**Архитектурни стилови и дизајн**

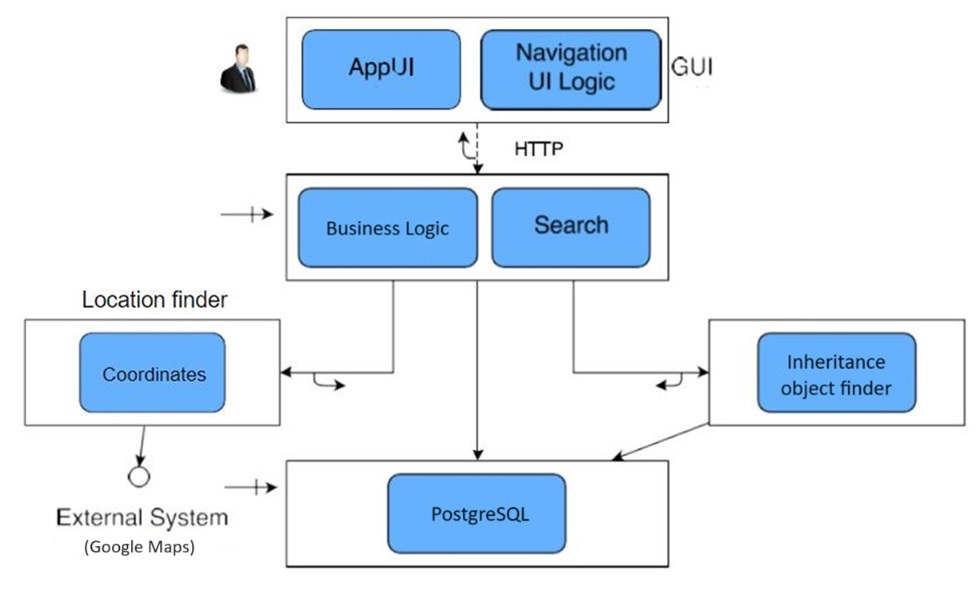
**Концепциска архитектура**

Концепциска архитектура пред се ни обезбедува изглед на првичниот архитектурен дизајн, истовремено определувајќи ги одговорностите на доменско ниво што произлегуваат од функционалните барања од засегнатите страни.

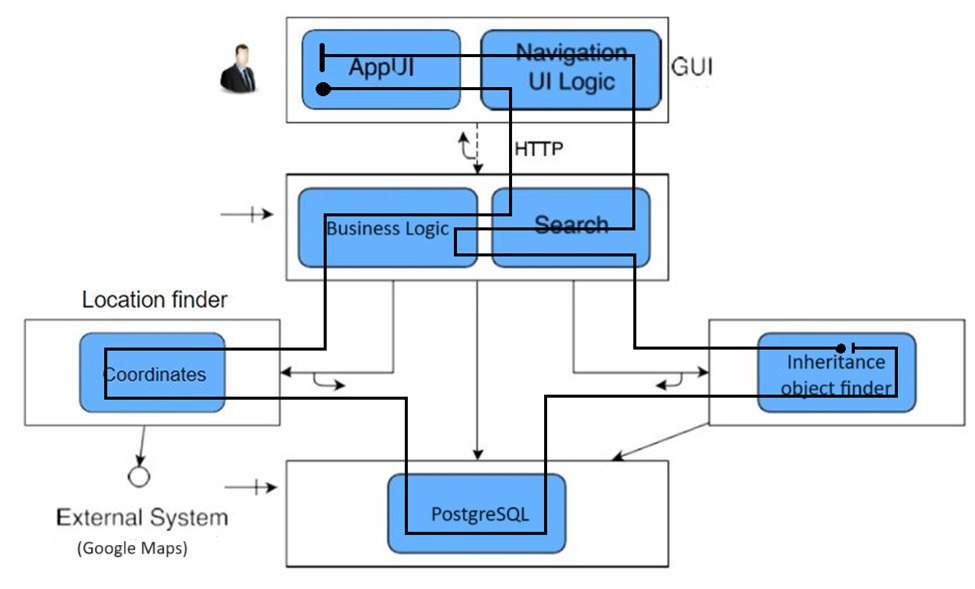
Во почеток, направивме глобална репрезентација на апликацијата, прегледувајќи ги функционалните и нефункционалните барања. Во овој првичен чекор, од функционалните и нефункционалните барања ги извлековме најзначајните концепти и термини и истите ги прикажавме во оваа табела.

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Data | Function | Stakeholder | System | Abs. concept |
| Археолошко наоѓалиште | прикажување | корисник | Веб пребарувач | Локација |
| Тврдина | пребарување |  | Веб сервер | Алатка за помош и упатство |
| Кула | филтрирање |  |  |  |
| Црква | Додавање/ бришење/ ажурирање на места |  |  |  |
| Манастир |  |  |  |  |
| Музеј |  |  |  |  |
| Галерија |  |  |  |  |
| Споменик |  |  |  |  |
| Градба |  |  |  |  |

На оваа слика е претставен погледот од концепциската архитектура. Претставено е дека AppUi со Navigation UI Logic кој се поврзува со Search и бизнис логиката кога се бара локација на објект од културно-историско наследство, притоа се пристапува до дата базата и се наоѓаат координатите на овој објект, а потоа се враќа тој поглед.

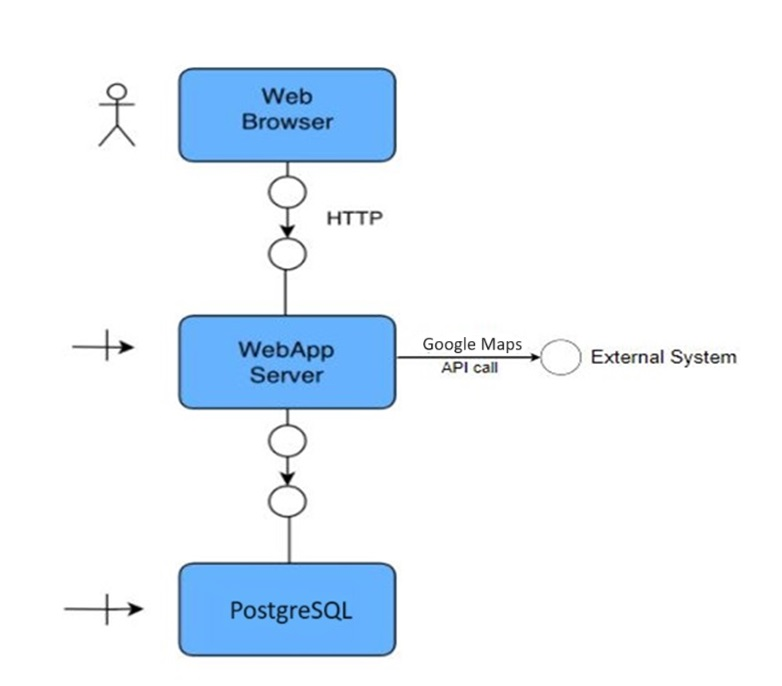


На наредната слика е прикажано однесувањето со движење низ системот, односно бихевиорален модел.

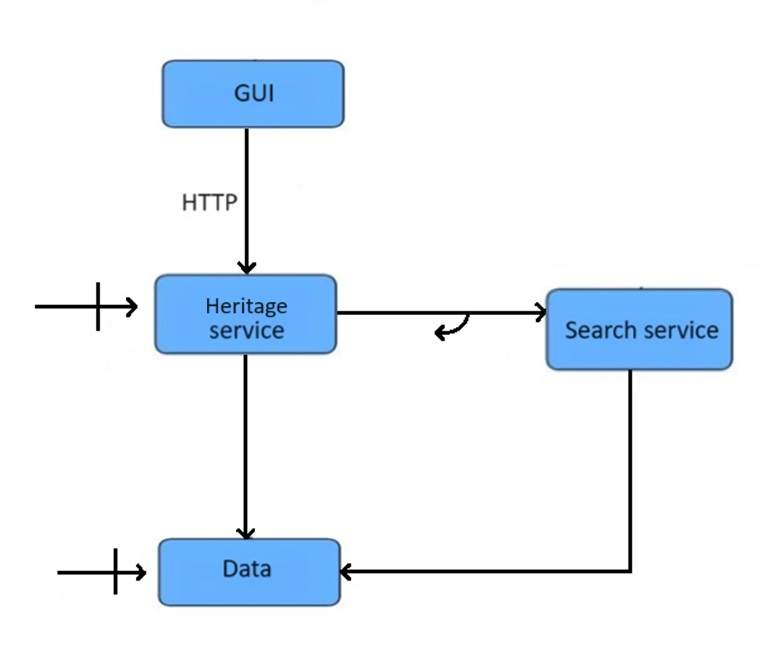


**Извршна архитектура**

Извршната архитектура се фокусира на тоа како компонентите на системот комуницираат и работат заедно во текот на извршувањето на апликацијата. На следните 2 слики се претставени дијаграми за тоа како изгледа погледот на извршната архитектура на нашиот систем.



Ова е другата слика.



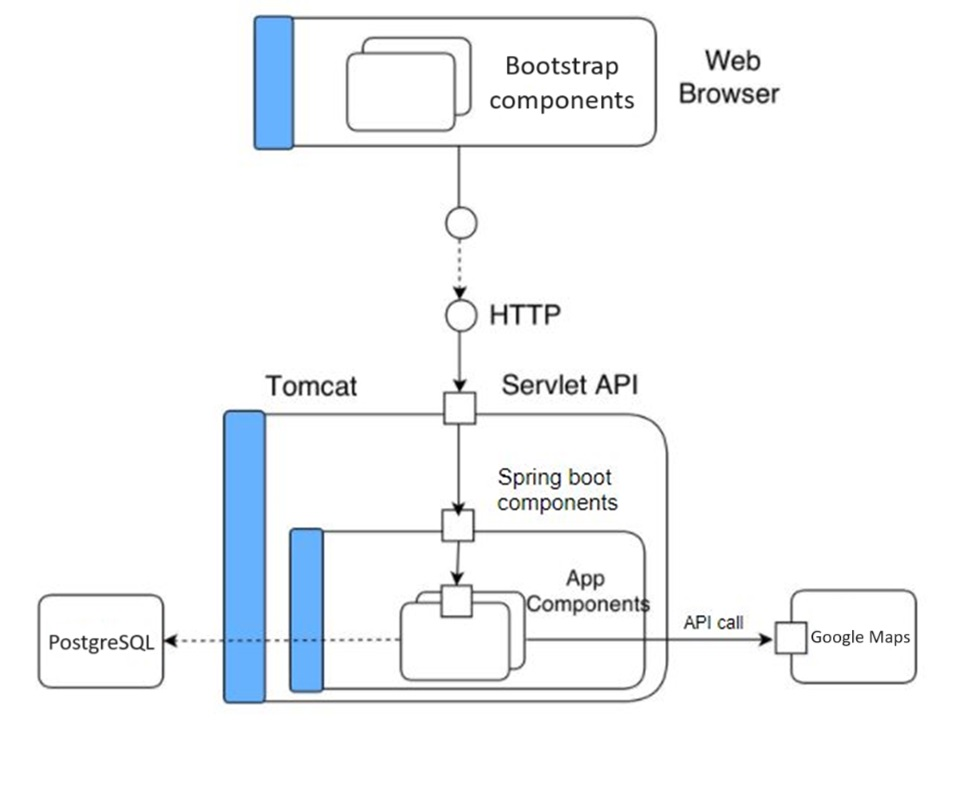
**Имплементациска архитектура**

Во изградбата на системот, имплементациската архитектура претставува секвенцата на технички елементи и нивното взаемно дејствување. Кога корисникот ќе ја користи апликацијата преку веб-прелистувач (Web Browser), тој испраќа HTTP барање до серверот. Серверот, преку користење на Tomcat, го пресретнува пристигнатото барање и го предава до соодветниот контролер.

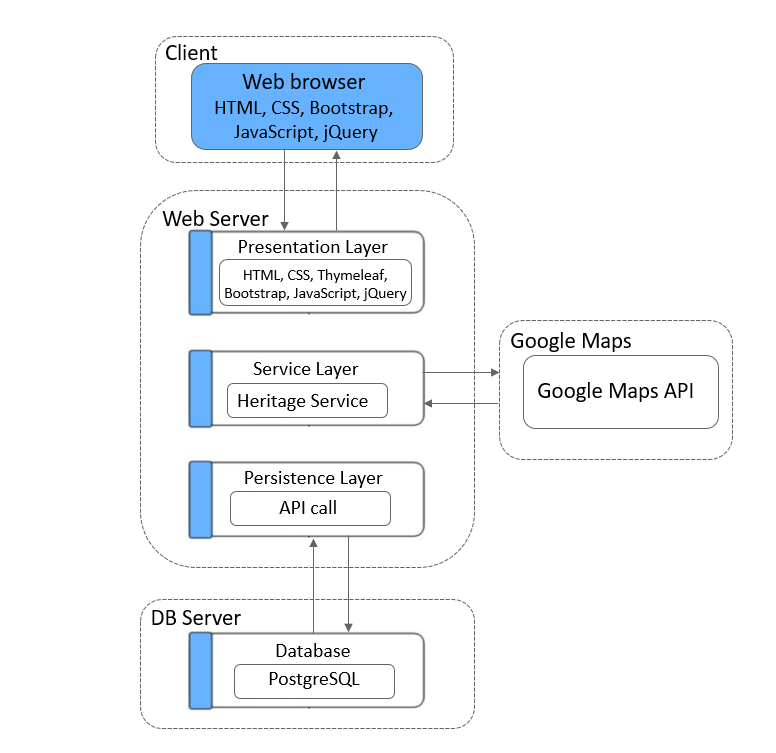
Барањето потоа се обработува во Spring Boot контејнерот. Во овој контејнер, апликациските компоненти извршуваат API повици до базата на податоци (PostgreSQL) и до Google Maps за да добијат релевантни информации за локациите на објекти од културно-историско наследство. Според добиените податоци, серверот создава одговор што се праќа преку контролер до веб-прелистувачот.

Во веб-прелистувачот, како дел од HTML одговорот се прикажува и интерактивна мапа. На мапата се обележани локациите на бараните објекти. Паралелно, апликацијата прикажува и дополнителни информации и филтри за пребарување, кои корисникот може да ги користи за персонализација на приказот на објектите.

Оваа архитектура обезбедува ефикасна комуникација меѓу компонентите и интерактивно искуство за корисниците, каде што лесно можат да пребаруваат и идентификуваат објектите на мапата.



Поспецифичен поглед кон имплементациската архитектура.



Овој секвенцијален дијаграм го креиравме со цел да го опишеме процесот на наоѓање на одредена локација на објект од културно-историско наследство и како ова се одвива низ системот.

