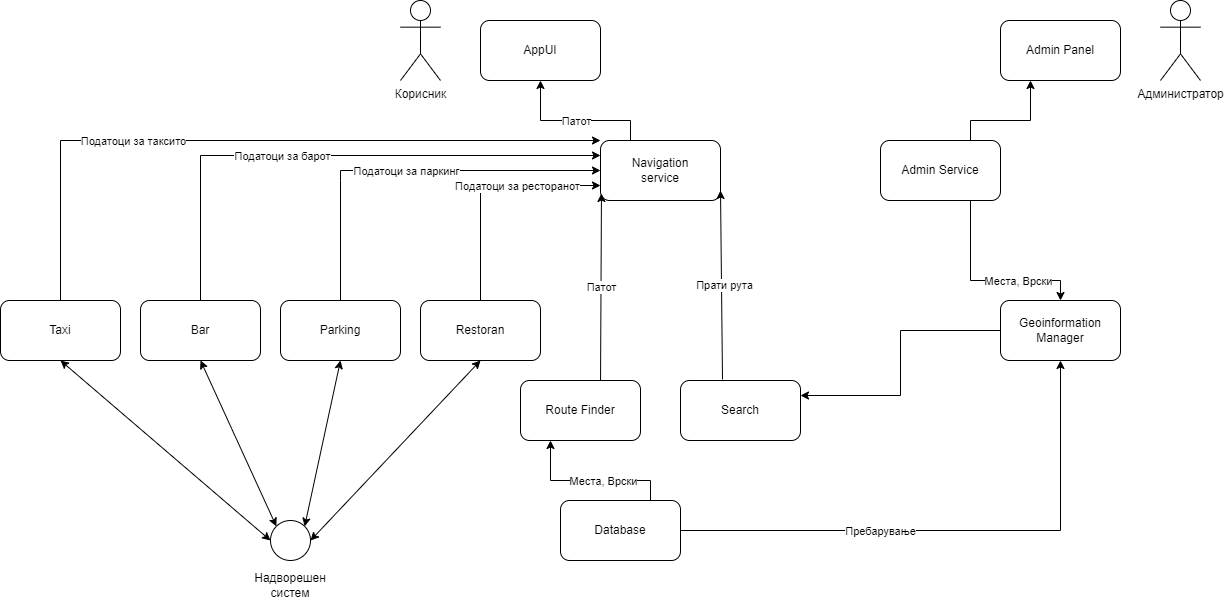
**//fali pocetna strana**

1. **Концептуална архитектура**

Концептуалната архитектура се фокусира на доменот на проблемот, притоа Земајќи ја предвид компонентата за збир на поврзани одговорности на ниво на домен ( domain-level), и одговорностите кои произлегуваат од функциските барања. Системот, користејќи ги функционалните барања како основа ги гради основните релации помеѓу ентитетите прикажувајќи едноставен поглед на функционалностите и однесувањата...

* 1. **Функционални барања**
     1. Апликацијата треба да овозможи приказ на листа со такси компании.
     2. Апликацијата треба да овозможи филтрирање на такси компаниите по: име, рејтинг.
     3. Апликацијата треба да овозможи приказ на листа со паркинзи.
     4. Апликацијата треба да овозможи филтрирање на паркинзите по: име, рејтинг, географска близина, близина до одбран објект.
     5. Апликацијата треба да овозможи приказ на листа со објекти (бар, ресторан, кафич).
     6. Апликацијата треба да овозможи филтрирање на објектите (бар, ресторан, кафич) по: име, рејтинг, географска близина.
     7. Корисниците ќе може да ја оценат услугата која ја нуди одредената такси компанија.
     8. Корисниците ќе може да ја оценат услугата која ја нуди одредениот паркинг.
     9. Корисниците ќе може да ја оценат услугата која ја нуди одредениот објект(кафич, бар или ресторан).
     10. Корисникот со улога администратор ќе може да додаде, излиста, избрише и измени податоци.
     11. Корисниците ќе може да ги прегледаат оценките кои ги оставиле за даден објект.
     12. Системот треба да е во интеракција со надворешни сервиси
         1. Системот треба да презема податоци од Google Maps преку API.
     13. Системот треба да овозможи менаџмент на корисници
         1. Регистрација на нови корисници
         2. Најава/одјава на постоечки корисници

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| Податоци | Функции | Stakeholders | Систем | Хардвер | Апстрактен концепт |
| Такси | Приказ на листа | Администратор | Надворешни сервиси |  | Географска близина |
| Паркинг | Филтрирање | Корисник | Google Maps |  | Близина до одбран објект |
| Објект (ресторан, бар, кафич) | Оценува |  | API |  |  |
| Оценка | Додаде |  |  |  |  |
| Име | Излиста |  |  |  |  |
| Рејтинг | Избрише |  |  |  |  |
|  | Измени |  |  |  |  |
|  | Прегледа оценки |  |  |  |  |
|  | Презема |  |  |  |  |
|  | Регистрација |  |  |  |  |
|  | Најава |  |  |  |  |
|  | Одјава |  |  |  |  |

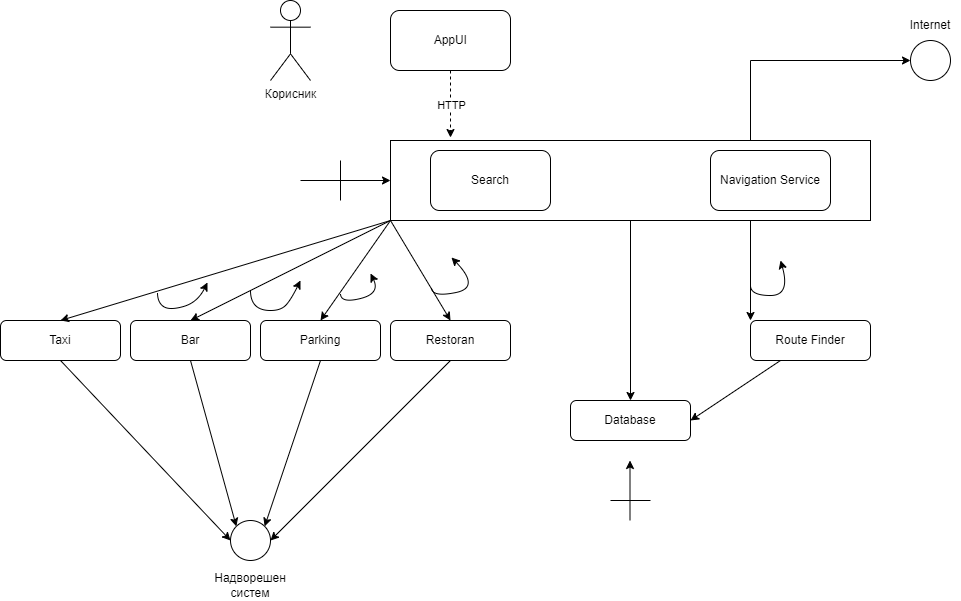


* AppUi
* приказ на географска близина
* избирање објект
* AdminPanel
* листа од такси компании
* Листа од објекти (бар, ресторан, кафич)
* Листа од паркинзи
* AdminService
* додавање нов објект
* додавање на нова такси компанија
* додавање на нов паркинг
* RouteFinder
* Ја пронаоѓа патеката до објектот/паркингот
* GeoInformation Manager  
  - Ги листа објектите и паркинзите во градот Охрид
* Navigation Service  
  - се земаат податоци потребни за избраниот објект/паркинг (име, телефонски број,работно време...).
* Search
* пребарува по име
* Пребарува по рејтинг
* Пребарува по географска близина
* Taxi
* прикажува телефонски број
* прикажува рејтинг
* Restoran
* прикажува телефонски број
* прикажува рејтинг
* Прикажува адреса
* Bar
* прикажува телефонски број
* прикажува рејтинг
* Прикажува адреса
* Parking
* прикажува адреса
* прикажува рејтинг
  1. **Нефункционални барања**
     1. Веб апликацијата треба да биде достапна за користење преку сите типови на уреди: персонален компјутер, лаптоп, мобилен уред, таблет.
     2. Апликацијата треба да биде разбирлива и лесна за употреба за секој корисник.
     3. Апликацијата треба да има кориснички интерфејс кој ќе им овозможи лесна употреба и навигација на своите корисници.
     4. Апликацијата треба да овозможи сортирање и филтрирање според одредена категорија на брз и доверлив начин.
     5. Веб апликацијата треба да е достапна за користење на следните веб прелистувачи: Firefox, Chrome, Edge, Safari.
     6. Апликацијата треба да биде скалабилна.
     7. Апликацијата треба да овозможи заштита на личните податоци на корисниците.

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| Податоци | Функции | Stakeholders | Систем | Хардвер | Апстрактен концепт |
| Лични податоци | Достапна | Корисници | Firefox | Персонален компјутер | Лесна употреба |
|  | Користење |  | Chrome | Таблет | Навигација |
|  | Сортирање |  | Edge | Лаптоп | Категорија |
|  | Филтрирање |  | Safari | Мобилен уред | Брз |
|  | Заштита |  |  |  | Доверлив |
|  |  |  |  |  | Разбирлива |
|  |  |  |  |  | Скалабилна |

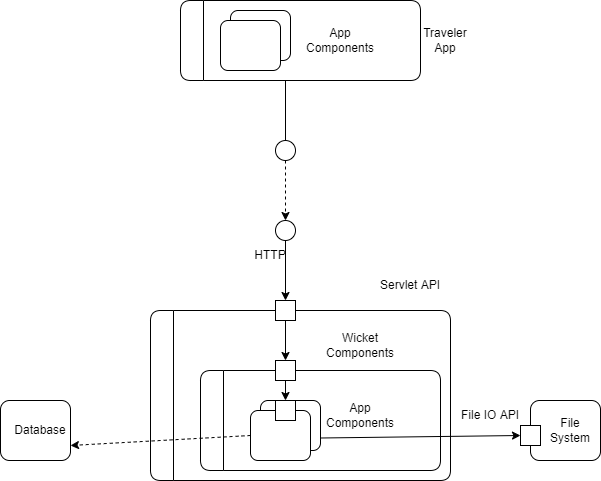
1. **Извршна архитектура**

Извршната архитектура во голема мера го одредува однесувањето и изведбата односно перформансите на системот во реално време. Фокусот од погледот на извршување се става на структурата на траење на системот, потоа хардверски елементи, потсистеми, процеси и нишки. Оваа архитектура е само мал дел од вкупната архитектура на системот.



1. **Имплементациска архитектура**

Имплементациската архитектура прикажува како е изграден еден систем односно кои технички елементи се потребни за спроведување на истиот. Оваа архитектура се однесува на non-runtime барања и атрибутите за квалитет. Истата е составена од компоненти и приклучоци кои се однесуваат на софтверските ентитети и нивните врски на ниво на изворен и бинарен код, каде што секој модел е фоксусиран на еден подсистем или процес.

****