buchungen eines Benutzers
- **Buchungen bearbeiten:** Nachträgliche Korrektur von Buchungen

Besonderheiten:

- Startguthaben wird als AUFBUCHUNG gebucht
- Admin-Status kann gesetzt werden
- Benutzer-Status (z.B. "Azubi", "Mitarbeiter") ist frei wählbar

3.2.2 Artikelverwaltung (admin_item.jsp)

Funktionen:

- **Artikel erstellen:** Neues item anlegen
- **Artikel bearbeiten:** Name ändern
- **Preis ändern:** Neuer Preis mit Gültigkeitsdatum
- **Artikel aktivieren/deaktivieren:** Sichtbarkeit steuern

3.2.3 Buchungsbearbeitung (buchung_bearbeiten.jsp)

Möglichkeiten:

- Summe ändern
- Buchungstyp ändern
- Kommentar ändern
- Artikel hinzufügen/entfernen
- Artikelmengen ändern

Workflow:

1. Admin wählt Buchung aus
2. Bearbeitung der Daten
3. System löscht alte Striche
4. Erstellt neue Striche mit aktuellen / historischen Werten
5. Aktualisiert Benutzerguthaben

3.3 Servlet-Architektur

Zentrale Servlets:

Servlet	URL-Pattern	Funktion
LoginServlet	/auth/anmeldung	Benutzer-Authentifizierung
LogoutServlet	/auth/abmeldung	Session-Beendigung
PinServlet	/auth/pin	PIN-Verwaltung
BuchungsServlet	/buchungen	Buchungsverwaltung (CRUD)
BenutzerServlet	/benutzer	Benutzerverwaltung
ItemServlet	/benutzer	Artikelverwaltung
CharsetFilter	/*	UTF-8 Encoding für alle Requests

3.4 Besondere Features

3.4.1 Historische Preiskorrektheit

Das System speichert zu jeder Buchung die verwendete Itempreis-ID. Dadurch bleiben alte Buchungen auch bei Preisänderungen korrekt.

3.4.2 Lösch-Zeitfenster

Buchungen können nur innerhalb von 2 Tagen nach Erstellung gelöscht werden (istLoeschbar()-Methode). Gilt nicht für Admin-Benutzer im Admin-Bereich.

3.4.3 Error Handling

- Eigene 404-Seite
- Eigene Exception-Seite
- Fehlerbehandlung in allen Servlets

4. Sicherheitsaspekte

4.1 Implementierte Sicherheitsmaßnahmen

- **Passwort-Hashing:** SHA-256 für alle Passwörter
- **Session-Management:** Automatische Weiterleitung bei ungültiger Session(auf jeder Seite)
- **SQL-Injection-Schutz:** Prepared Statements in allen Datenbankabfragen
- **Admin-Zugriff:** Prüfung des Admin-Status für geschützte Bereiche

5. Wartung und Weiterentwicklung

5.1 Bekannte Limitierungen

- Icon-Upload für Artikel nicht implementiert (aktuell: "/" als Platzhalter)
- Keine Export-Funktion für Buchungsdaten
- Keine Benutzer-Gruppenverwaltung

5.2 Mögliche Erweiterungen

- User-Gruppenverwaltung: Gruppierung von Benutzern
- Reporting: Statistiken und Auswertungen (meistgekaufte Getränke, etc.)
- Kategorien: Gruppierung von Artikeln