01.12.2021

Опис на документот

Концептуален поглед на апликацијата

ДОМАШНА РАБОТА БР.2

*Дизајн и архитектура на софтвер*

*Професор: Д-р Петре Ламески*

*Од членовите на тимот*

Концептуалната архитектура ja разгледува структурата на системот од аспект на неговите функционалности на ниво на домен. За таа цел, при процесот на дизајнирање на концептуалната архитектура ги обележуваме клучните концепти во секое од барањата за подоцна истите да се класифицираат во соодветни категории (податоци, системи, засегнати страни, функции, апстрактни концепти). Иницијално се анализираат функциските, а потоа и нефункциските барања. Поврзаноста на издвоените клучни концепти подоцна е опфатена во крајниот модел кој всушност ја претставува концептуалната архитектура на апликацијата.

**1.1. Функциски барања**

1. Системот треба да е поврзан со **база на податоци**. *Приоритет1*

2. Системот треба да овозможи **пребарување** на **медицински установи** по **град** во кој се наоѓаат. *Приоритет1*

3. Системот треба да овозможи пребарување на медицински установи по **категорија** во која спаѓаат. *Приоритет1*

4. Системот треба да овозможи пребарување на медицински установи по **име на установата.** *Приоритет2*

5. На барање на **корисникот**, системот треба да **прикаже** **листа** од сите медицински установи во даден град. *Приоритет1*

6. На барање на корисникот, системот треба да **прикаже** листа од сите медициснки установи кои спаѓаат во дадена категорија. *Приоритет1*

7. На барање на корисникот, системот треба да **генерира** листа од сите медицински установи со внесеното име. *Приоритет1*

8. На барање на корисникот, системот да овозможи **избор** на установа, од понудената листа на установи.*Приоритет1*

9. При избор на дадена установа од страна на корисникот, системот треба да прикаже **локација** на установата на **мапа**. *Приоритет1*

При избор на дадена установа од страна на корисникот, системот треба да ја прикаже **локација** на установата на **мапа**. *Приоритет1*

10. При избор на дадена установа од страна на корисникот, системот треба да прикаже **информации за медицинската установа**. *Приоритет1*

11. При избор на дадена установа од страна на корисникот, системот треба да овозможи **закажување** **термин** во истата, доколку постои таква опција. *Приоритет1*

12. Системот треба да ја **преземе** локацијата на корисникот, со негова **дозвола**. *Приоритет1*

13. Системот треба да прикаже дали избраната медицинска установа нуди **тестирање** за **Ковид-19**. *Приоритет1*

14. На **администраторот** треба да му се овозможи функционалност, така што ќе може да ги **ажурира** податоците за вршење на Ковид-19 и **антигенски тестирања** во рамките на медицинските установи. *Приоритет1*

14. На барање на корисникот, системот треба да овозможи **приказ** на **најблиските** медицински установи во кои се врши Ковид-19 тестирање. *Приоритет1*

15. Системот треба да прикажува **слика** за секоја медицинска установа. *Приоритет1*

16. Системот треба да содржи информации за **членовите** од тимот. *Приоритет3*

17. Системот треба да овозможи **пребарување** на медицински установи според Ковид-19 тестирање. *Приоритет1*

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Податоци | Функции | Засегнати страни | Системи | Апстрактни концепти |
| База на податоци | Прикажува | Корисник | Надворешен систем | Локација |
| Медицински установи | Генерира | Администратор |  | Приказ |
| Град | Пребарува |  |  | Дозвола |
| Категорија на установи | Презема |  |  | Најблиска (установа) |
| Име на установа | Закажува |  |  | Избор |
| Листа | Ажурира |  |  |  |
| Мапа |  |  |  |  |
| Ковид-19 тестирања |  |  |  |  |
| Антигенски тестирања |  |  |  |  |
| Слика |  |  |  |  |
| Членови |  |  |  |  |
| Информации за установа |  |  |  |  |
| Термин |  |  |  |  |

**1.2. Нефункциски барања**

1. Просечното **време за одговор** на системот треба да биде 3 секунди. *Приоритет1*

2. Системот треба да е **достапен** за употреба 24/7 во 99.9% од времето. *Приоритет1*

3. Системот треба да работи на **уред** кој е поврзан на Интернет.*Приоритет1*

4**. Стапката на дефект** на системот треба да биде помала од 1 дефект на 1500 часа работа. *Приоритет1*

5. **Корисници** без претходна обука треба да можат да го користат системот. *Приоритет1*

6. Системот треба да му овозможи на корисникот да изврши **пребарување** за помалку од една минута. *Приоритет1*

7. Средното **време на справување со грешка** не треба да изнесува повеќе од 5 минути. *Приоритет1*

8. Системот мора да се **опорави** од пад за помалку од 1 час во 95% од случаевите. *Приоритет1*

9. Процесот за **рестартирање** на системот мора да се изврши целосно за помалку од 60 секунди. *Приоритет1*

10. Инсталирањето на **нова верзија** треба да ги остави сите содржини на **базата на податоци** **непроменети**. *Приоритет1*

11. Системот мора да **поддржи** максимум 1000 корисници **одеднаш**. *Приоритет1*

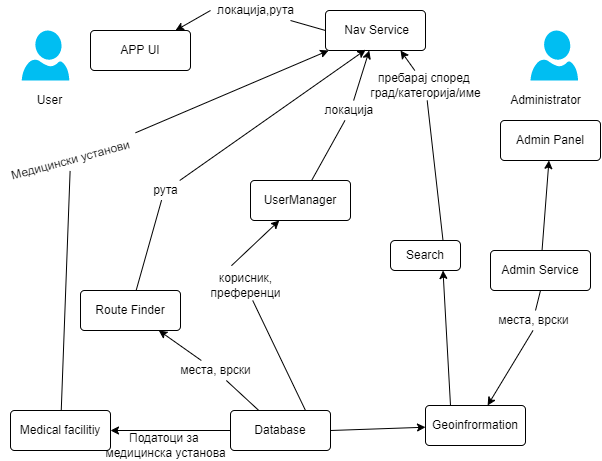
12. Системот треба да се базира на законите за **заштита** на **личните податоци.** *Приоритет1*

13. Системот треба да има интегриранo заштита од **DDos** **напади.** *Приоритет1*

14. Системот мора да биде **веб-апликација**. *Приоритет1*

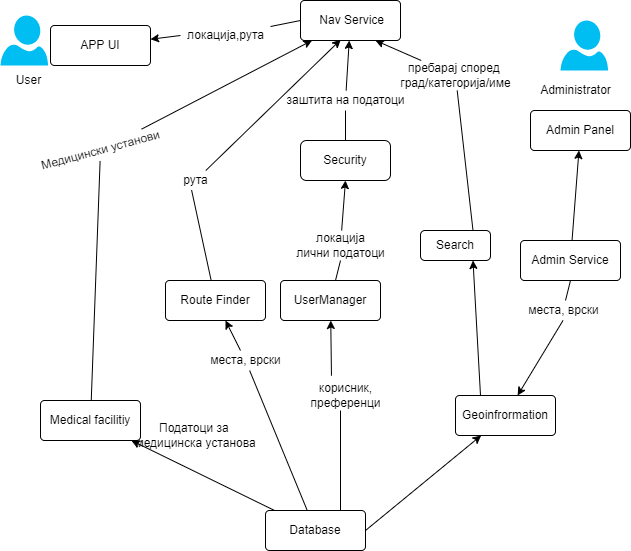
*15.* Системот мора да биде **достапен** за употреба за време на надградувањето на истиот*. Прироритет2*

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Податоци | Функции | Засегнати страни | Системи | Апстрактни концепти |
| База на податоци | Пребарување | Корисник | Надворешен систем  Phantom Buster | Време на одговор |
| Лични податоци | Опоравува |  |  | Достапност |
| Уред | Рестартирање |  |  | Стапка на дефект |
|  | Поддржува |  |  | Време на справување со грешка |
|  | Заштита |  |  | Нова верзија |
|  |  |  |  | DDoS напад |

**I Итерација:**

**II Итерација:**

\*Додаден концепти од нефунцкиски барања



**АppUI**

-Прикажи локација на медицинската установа

-Исцртај ја најблиска рута до медицинската установа

**Admin Panel**

-Излистај медицински установи

**Admin Service**

-Ажурирај податоци

-Додади медицинска установа

-Избриши медицинска установа

**Navigation Service**

-Земи податоци од медицинската установа

-Земи рута до медицинската установа

**Route Finder**

-Генерирај најблиска рута до соодветната установа

**Medical Facility**

-Излистај ги податоците за медицинската установа

**Geoinformation**

-Побарај податоци

**UserManager**

-Прибери податоци за корисник за креирање на корисничка сметка

-Прибери локација на корисник со цел изнаоѓање на најблиска рута до соодветната установа

**Search**

-Излистај ги медицинските установи соодветни на внесениот град

- Излистај ги медицинските установи соодветни на внесената категорија

- Пребарај медицинска установа според име

**Security**

-Заштити ги локацијата и личните податоци на корисникот од било каков напад