

## Домашна работа 2

### Архитектура

#### 1. Концептуална архитектура

- Во првите чекори на концептуаната архитектура ги земаме функциските барања за системот и ги подвлекуваме клучните концепти.

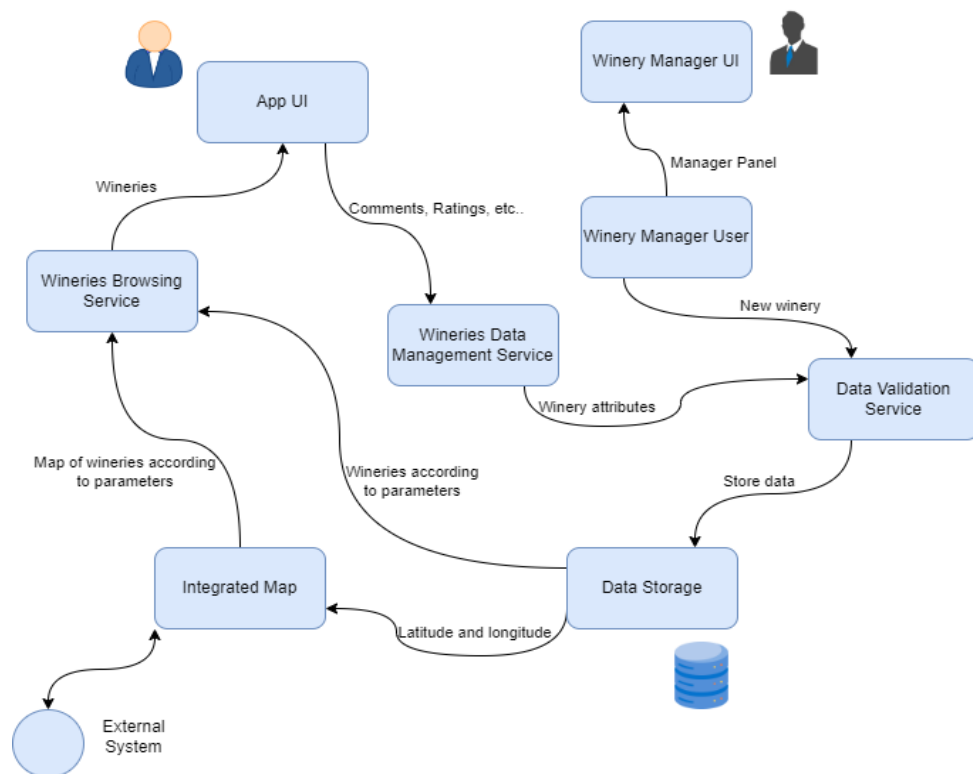
-

1. Системот ќе овозможи приказ на листа на винариите.
2. Системот ќе овозможи додавање на нови винарии.
2. Системот ќе овозможи на корисниците приказ на подетални информации при клик.
3. Системот ќе овозможи преглед на локациите за винариите преку интегрирана мапа.
4. Системот ќе овозможи приказ на одалеченоста до локациите на винариите.
5. Системот ќе овозможи приказ на насоките до одбраната винарија.
6. Системот ќе овозможи пребарување на винарии според рејтинг.
7. Системот ќе овозможи пребарување на винарии според одалеченост.
8. Системот ќе овозможи пребарување на винарии според локација.
9. Системот треба да овозможи бришење на веќе непостоечка винарија.
10. Системот треба да овозможи оценување и коментирање на винариите од корисниците.
11. Системот ќе овозможи приказ на работно време на винариите.
12. Системот треба да биде поврзан со база на податоци во која се чуваат податоци за винариите.
13. Системот треба да овозможи можност за едитирање на доделен кориснички коментар или рејтинг.
14. Системот ќе овозможува креирање на листа на омилени винарии од страна на корисникот.
15. Системот ќе прави валидација на сите внесени податоци.

Потоа клучните концепти ќе бидат категоризирани според следните категории: **Data, Function, Stakeholder, System, Abstract concept.**

Data	Function	Stakeholder	System	Abs. Concept
база на податоци	приказ	обични корисници	Wineries.mk	листа на винарии
винарии	додавање	менаџери на винарии		подетални информации
корисници	бришење			интегрирана мапа
рејтинг	преглед			насоки
одалеченост	пребарување			омилени винарии
локација	оценување			
работно време	коментирање			
коментар	едитирање			
	валидација			

Табела 1 Категоризација на клучни концепти



Слика 1 Концептуална архитектура

Објаснување за секој од компонентите

### **AppUI:**

Крајниот интерфејс прикажан на корисникот, преку кој се остварува интеракција со апликацијата.

### **Winery Manager UI:**

Крајниот интерфејс во форма на кориснички панел за внесување на винарии.

### **Winery Manager User:**

Контролер задолжен за управување на корисничкиот профил на менаџерот на винаријата како и испраќање на податоците внесени за винаријата до сервисот за валидација на податоци.

### **Integrated Map:**

Интегрирана Google Maps, чија единствена улога е да ги прикаже локациите на винариите, а доколку корисникот сака навигација ќе биде пренасочен кон надворешниот сервис на Google Maps со испратена локација до саканата винарија.

### **Wineries Browsing Service:**

Сервис задолжен да ги обработи филтрите и текстот за пребарување внесени од корисникот, и да ги врати винариите кои ги исполнуваат условите од пребарувањето

### **Wineries Data Management Service:**

Секој внес на коментар или рејтинг за винаријата треба да биде валидиран дали е во правилна форма која одговара на базата на податоци. Овој сервис ги превзема податоците од корисникот и ги препраќа до сервисот за валидација во потребниот формат.

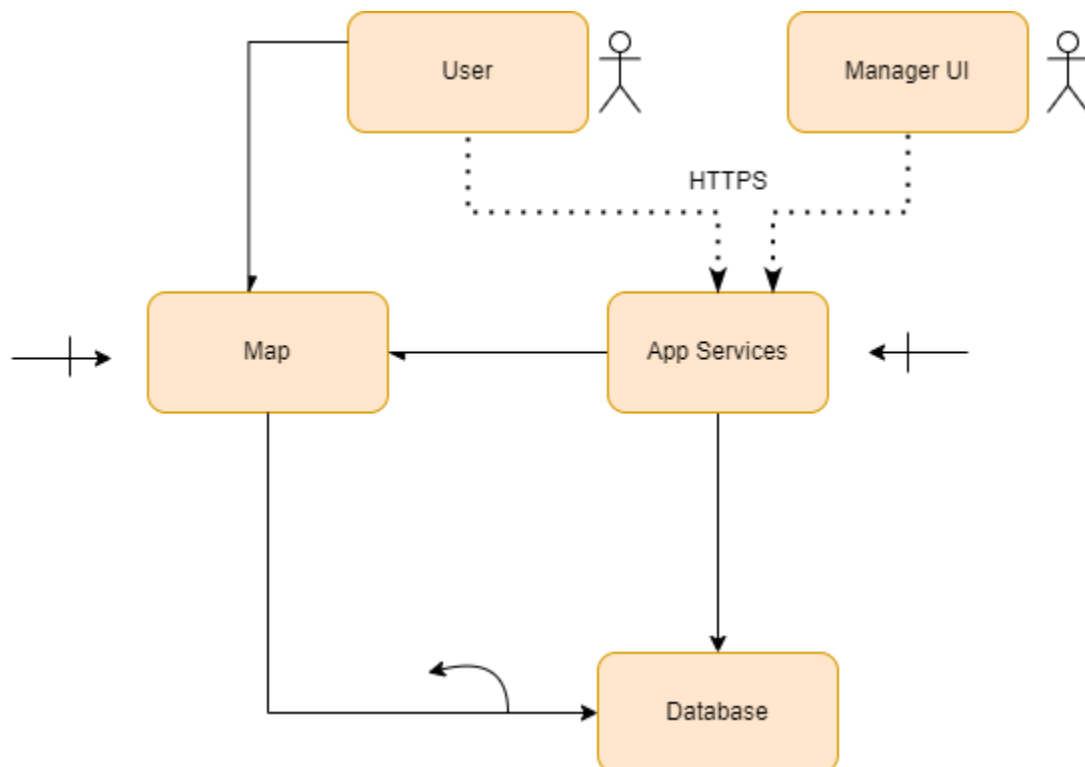
### **Data Validation Service:**

Сите пристигнати податоци кои треба да се запишат во дататазата треба да имаат соодветен формат. Овој сервис проверува дали сите внесени податоци се во валиден формат кој соодветствува на базата, и доколку се ги запишува.

### Data Storage:

База на податоци која ги содржи следните информации за винариите: ид во базата, google id, име, тип, адреса, град, телефонски број, работни саати, вебсајт, дали е отворена или не, рејтинг, коментари, достапност за инвалиди, географска ширина и должина.

## 2. Извршна архитектура



Слика 2 Извршна архитектура

### User:

Графички интерфејс на обичниот корисник на апликацијата.

### Manager UI:

Графички интерфејс на менаџер на винарија.

### App Services:

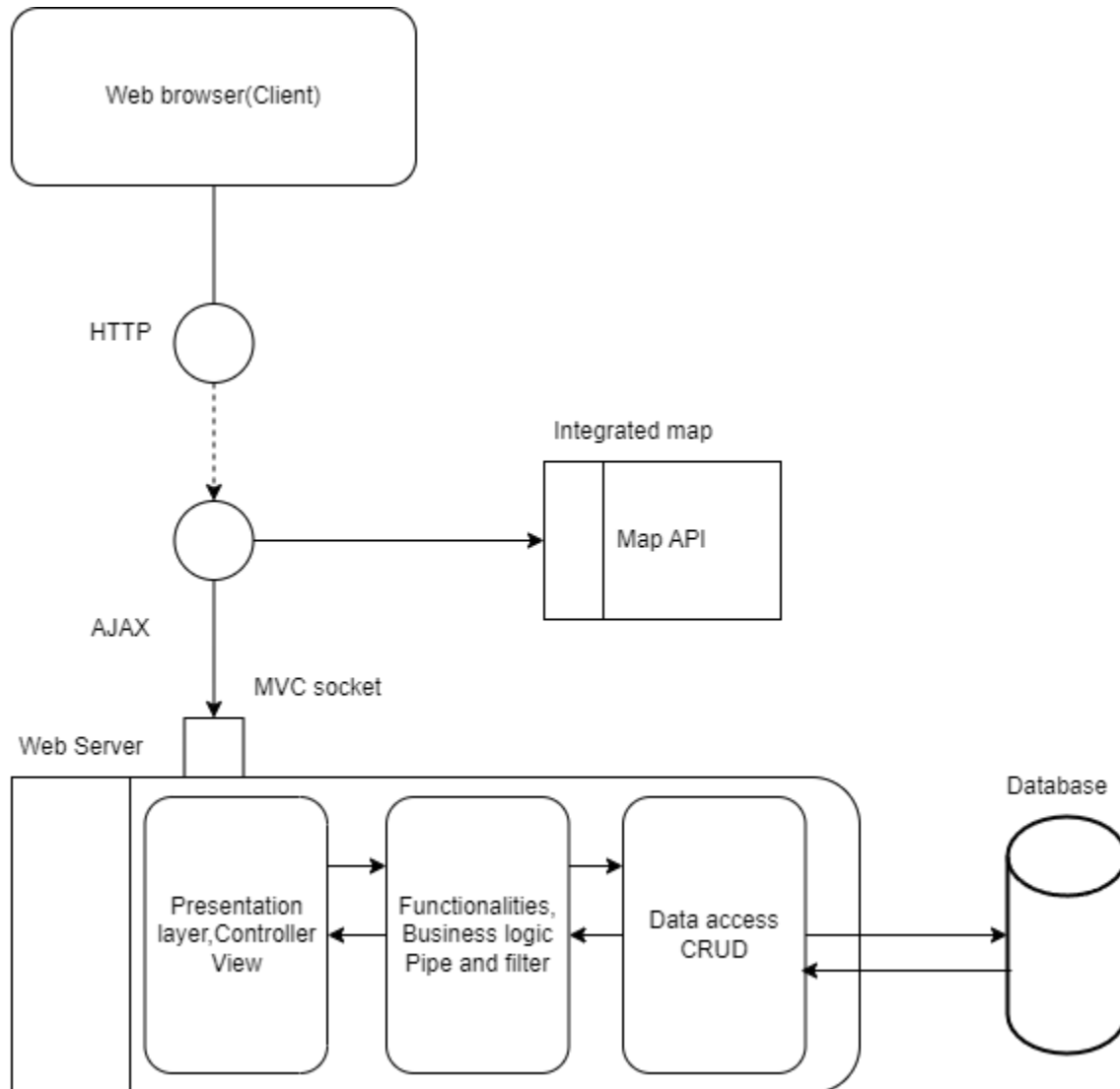
CRUD функционалности, валидација на податоци, ракување со добиените податоци, синхрона врска со датабазата

**Map:**

Надворешен сервис (врска кон GoogleMaps), приказ на локациите на винариите

**Database:**

Податоци за винариите, CRUD функционалности

**3. Имплементациска архитектура**

Објаснување за компонентите

## **Web browser**

Рендерира поглед(view) на корисникот

Праќа HTTP барање и чека одговор од Web Serverot

## **Web Server**

Користи MVC шаблон

- Првиот слој задолжен за препраќање на барањата од клиентите до серверот и од серверот до клиентот како и неговиот изглед. Технологии користени за Frontend како HTML, CSS, JavaScript.
- Вториот слој е апликациската логика комуницирање со првиот слој, филтрирање на текстот за пребарувањата на корисникот и враќање на резултат.
- Третиот слој е Data Access слојот кој користи CRUD операции за функционалностите исто така задолжен за комуникација со самата база.