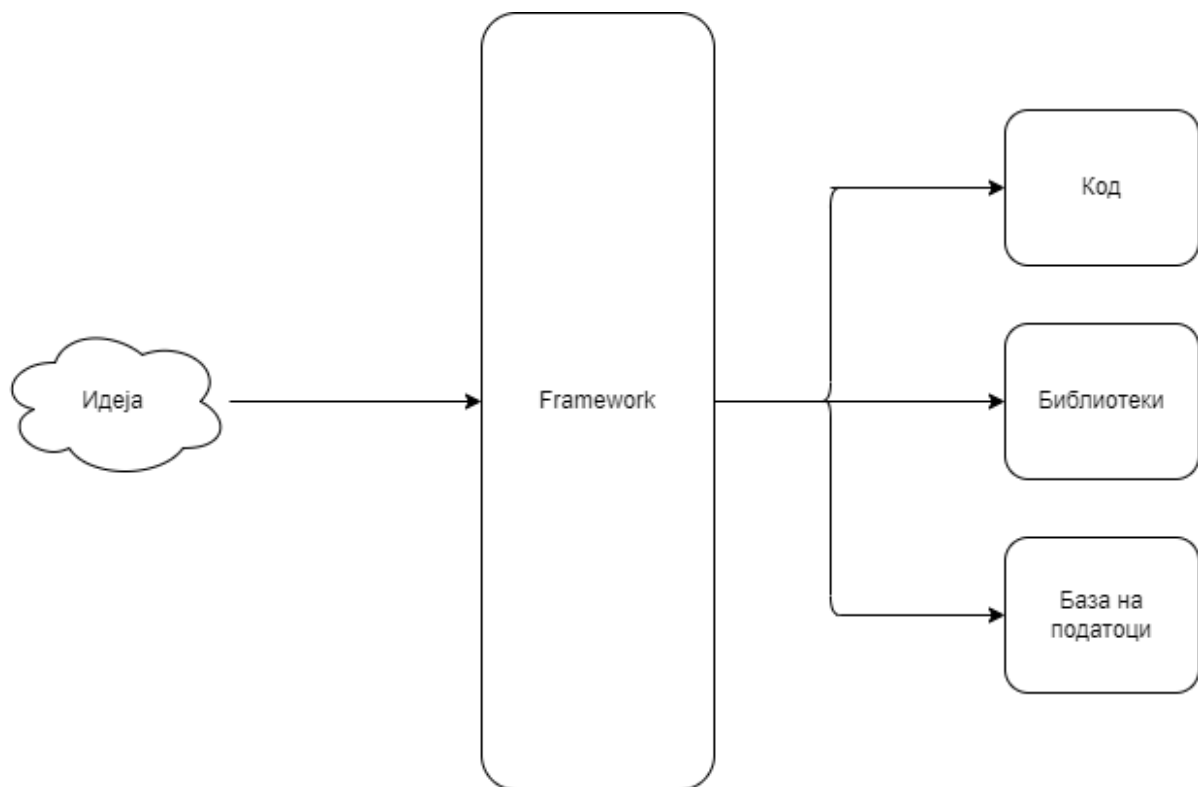


Имплементациска архитектура

Имплементациската архитектура се состои од компонентите и конекторите на системот, како различните елементи се поврзани меѓу себе, односно како е изграден системот.

Компонентите и конекторите овде ги дефинираат елементите на ниво на код.

Идејата која ја дефинираме за овој проект сакаме да ја имплементираме функционална веб апликација, поточно користејќи framework да ги дефинираме класите, соодветните функции, библиотеки, пакети и базата во која се чуваат податоците. Покрај имплементирање на идејата, внимаваме да ги да останат опфатени сите функционални и нефункционални барања.



Слика 1:

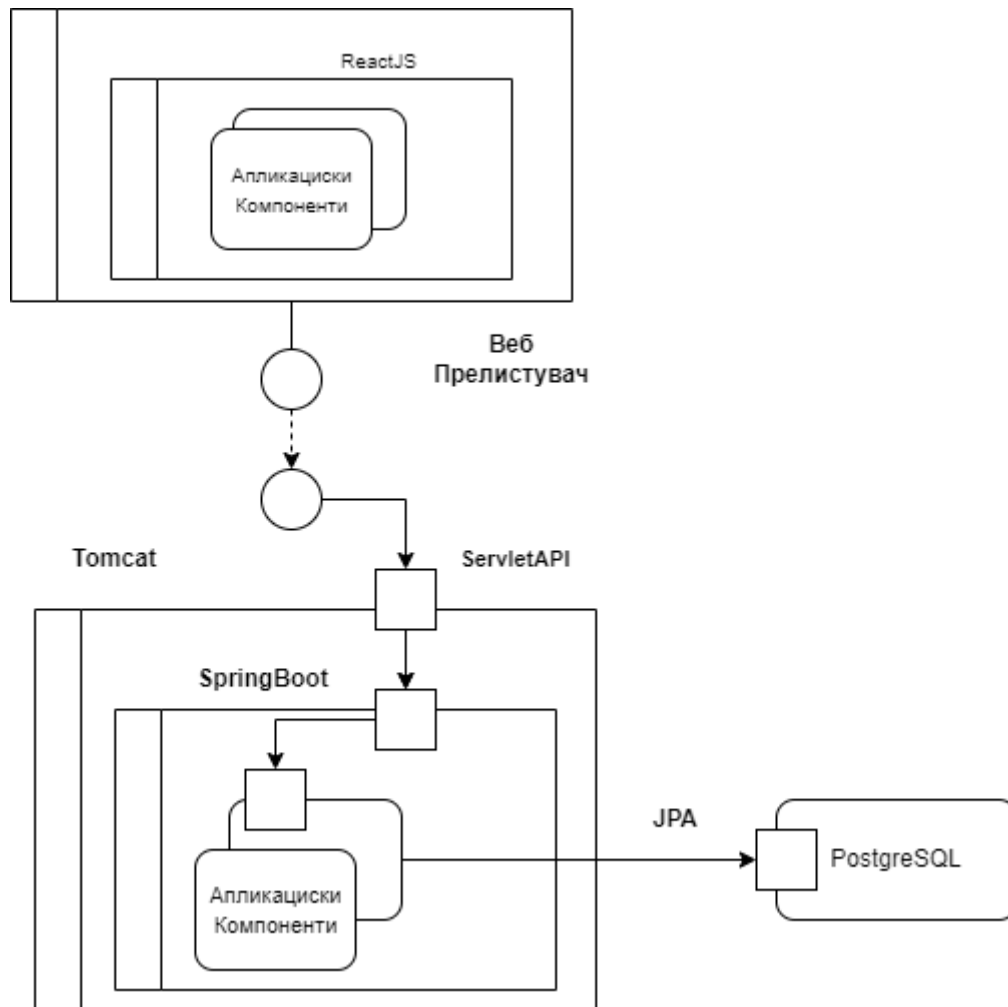
Нашата веб апликација е заснована на клиент – сервер архитектурата и се користи HTTP протоколот. За пристап и употреба на апликацијата корисникот ќе користи веб прелистувач.

Два типа на компоненти во имплементациската архитектура се апликациските и инфраструктурните компоненти.

Одговорност на апликациските компоненти е имплементацијата на одговорностите на ниво на домен. Во нашата веб апликација, апликациска компонента е source кодот.

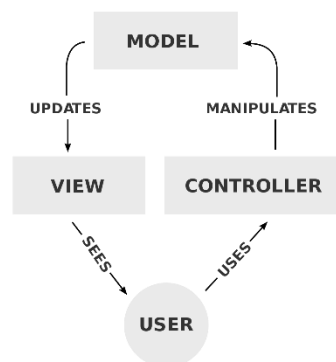
Во поглед на инфраструктурните компоненти HTTP Connection Handler – от и веб прелистувачот се неопходни затоа што нашата апликација е веб апликација. Преку веб прелистувачот се праќа корисничкото барање и се рендерира одговорот.

Како инфраструктурна компонента ние го одбравме Java Springboot framework – от, кој е наменет за изработка на динамички веб апликации.

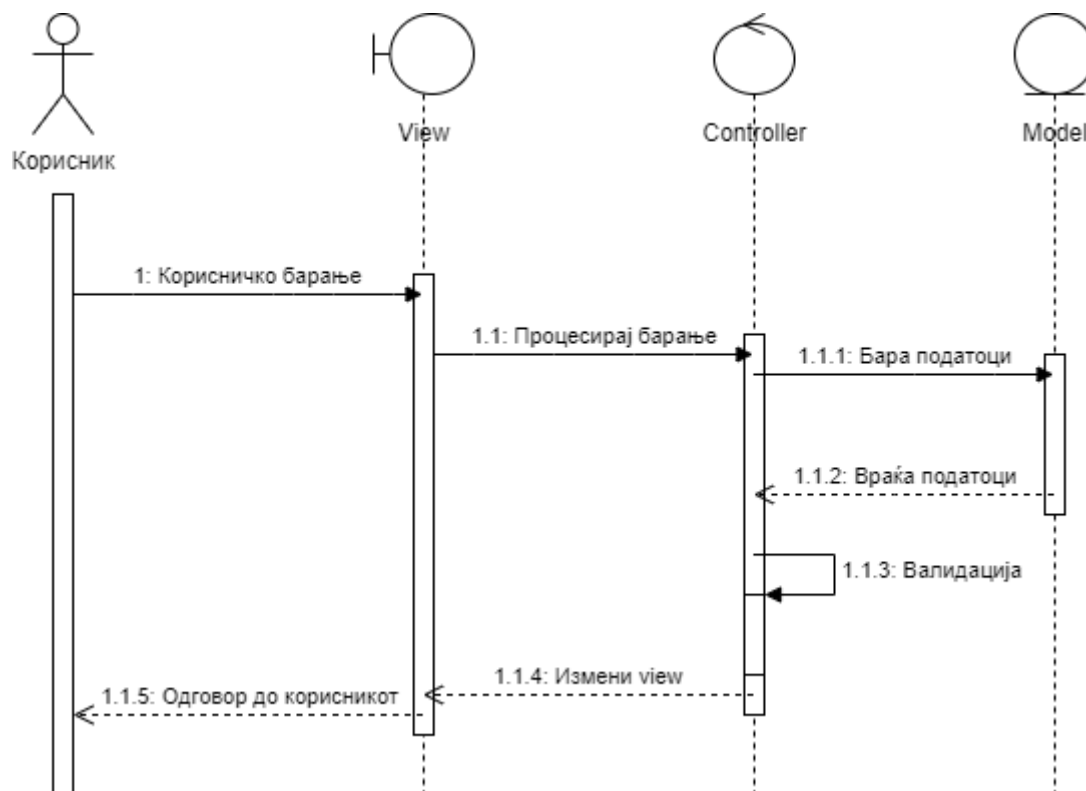


Слика 2: Имплементациска архитектура на веб апликацијата

Веб апликацијата е заснована на MVC шаблотот кој е составен од три дела секој со посебна улога. Тие компоненти се model, view и controller.



Слика 3: MVC шаблон



Слика 4: Секвенцен дијаграм за имплементациска архитектура

Model – тука се дефинирани податоците, однесувањето на апликацијата и бизнис правилата за менување и управување со податоци. Моделот е независен од корисничкиот интерфејс.

View – приказот на апликацијата, генериран приказ за корисникот. Веб прелистувачот не може да посочи на view и успешно да го изрендерира. Неопходно е controller – от да овозможи одредени информации за view – то, а потоа да му предаде податочен објект – model. Задача на view – то е да го трансформира моделот во формат соодветен за презентација на корисникот.

Controller – задолжен за комуникација со корисникот, текот на апликацијата и раководење со барањата.

Тек на апликацијата – Корисникот испраќа барање преку веб прелистувачот. Router – от од MVC шаблонот избира соодветниот контролер да се справи со барањето. Контролерот го процесира барањето и формира соодветен податочен модел кој се препраќа на view – то кое потоа го трансформира во соодветен излезен формат кој се враќа до корисникот, а се рендерира од веб прелистувачот.

Дополнителна инфраструктурна компонента во нашата апликација е структурираната базата на податоци. За нејзина имплементација користиме PostgreSQL.