

# Analýza pro systém na rezervaci letenek

Karel Čemus, cvičení 9:15

## 1. Zadání

Pomocí webových služeb realizujte jednoduchý systém pro rezervaci letenek využívající 3 vrstvy serverů. V první vrstvě bude klient realizovaný jako aplikace nebo webová stránka. Druhou vrstvu bude tvořit Interface Server, který představuje jediné spojení klienta se třetí vrstvou, která je tvořena 3 servery. Booking Server se stará o práci s rezervacemi, Payment Server musí přijímat několik druhů plateb a Print Server umožňuje tisk různých dokumentů.

## 2. Funkcionální požadavky

1. Vypsát dostupné lety
2. Provést rezervaci na daný let
3. Stornovat rezervaci
4. Zaplatit rezervaci přes VISA kartu
5. Zaplatit rezervaci převodem částky z jiné stornované rezervace
6. Tisknout potvrzení o platbě, rezervaci a e-letenku

## 3. Architektura

Systém je navržen silně modulárně pro zajištění maximální testovatelnosti a minimálního opakování kódu, čímž lze dosáhnout výborné udržitelnosti a snadných aktualizací.

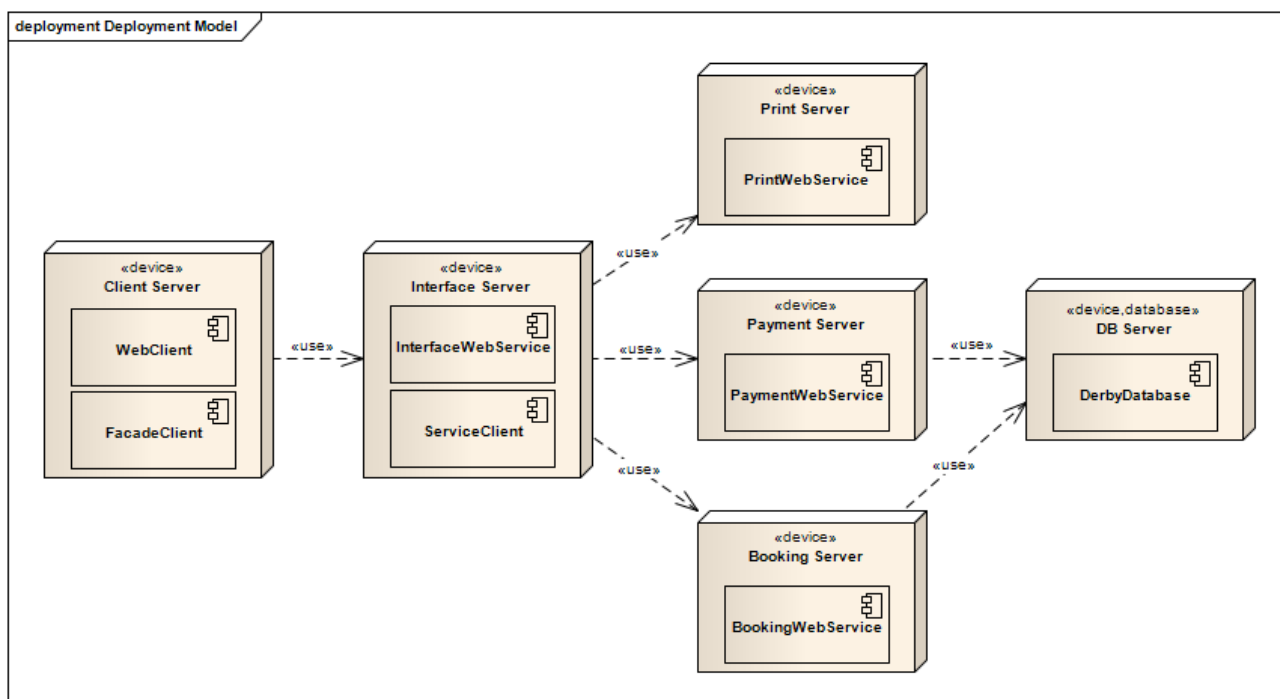


Diagram 1: Deployment diagram systému

Použité komponenty:

1. DB Server
  - persistentní úložiště dat
2. Print Server
  - umožní tisk různých dokumentů podle příchozích požadavků
  - není napojený na databázi, všechny informace dostává přes komunikační rozhraní
  - je stateless
3. Payment Server
  - přijímá požadavky na platby
  - pracuje transakčně
  - převedené peníze ukládá do databáze
4. Booking Server
  - obsahuje informace o letech a destinacích
  - přijímá požadavky pro práci s rezervacemi
  - pracuje transakčně
  - užívá persistentní úložiště
5. Service Client
  - implementuje klienta pro nabízené služby 3. vrstvy, umí volat vzdálená API
6. Interface Server
  - funguje jako fasáda pro služby 3. vrstvy
  - poskytuje API potřebné pro vlastní business aplikace
  - integruje služby 3. úrovně
  - deleguje požadavky na příslušné servery
  - obsahuje adaptér mezi objekty 2. a 3. vrstvy
7. Facade Client
  - implementuje klienta pro API 2. vrstvy, umí volat vzdálený server
8. Web Client
  - nabízí webové rozhraní pro práci se systémem

## 4. Technologie

Celý systém bude realizovaný pomocí SOAP webových služeb definovaných prostřednictvím WSDL specifikace. Pro implementaci budou použity následující technologie s přihlédnutím k reálným požadavkům zlehčeným o fakt, že se jedná o semestrální projekt.

- Databázový engine Derby, ačkoliv tato databáze má mnoho nedostatků jako špatná podpora constraints či transakcí, tak pro prezentaci funkčnosti to postačuje a díky použitému rozhraní JDBC pak při případném nasazení není komplikované provést výměnu databáze za kvalitnější engine, např. Postgres
- serverový kontejner Tomcat 7
- Hibernate 3 jako ORM provider
- poskytovatel webových služeb Apache CXF
- integrační framework Spring 3 pro DI, bean a transaction management

- JSF 2.1 pro realizaci webového klienta
- TestNG pro implementaci testovacích skriptů pro ověření dostupnosti a funkčnosti webových služeb

## **5. Testovatelnost**

Pomocí frameworku TestNG a jednotlivých implementovaných klientských modulů (FacadeClient, ServiceClient) bude vytvořena sada testovacích scénářů, které prověří chování vystavených webových služeb a ověří je proti specifikaci.