

Slovenská technická univerzita  
Fakulta informatiky a informačných technológií  
Ilkovičova 2, 842 16 Bratislava 4

# Inžinierske dielo

Dokumentácia k tímovému  
projektu

---



BEHAWORKS

---

Ághová Adriana  
Civáň Martin  
Cukerová Zuzana  
Kováč Matúš

Pilňan Matúš  
Schmidt Andrej  
Villár Viliam

---

# 1 Obsah

1.	Big picture.....	3
1.1.	Úvod.....	3
1.2.	Globálne ciele projektu .....	3
1.2.1.	Zimný semester .....	3
1.2.2.	Letný semester .....	3
1.3.	Celkový pohľad na systém.....	4
2.	Moduly systému.....	5
2.1.	Modul server .....	5
2.1.1.	Modul vizualizácie nameraných črt .....	5
2.1.2.	Modul pre výpočet metrík črt jednotlivých používateľov.....	5
2.1.3.	Modul konfigurácie.....	6
2.1.4.	Modul reprezentácie nameraných dát .....	6
2.2.	Modul logovania – logger .....	6
2.3.	Modul api klienta .....	7

---

# 1. Big picture

## 1.1. Úvod

Nasledujúci dokument obsahuje sumarizovaný pohľad na tímový projekt „BehaPass“. Bližšie si v ňom priblížime nami stanovené globálne ciele, celkovú štruktúru navrhovaného systému (architektúru, dátový model a diagram tried)

## 1.2. Globálne ciele projektu

### 1.2.1. Zimný semester

V rámci šprintov v zimnom semestri sme sa zo začiatku zamerali na analýzu technológií, analýzu samotnej VR ako aj už vyvinutých podobných riešení alebo častí, ktoré môžeme v našom projekte využiť.

V rámci samotnej implementácie projektu sme si stanovili tieto ciele pre ZS:

- Logovanie a spracovanie údajov získaných z ovládačov a okuliarov HTC Vive
- Vizualizácia dát
- Vytvorenie jednotlivých črt na základe ktorých rozpoznáme jednotlivých používateľov
- Vytvorenie modelu na základe ktorého s určitou pravdepodobnosťou rozpoznáme daného používateľa

### 1.2.2. Letný semester

V letnom semestri sme sa už zamerali na zhotovenie samotného systému, ktorého úlohou je úspešná identifikácia používateľa na základe jeho biometrických črt. Systém tvorí webová služba ponúkajúca vytvorenie modelov ľubovoľného počtu používateľov uložených v databáze a identifikáciu z danej množiny používateľov. Aby sme dospeli už ku vytvoreniu samotného produktu, bolo potrebné implementovať:

- Extrahovanie črt z dát z gest
- Vylepšenie modelu
- Vytvorenie klientskej aplikácie
- Rozlišovanie nového a existujúceho používateľa

## 1.3. Celkový pohľad na systém

V rámci nášho vyvíjaného systému môžeme identifikovať dve časti, a to:

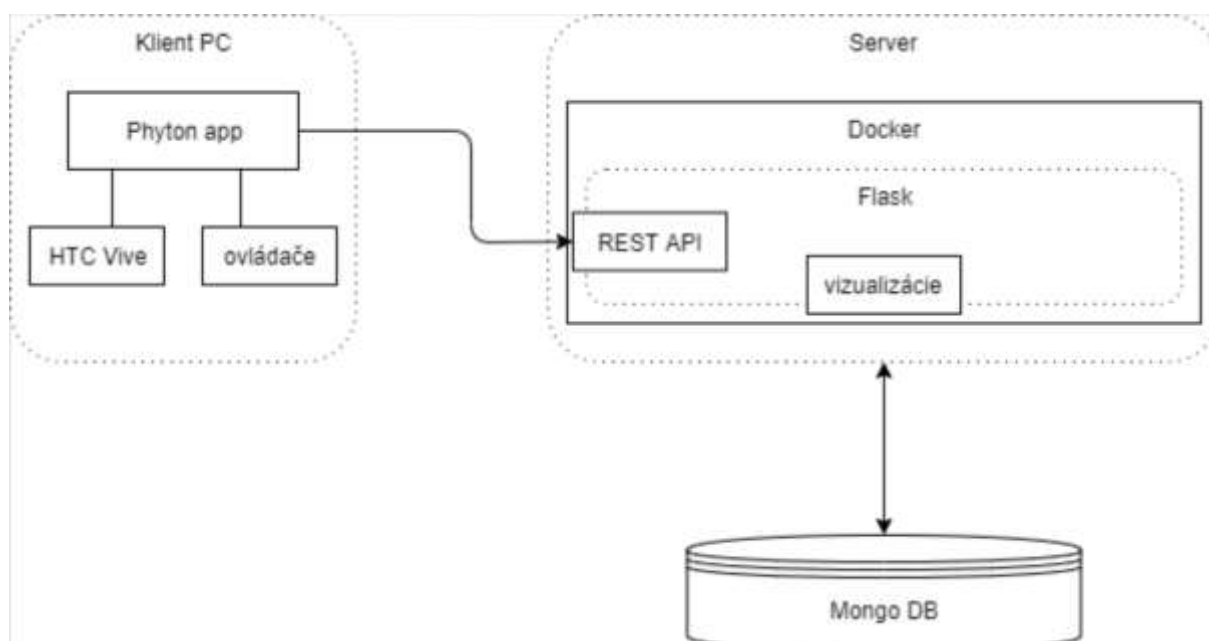
- Klientsku časť
- Server

Klientsku časť predstavuje používateľ virtuálnej reality ktorý využitím ovládačov a okuliarov HTC Vive zaznamenáva vykonané pohyby. Nadobudnuté dáta sú následné odoslané na server použitím REST Api. REST Api je poskytované v rámci webového frameworku Flask, pričom Flask aplikácia beží v Docker kontajneri.

Čo sa týka serverovej časti, samotný server je rozvrhnutý do modulov, ktoré sú zodpovedné za rôzne časti systému. Príkladom môže byť modul zodpovedný za vizualizácie nameraných dát alebo taktiež aj modul vytvorená pre výpočet metrík črt jednotlivých používateľov.

Technická dokumentácia ku api je automaticky generovaná a je dostupná na

<https://team12-19.studenti.fiit.stuba.sk/api/documentation>



Obrázok 1: Architektúra systému

---

## 2. Moduly systému

### 2.1. Modul server

Modul server zodpovedá v našej implementácii za spracovávanie REST api požiadaviek. Ukladá taktiež získané dáta od klienta do databázy. Samotný modul sa skladá z niekoľkých ďalších modulov opísaných nižšie.

#### 2.1.1. Modul vizualizácie nameraných črt

Modul vizualizácie nameraných črt ma za úlohu zobrazovanie jednak nameraných pohybov, ako aj ich charakteristiky vykreslene pomocou grafov.

Na adrese /visualisations je možné zobraziť vizualizácie nalogovaných dát. Vizualizácie sú rozdelené podľa mena používateľa. Každý používateľ má niekoľko sessions. Jedna session reprezentuje jeden pohyb.

#### 2.1.2. Modul pre výpočet metrík črt jednotlivých používateľov

**Modul obsahuje implementáciu týchto črt:**

1. Acceleration (Zrýchlenie)
2. Angular velocity (Uhlová rýchlosť)
3. Controller rotation distance (Vzdialenosť uhlov v kvaterniónoch)
4. Device distance (Vzdialenosť ovládača od headsetu)
5. Jerk (Derivácia zrýchlenia)
6. Straightness (Priamočiarosť)
7. Stroke length (Dĺžka pohybu)
8. Time length (Dĺžka trvania záznamu)
9. Velocity (Rýchlosť)

Z jednotlivých črt sa následne pre každého používateľa vypočítajú atribúty priemer, medián, minimum, maximum, štandardná odchýlka a medzikvartilové rozpätie. Systém podporuje spôsob rozdelenia nameraných črt na menšie kúsky (chunks) za účelom zvýšenia kvality modelu.

Pythonový modul metrix obsahujúci systém počítania črt a metrík sa nachádza v repozitári behapass server.

---

### 2.1.3. Modul konfigurácie

Medzi zoznam modulov patrí aj modul konfigurácie, ktorý obsahuje nastavenia databázy a taktiež cesty k api.

### 2.1.4. Modul reprezentácie nameraných dát

Tento modul bol vytvorený pre triedy na reprezentáciu pohybov.

### 2.1.5. Trénovací model

Systém využíva model IndexFlatL2 z knižnice Faiss na vyhodnotenie podobnosti používateľov.

## 2.2. Modul logovania – data logger

Aplikácia na zbieranie údajov z VR ovládačov a ich odosielanie na server. Formát dát je prevzatý z Open VR API a zahŕňa :

- pozíciu ovládača v priestore
- jeho samotné otočenie
- stlačenie jednotlivých tlačidiel a iné

### Fungovanie

1. Logger zistí počet pripojených ovládačov.
2. Logger čaká na vstup od používateľa.
3. Akonáhle používateľ stlačí tlačidlo nahrávania (nastaviteľné v konfigurácii), začne sa zaznamenávať pohyb. Pohyb sa zaznamenáva kým je dané tlačidlo stlačené.
4. Akonáhle používateľ pustí tlačidlo, nahrávanie skončí a zaznamenané údaje sa odošlú na server.

Pokiaľ sa nepodarí načítať VR rozhranie (môže byť nepripojený headset, prípadne vypnuté SteamVR), aplikácia otestuje odosielania dát na server bez potreby mať pripojený VR headset.

### Konfigurácia

Konfiguračný súbor config/config.json obsahuje nasledovné polia:

- sid\_length - Dĺžka session ID stringu,
- sample\_rate - Vzorkovacia frekvencia snímaného pohybu,
- api\_host - URL, kde sa nachádza logovacie API,

- 
- button - Tlačidlo, ktorým sa ovláda nahrávanie pohybu. Pokiaľ sa zadá neplatná hodnota, používa sa trigger,

## 2.3. Modul api klienta

Automaticky generovaný modul podľa open api špecifikácie tvorený v závislosti od našej api. Slúži na komunikáciu medzi klientom a serverom.

## 2.4. Databáz

Systém využíva MongoDB ako hlavnú databázu na ukladanie záznamov, metrík a používateľov. Všetky údaje sú uložené ako súbory vo formáte json.

## 2.5. Sada utilných skriptov

Dôležitou súčasťou systému je aj sada utilitných skriptov nachádzajúcich sa na serveri v priečinku utils. Skripty ponúkajú širokú škálu služieb užitočných pri tvorbe a evaluácii modelu.

1. compute\_metrix – vypočíta metriky pre záznamy v databáze
2. db\_train\_test\_split – rozdelí vypočítané metriky na trénovaciu a testovaciu sadu
3. db\_train\_test\_split\_by\_user – rozdelí vypočítané metriky na trénovaciu a testovaciu sadu bez toho, aby sa rovnaký používateľ vyskytol v jednej aj druhej sade
4. evaluation – vyhodnotí model na základe metrík
5. feature\_selection – vykoná výber atribútov
6. json\_encoder – potrebná súčasť kvôli MongoDB
7. remove\_outliers – odstráni vychýlené hodnoty