За да го прикажеме концептуалниот поглед на системот кој се однесува на domain-level functionality потребно е да ги извлечеме т.н. главни концепти (key concepts) од функционалните и нефункционалните барања и потоа со нивно анализирање можеме да дојдеме до основите за тоа како системот ќе функционира

-Функционални барања

1. Овозможување на **приказ на локалите** на **мапа** или **листа.**
2. **Селекција на локал** според **град (опфатени се градовите во РСМ).**
3. **Селекција на локал** според тип на локал (**клуб** или **кафуле**).
4. **Приказ на листа од 5те најбарани објекти** што спаѓаат во категоријата која што ја селектирал корисникот со претходна **калкулација** од страна на системот.
5. Приказ на детали за селектираниот објект како што се: **име, телефонски број, адреса, електронска пошта** и сл.
6. Брз и лесен пристап.

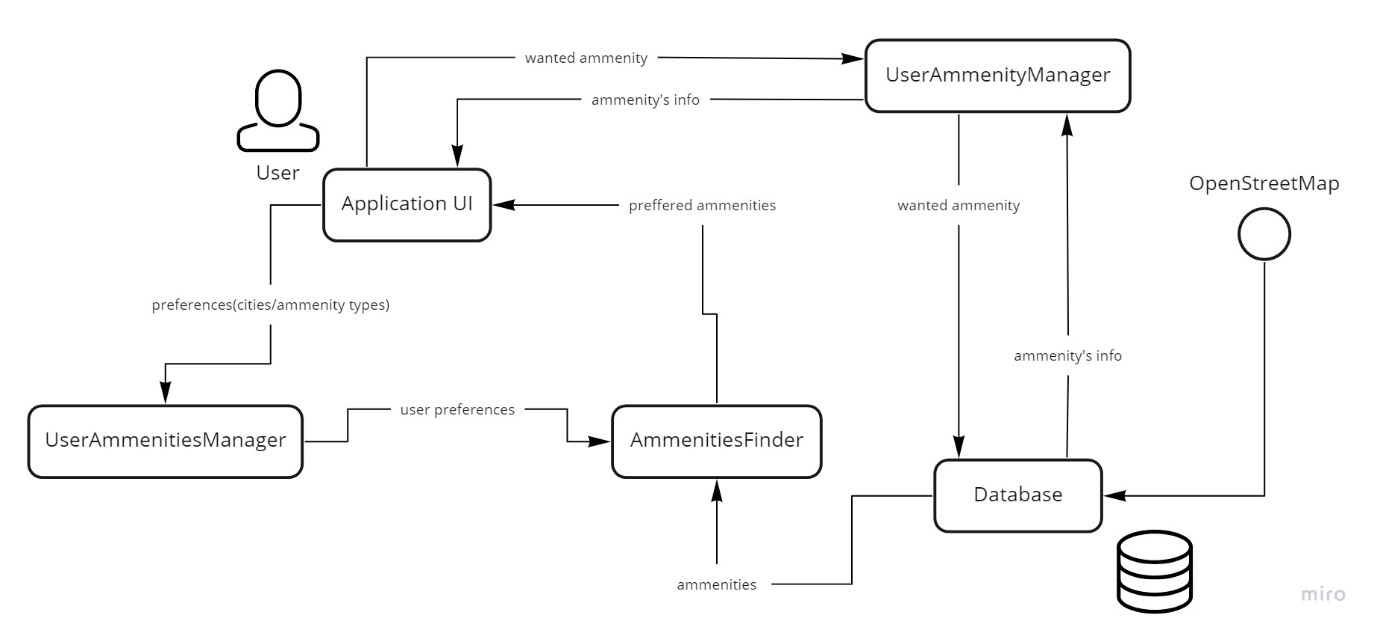
-Нефункционални барања

1. Веб апликацијата треба да биде достапна во реално време и податоците да бидат навремено **ажурирани.**
2. Апликацијата треба да биде достапна од секој пребарувач.
3. Апликацијата треба да биде достапна од секој уред.
4. Овозможување на **кориснички интерфејс** кој ќе им овозможи на корисниците брз и лесен пристап до функционалностите на апликацијата.
5. **Преземањето на информациите од база** треба да е од времетраење не поголемо од 1 секунда.
6. **Корисниците** нема да имаат пристап да ги променуваат податоците во врска со локалите. Корисниците ќе можат само да ги гледаат информациите.
7. **Менаџерите на апликацијата** ќе имаат можност да ги променуваат податоците и исто така одговараат за точноста на информациите.
8. Отворањето на страницата треба да биде реализирано во рок од неколку секунди.

-Key Concepts:

1. Приказ на локал на мапа – функција
2. Приказ на локал на листа – функција
3. Локал – податок
4. Мапа – абстрактен концепт
5. Листа – абстрактен концепт
6. Селекција според град (филтрирање) – функција
7. Селекција според тип на локал (филтрирање) – функција
8. Град (место) – податок
9. Тип на локал (клуб или кафуле) – податок
10. Калкулација – функција
11. Приказ на 5те најбарани локали – функција
12. 5 најбарани локали – абстрактен концепт
13. Име на локал – податок
14. Телефонски број – податок
15. Адреса – податок
16. Електонска пошта – податок
17. Ажурирање – функција
18. Кориснички интерфејс – абстрактен концепт
19. Преземање на информации од база – функција
20. Корисник – засегната страна
21. Менаџер на апликација – засегната страна

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Податоци | Функции | Засегнати страни | Систем | Абстрактен концепт |
| локал | приказ на мапа | корисник | OpenStreetMap | мапа |
| град | приказ на листа | менаџер |  | листа |
| тип на локал | филтрирање(град) |  |  | 5 најбарани локали |
| име на локал | филтрирање(тип на локал) |  |  | кориснички интерфејс |
| телефонски број | калкулација |  |  |  |
| адреса | приказ на 5 најбарани локали |  |  |  |
| електронска пошта | ажурирање |  |  |  |
|  | преземање податоци од база |  |  |  |



* Application UI
* Кориснички интерфејс за приказ на мапата/листата со локали и нивните информации;
* UserAmmenitiesManager
* Сервис кој ги прима побаруваните параметри за филтрирање (град и тип на локал) од корисничкиот интерфејс и ги испраќа кон филтерот AmmenitiesFinder;
* Database
* База каде што се чуваат локалите и информациите за нив
* AmmenitiesFinder
* Филтер кој ги прима податоците од базата и доколку се поставени параметри за филтрирање ги филтрира податоците и ги испраќа кон корисничкиот интерфејс. Доколку нема параметри ги испраќа сите податоци за приказ;
* UserAmmenityManager
* Сервис кој како аргумент го прима посакуваниот локал и за истиот ги презема информациите(адреса, телефонски бр и сл.) од база па го враќа кон корисничкиот интерфејс. Покрај тоа при секој повик на локал од база го зголемува бројчаникот (број на пристап до информациите) на локалот со што подоцна би се искористило ова за приказ на листата со 5те најбарани локали;