

# Архитектурни стилови на веб апликацијата

Веб апликацијата претставува хибридна архитектура составена од следните архитектурни стилови:

## Архитектура за проток на податоци

Поконкретно од архитектурата за проток на податоци ние веќе го искористивме концептот на цевка и филтер каде што како влез зедовме податоци од OpenStreetMap. Преку филтерот ги издвоивме оние податоци кои ни се потребни и ги зачувавме во .csv фајлови. Оваа архитектура има за цел да достигне повторна искористливост на кодот. Исто така филтерите не знаат ништо за другите филтери односно се loose coupling.

## Централизирана архитектура

Најпознат пример од централизираната архитектура е базата на податоци. Во нашата апликација .csv фајловите кои ги добивме со филтрирањето претставуваат база на податоци и преку нив ќе ги добиваме информациите за точната локација на локалите. Главната цел на оваа архитектура е да се достигне поголем интегритет на податоците. Исто така се достигнува надежност, сигурност и клиентите се независни од системот.

## Клиент-сервер архитектура

Веб апликацијата се базира на клиент-сервер архитектурата. HTTP е протоколот за комуникација. Клиентите обично се активни корисници кои праќаат барања до серверот. Серверот е пасивен, тој чека на барањата пратени од корисниците и ги праќа соодветниот одговор одназад до корисникот. Локацијата на серверот е транспарентна. Има loose coupling помеѓу серверот и клиентот. Функционалностите се поделени меѓу серверот и клиентот. Со тоа се овозможува јасно дефинирана и дизајнирана комуникација помеѓу нив.

## GUI архитектура

Веб апликацијата се заснова на MVC шаблонот. Го користиме бидејќи е наменет за развој на кориснички интерфејс кој ја дели логиката на поврзаната програма на три меѓусебно поврзани елементи. Тие елементи се: Model, View и Controller. Во Model-от се дефинирани податоците и однесувањето на апликацијата, View-то е всушност приказот на апликацијата, односно генериран изглед кој му се прикажува на корисникот. Controller-от

е задолжен за комуникација со корисникот и текот на апликацијата, односно се справува со барањата.

Корисникот праќа барање преку browser-от. Controller-от го процесира барањето пратено од корисникот и формира податочен Model. View-то го трансформира Model-от во соодветен излезен формат и одговорот се праќа до корисникот.

## **SaaS – Software as a Service**

Како краен дел од изработката на веб апликацијата таа ќе биде хостирана на Cloud, не на сервер кој што припаѓа на некоја друга организација. Со SaaS архитектурата апликацијата е достапна на Cloud. Со оваа архитектура за сите клиенти е достапна една верзија на апликацијата со единствена конфигурација. Предноста на SaaS е тоа што не треба да се грижime за одржливоста и сигурноста на апликацијата.