ТЕХНИЧЕСКИ УНИВЕРСИТЕТ - СОФИЯ

Катедра: „Информатика и софтуерни науки”
Дисциплина: „Софтуерни архитектури”

АРХИТЕКТУРНА ДОКУМЕНТАЦИЯ НА СОФТУЕР

Система за онлайн поръчка на таксиметров превоз

Десислава Емилова Милушева
ИСН, курс II, гр. 77,
Фак. № 471219007

Разработил: …….
/ Десислава Милушева/

Проверил: ……
/ ас. Виктор Главев /

София, 2021г.

Съдържание

1. Въведение

Този документ предоставя общ преглед и описание на архитектурата и дизайна на „Система за онлайн поръчка на таксиметров превоз”. Обяснява се как потребителят ще може да поръчва таксиметров превоз, а шофьорът на превозното средство ще може да визуализира направените поръчки. Ще бъде обърнато подробно внимание на архитектурно значимите функционални изисквания и няколко от изгледите на системата. И накрая, ще бъде направена качествена оценка на системата и нейните функционални изисквания, които ще завършат документа.

1.1. Цел

Документът за софтуерната архитектура предоставя изчерпателен архитектурен преглед на системата за онлайн поръчка на таксиметров превоз, като представя редица различни архитектурни изгледи, за да изобрази различните аспекти на системата. Тази документация е предназначен да отбележи и предаде значимите архитектурни решения, които са взети във връзка с изграждането на системата.

За да изобрази софтуера възможно най-точно, структурата на този документ се основава върху разглеждането и описанието на три основни изгледа оформени около случаите на употреба и използване на системата:

- изглед на случаи на употреба
- логически изглед
- изглед на процеса

1.2. Обхват

Приложението ще бъде разработено с помощта на архитектурния стил микроуслуги, защото този тип архитектура представлява подход за разработването на едно приложение като набор от малки услуги, всяка от които работи в свой собствен процес и комуникира с леки механизми. Тези микроуслуги са изградени около бизнес възможности и могат да бъдат внедрени независимо. Съществува минимум централизирано управление на тези микроуслуги, което може да бъде написано на различни програмни езици и да се използват различни технологии за съхранение на данните. Използвайки микроуслугите като архитектуен стил, се цели при разработката на „Системата за онлайн поръчка на таксиметров превоз”, всяка от функционалностите да се оформя в една микроуслуга, като всяка микроуслуга се разполага поотделно и може да комуникира с

останалите компоненти, което би позволило бърза разработка и лесно внедряване на нови функционалности.

2. Архитектурни цели и ограничения

Разработката на системата за онлайн поръчка на таксиметров превоз, откроява няколко ключови изисквания и системни ограничения, които имат съществено значение за архитектурата ѝ . Те са:

- архитектурата на системата трябва да може да осигурява бърза обработка на потребителски заявки, като това действие трябва да се извършва в оптимални времеви граници за не повече от 2 секунди;

- системата трябва да е способна да осигури безпроблемната обработка на огромен брой потребителски заявки, дори при многократно нарастване на броя на заявките едновременно;

- системата трябва да има възможност да преизползва определени ресурси, когато това е допустимо, с цел оптимална бързина на работата;

- системната архитектура трябва да позволява лесно преконфигуриране на отделните модули при нарастване на обема на потребителски заявки с цел запазване на определената производителност;

- архитектурата на системата трябва да позволява лесно разрастване и добавяне на нови функционалности, без това да засегне бързодействието и работата на системата;

- системата трябва да осигурява защита на обменяните данни и потребителската информация;

3. Изглед на случаи на употреба

3.1. Общ преглед

Целта на изгледа на случая на употреба е да даде допълнителен контекст около използването на системата и взаимодействията между нейните компоненти. Включва определени случаи на употреба и сценариите около тях, които представляват значима, основна функционалност на завършената система. Този изглед представя изискванията и нуждите на крайните потребители на

системата и се разглежда в детайли в последващото ниво на проектиране на системата. В този документ ще бъде разгледана диаграмата на случаите на употреба, включвайки описание на актьорите и възможните сценарии, които се свързват с тях.

Клиент
Клиентът ще има възможност да поръчва таксиметров превоз, създавайки заявка, чрез уеб сайт (Web API) или чрез мобилно приложение (Mobile App API).

Таксиметров шофьор
Таксиметровият шофьор ще приема и извършва съществуващите заявки за превоз.

Оператор
Операторът ще има възможност да създава клиентски заявка за таксиметров превоз, връщайки детайли на клиента относно извършената операция.

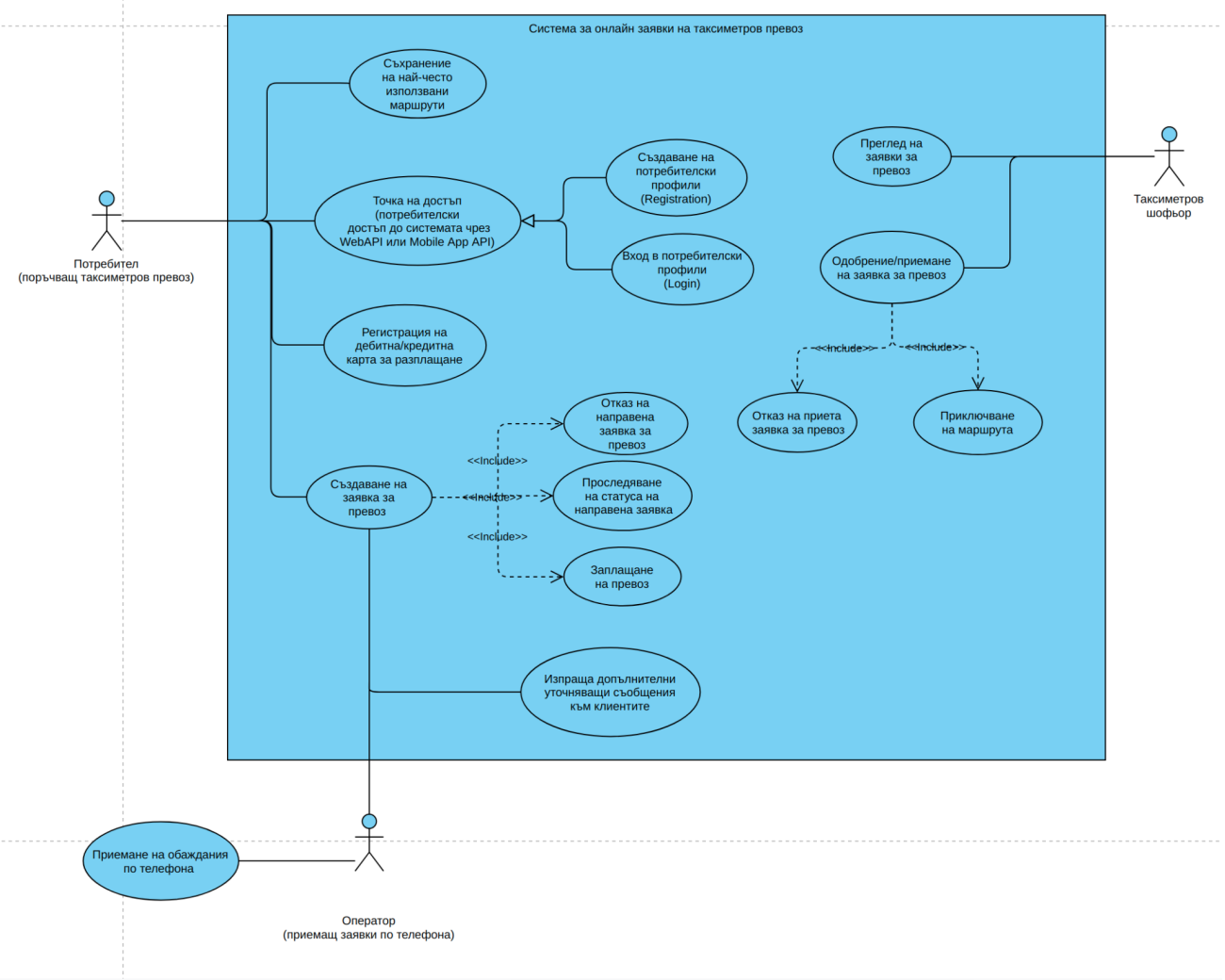
Списък с случаите на употреба, които представляват основната функционалност на крайната система:

Клиент
- Точка на достъп (потребителски достъп до системата)
- Съхраняване на най-често използвани маршрути
- Регистрация на карта за разплащане
- Създаване на заявка за превоз

Таксиметров шофьор
- Преглед на заявки за превоз
- Одобрение/приемане на заявка за превоз

Оператор
- Създаване на заявка за превоз
- Изпращане на допълнителни уточняващи съобщения

3.2. Реализации на изгледа за случаи на употреба - диаграма на случаите на употреба

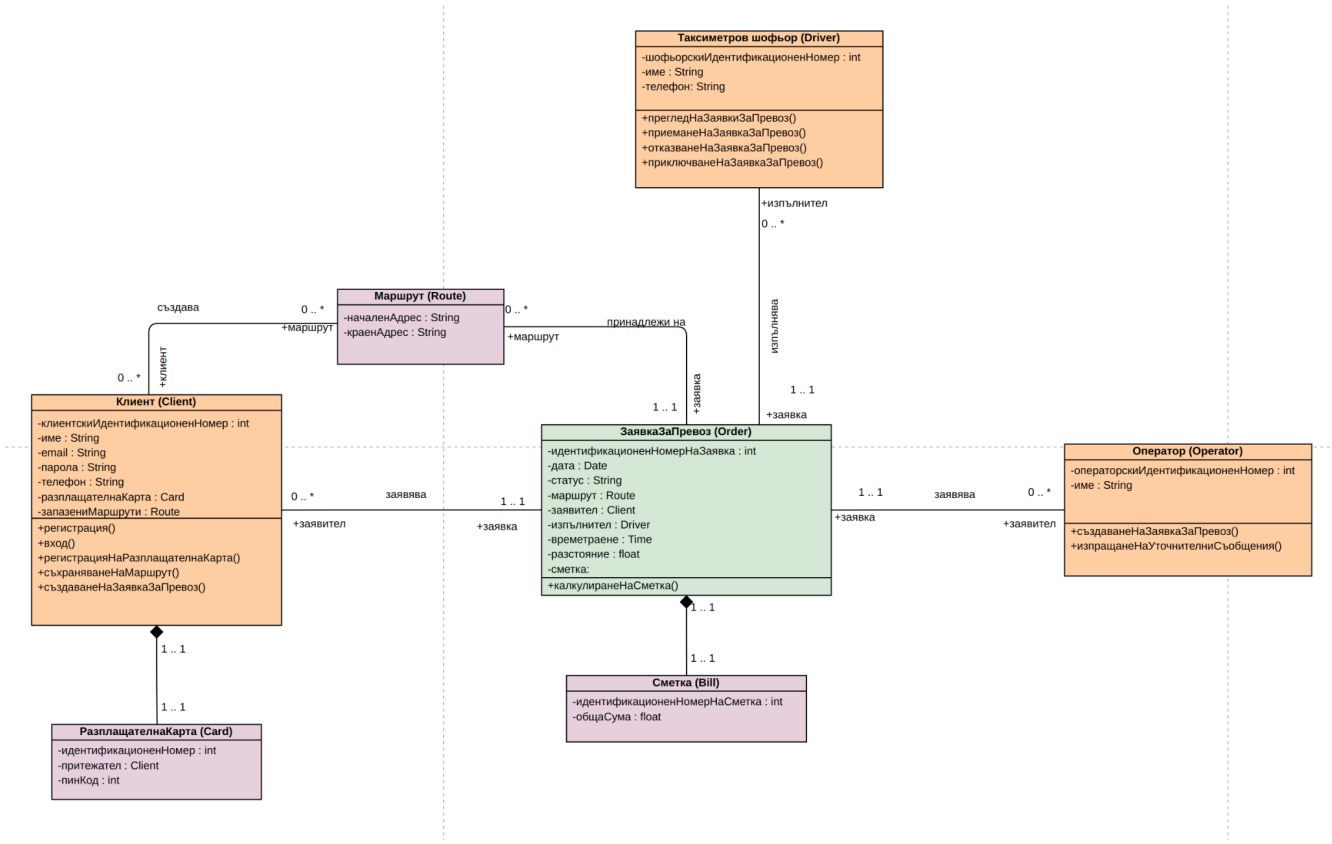


4. Логически изглед

4.1. Общ преглед

Целта на логическия изгледа е да опише функционалните изисквания, обектния структура на системата, както и да декомпозира системата на основните ѝ компоненти. Ключови елементи на модела са класовете, обектите и взаимодействие между тях. В този документ ще бъде разгледана диаграмата на класове, включвайки описание на основните класове, техните отговорности, основни операции и атрибути.

4.2. Реализации на логически изгледа - клас диаграма

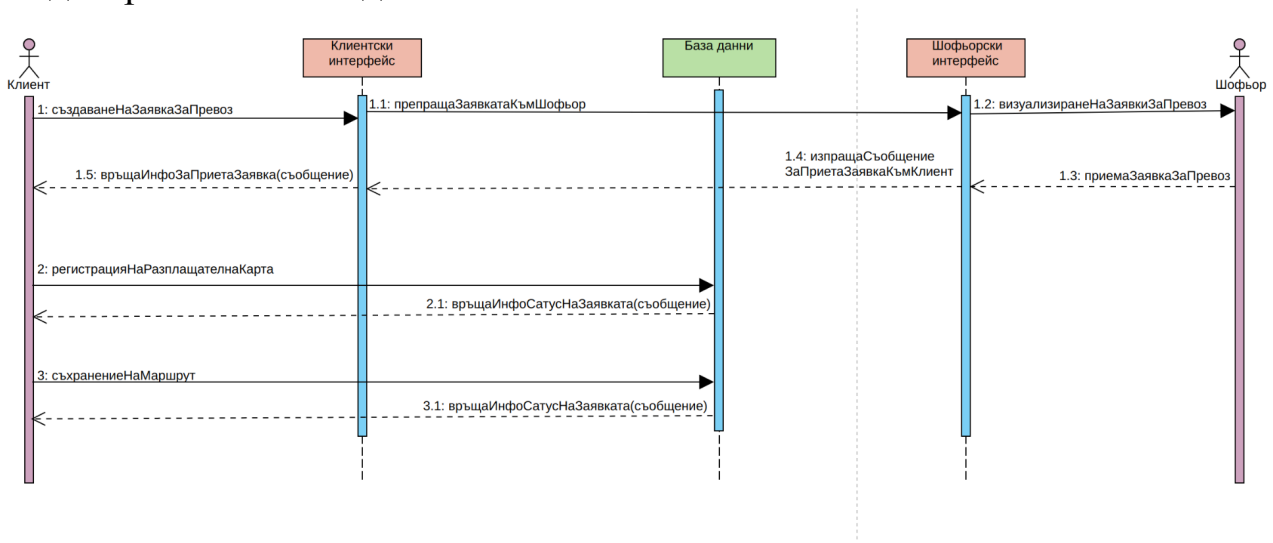


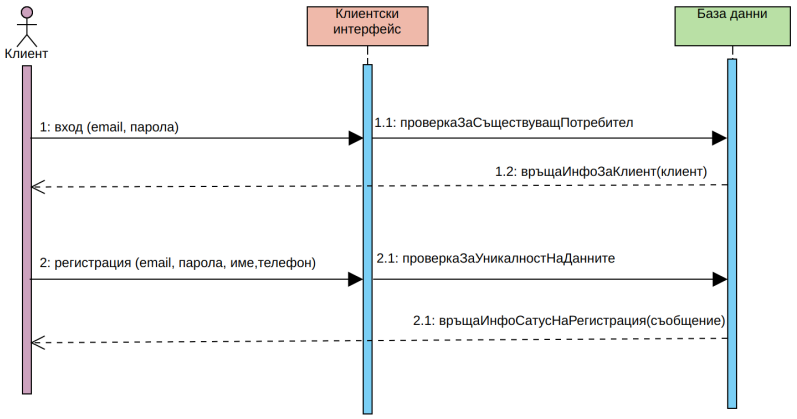
5. Изглед на процеса

5.1. Общ преглед

Целта на изгледа на процеса е да представи динамичните аспекти на системата, обяснявайки системните процеси и начина, по който те комуникират. Този изглед се фокусира върху поведението на системата по времето на нейното изпълнение. В този документ ще бъде разгледана диаграма на последователностите, включвайки описание на основните изпълними процеси от гледна точка на тяхното изпълнение във времето.

5.2. Реализации на изглед на процесите - диаграма на последователностите





6. Качество на системата

Микроуслугите като архитектурата избрана за разработката на „Система за онлайн поръчка на таксиметров превоз”, осигурява изпълнение на основните и най-важни нефункционални изисквания, които биха били търсени и необходими за системи от подобен мащаб. Те са:

Мащабируемост

Възможността за лесно мащабиране на системата, като бързо се репликират само необходимите части, е важен аспект, който избраната архитектура предоставя. В практическо отношение мащабируемостта на системата може да се дефинира в това колко потребители или на колко заявки системата ще трябва да обработи в пика на тяхното изпълнение, като например в определени части на деня системата, че трябва да се справи с повече заявки за превоз от клиенти. Още един важен аспект, който микроуслугите предоставят, може да бъде засегнат, когато системата изисква различни характеристики на различни функционалности. Например: по-вероятно е регистрацията за потребители на приложение да бъде използвана много по-рядко, отколкото например поръчването на превоз. Това е от значение, като се има предвид, че в микроуслугите могат да бъдат мащабирани по функционалност.

Производителност

В контекста на екосистемата на микроуслугите, мащабируемостта е свързана с това колко заявки може да обработи една микроуслуга. Следващият важен аспект е производителноста, която се отнася до това колко бързо се обработват и изпълняват от системата, поетите заявки. Естествената тенденция, когато се определя мащаба на растеж на една микроуслуга, е да се формулира оценъчна скалата на растеж по отношение на заявките в секунда, които услугата може да поддържа.

Ако броят на заявките, отправени към услугата, е пряко свързан с броя на хората, които отварят телефонно или уеб приложение, тогава може да бъде планирано мащабиране на услугата, като предскажем колко хора ще отворят приложението. Ако броят на заявките, отправени към услугата, се определя от броя на хората, които правят заявка за превоз, тогава услугата се мащабира със обработката на заявките за превози. Така с използване на оценъчен модел и предвид лесното и бързо мащабиране на микроуслугите, може да бъде достигнато високо ниво на необходима производителност.

Сигурност

Микроуслугите позволяват засилване на сигурността на определени компоненти, когато това е необходимо, благодарение на тяхната същност да работят като отделни единици. Например, онлайн поръчването на превоз изисква по-малко сигурност от частта за регистрация на дебитна карта за разплащане. Самите данни обвързани с клиентските сметки и карти са много по-критична информация от това кой, кога и къде е поръчал превоз. Микроуслугите ще позволят на

приложението да имат различни нива на сигурност за всеки компонент, което би могло да бъде предимство за развитието.

Възможност за поддръжка

Поддържането на софтуера се определя като лекотата, с която софтуерна система или компонент могат да бъдат модифицирани, за да коригират грешки, да подобрят производителността или други атрибути, да се адаптират към промени в средата. Най-честите проблеми в една система са тези, базирани на размера, сложността и свързването на ниво изходен код. Предвид изброените проблеми, можем да бъде заключено, че по-малките, по-малко сложни и отделени микроуслуги могат много по-лесно да бъдат поддържани в подобна голяма система. Освен това, при разработката на микроуслуги, екипите може да бъдат разделени на по-малки и по-специализирани екипи, което помага с поддържането на софтуера, като се има предвид, че някои технологии и програмни езици могат да бъдат по-подходящи за някои цели от други.