

Slovenská technická univerzita v Bratislave

Fakulta informatiky a informačných technológií

Ilkovičova 2, 842 16 Bratislava 4

Podklady k aplikácii MuseStream

1. milestone z predmetu MTAA

15.3..2022

Sebastián Petrík (ID 110872), Štefan Otto Novák

Predmet: **Mobilné technológie a aplikácie**

Cvičiaci: Ing. Peter Trúchly, PhD

Akademický rok 2021/2022, 2. ročník

Popis funkcionality

Aplikácia slúži na správu hudobných lekcií kde cieľová skupina sú žiaci a učitelia. Učiteľ môže vytvoriť triedu, ktorá má v sebe študentov.

Každý študent triedy má pridelené lekcie. V rámci lekcie je možné posilať správy medzi učiteľom a študentom. Študent môže požiadať o pridanie do triedy alebo je pridaný učiteľom.

Lekcia poskytuje učiteľovi a študentovi pripojiť sa do videohovoru.

Návrh obrazoviek

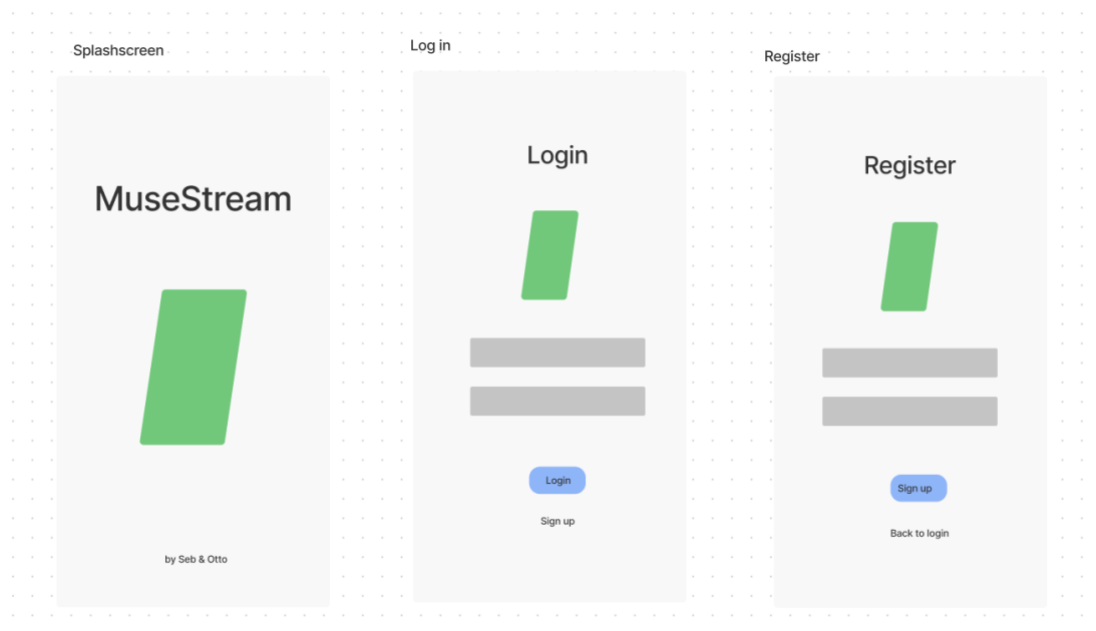
Obrazovky sme navrhli v nástroji Figma Jam. Obrazovky sú rozdielne pre rozdielne typy používateľov – máme 3 typy používateľov – student, teacher a admin.

Každá obrazovka má menu (dostupné cez hamburger tlačidlo).

Pre lepšie prezeranie obrazoviek je možné navštíviť návrh cez link:

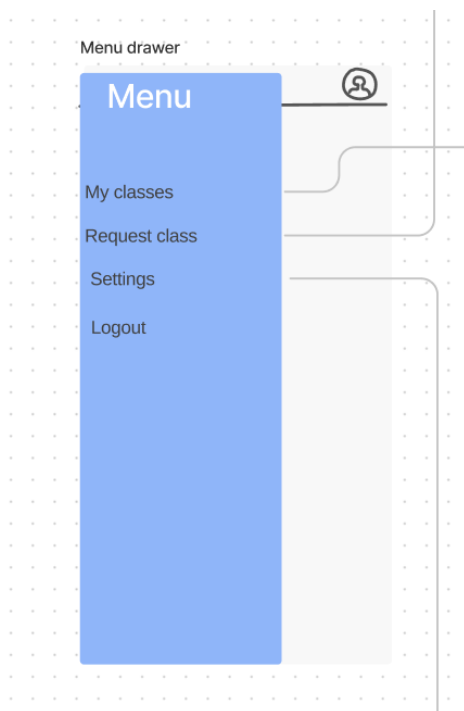
<https://www.figma.com/file/dKFQ21pcRz7ubFZ0WihLdG/MTAA-JAM?node-id=0%3A1.ň>

Úvodné obrazovky

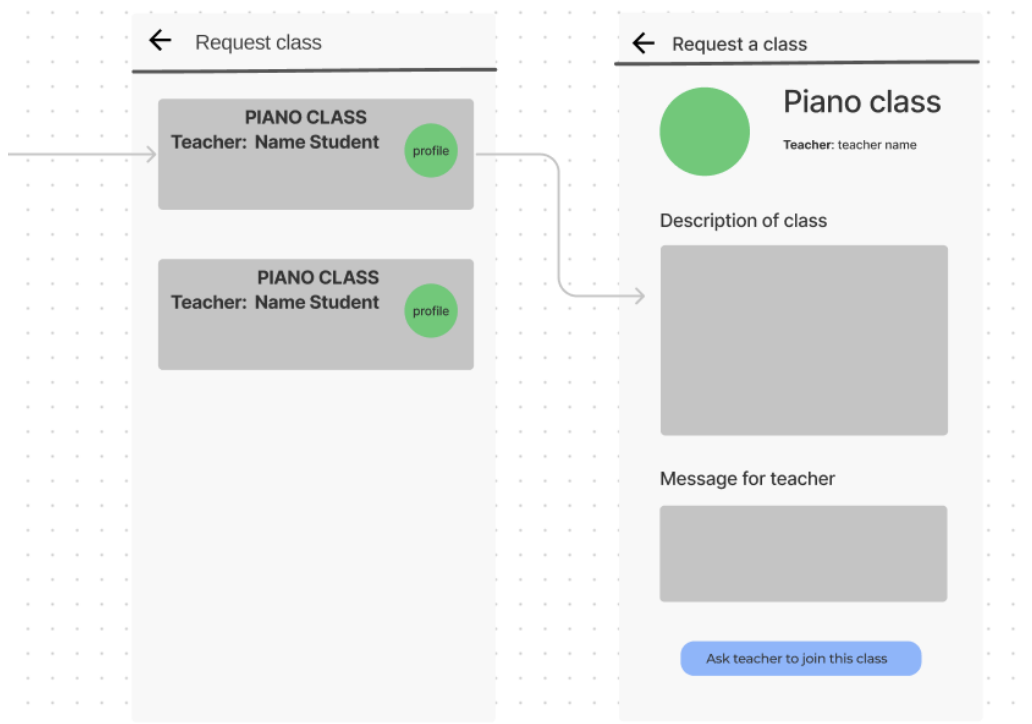


Obr.: Úvodné obrazovky

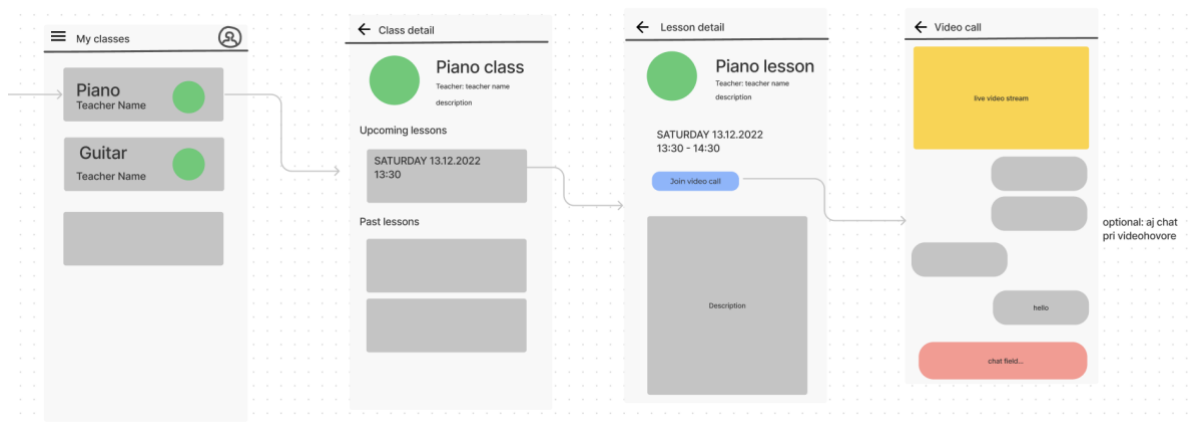
Obrázovky študenta



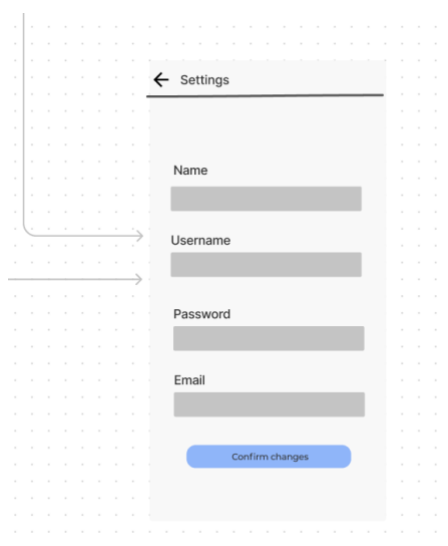
Obr.: Menu študenta



Obr.: Požiadanie učiteľa o pridanie do triedy

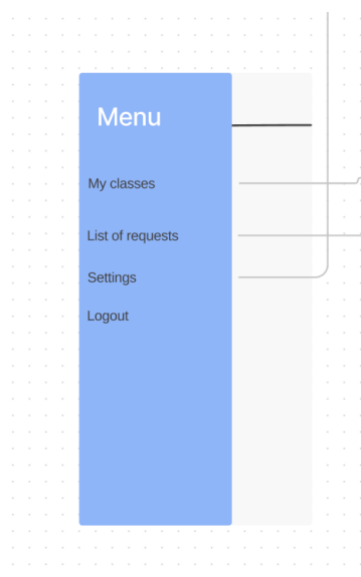


Obr.: Priebeh od zobrazenia tried študenta až po samotný videohovor

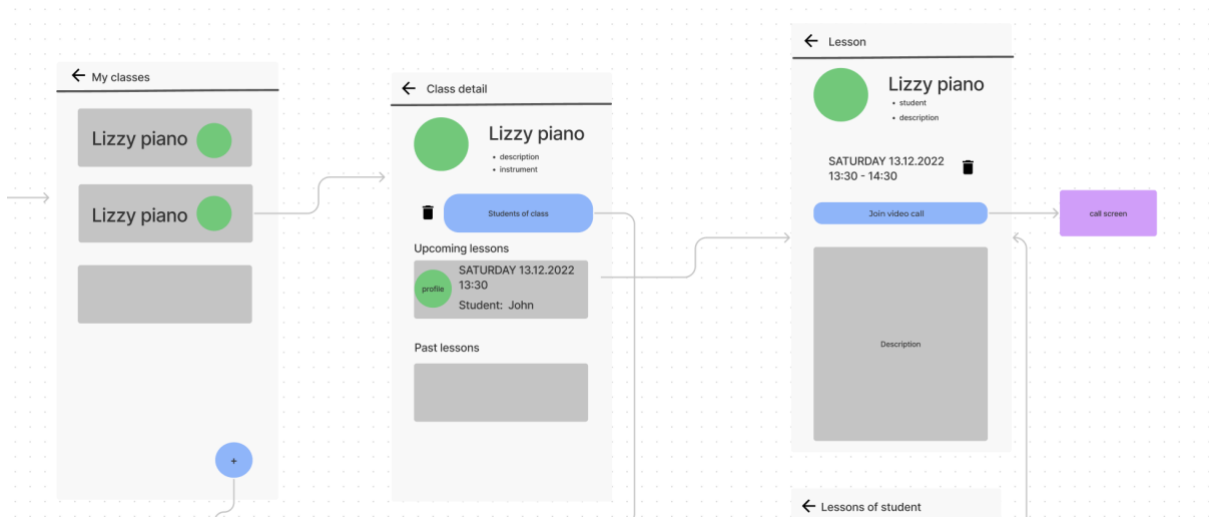


Obr.: Nastavenia študenta (rovnaké pre učiteľa)

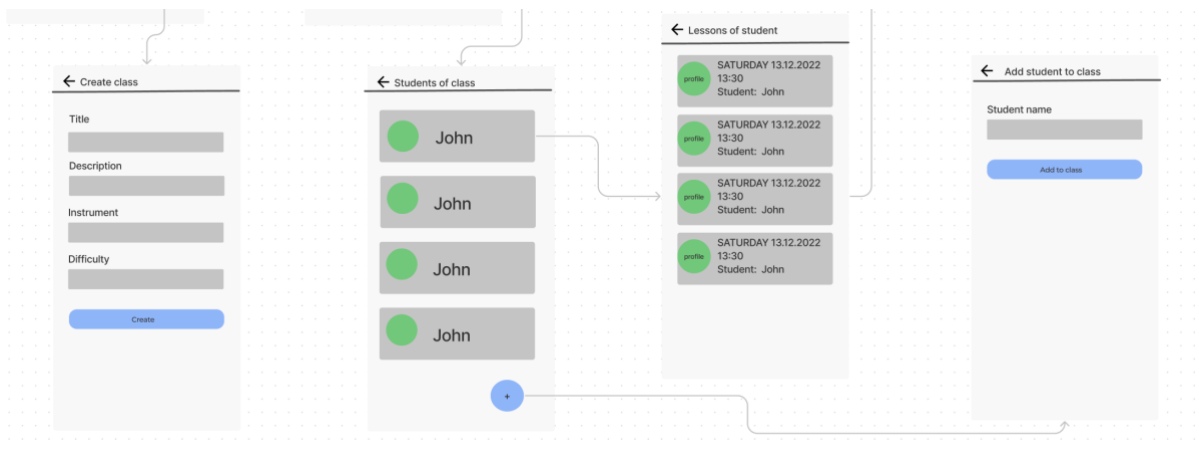
Obrazovky učiteľa



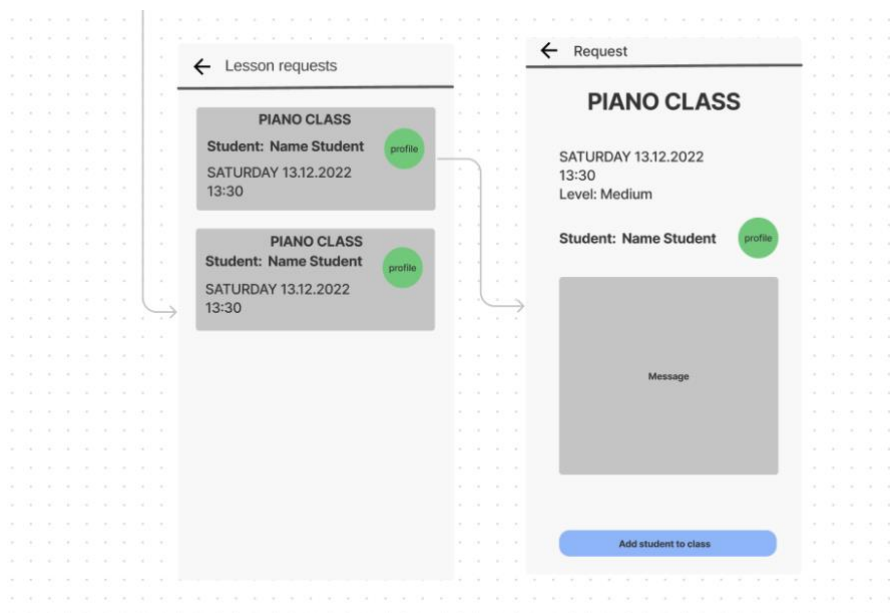
Obr.: Menu učiteľa



Obr.: Priebeh toku od zobrazenia tried až po videohovor pre učiteľa

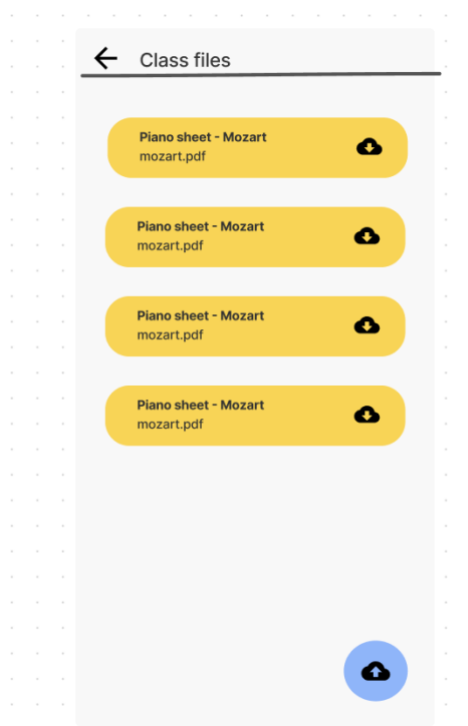


Obr.: Vytvorenie triedy, zobrazenie študentov, lekcie študenta a pridanie študenta do triedy



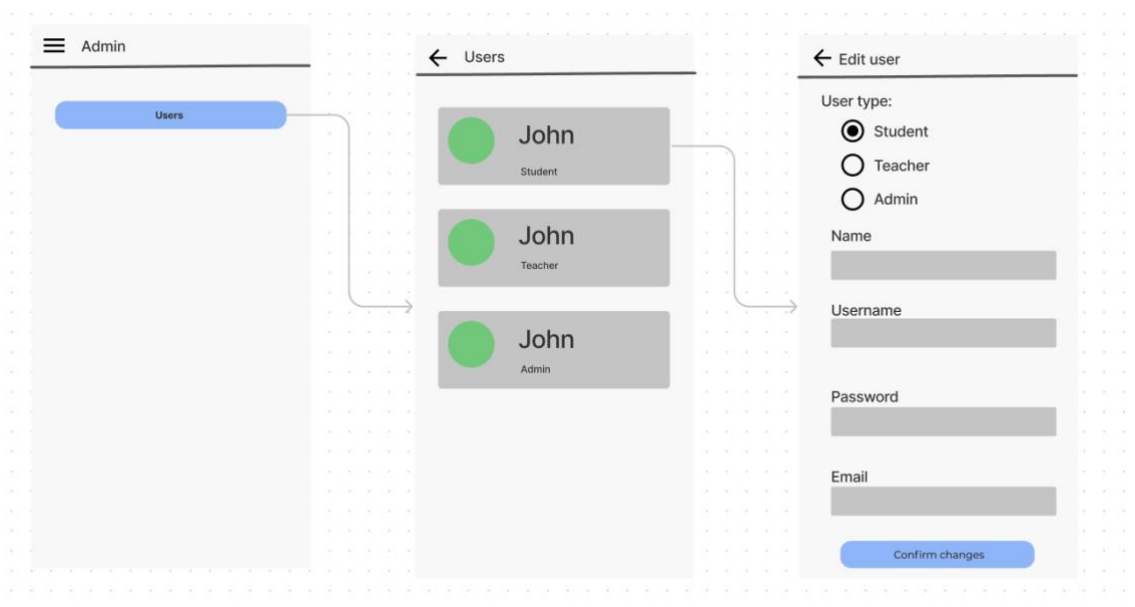
Obr.: Správa požiadavok na pridanie do triedy pre učiteľa

Obrazovka súborov



Obr.: Zoznam súborov s tlačidlami pre stiahnutie a odoslanie

Obrazovky administrátora



Obr.: Obrazovky administrátora – hlavná obrazovka, používateľa, úprava používateľa

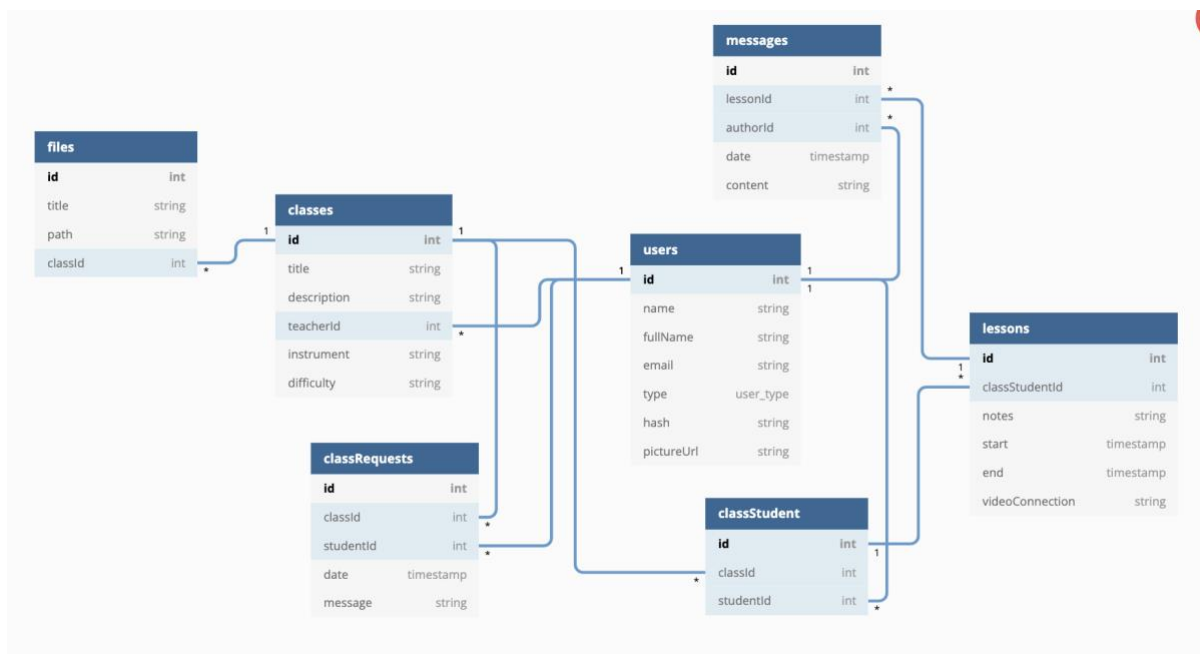
Dokumentácia API endpointov

Dokumentácia API bola písaná vo formáte OpenAPI a je dostupná na linku:
<https://app.swaggerhub.com/apis-docs/Plasmox/MTAA-api/1>.

Kód dokumentácie je dostupný na linku:
<https://app.swaggerhub.com/apis/Plasmox/MTAA-api/1>.

Návrh dátového modelu

Dátový model je možné prezrieť si na diagrame na linku:
<https://dbdiagram.io/d/62279eb761d06e6eadc66795>.



Obr.: Dátový model

Frontend akceptačné testy

Test 1: Žiadosť o registráciu na triedu (positive)	
Vstupné podmienky:	Používateľ, ktorý má rolu študenta je prihlásený v aplikácii, aplikácia má pripojenie na internet
Výstupné podmienky:	Študent vytvorí žiadosť pre registráciu na triedu
Postup:	<ol style="list-style-type: none">1. Používateľ stlačí tlačidlo HAMBURGER IKONA a následne sa otvorí menu na ľavej strane2. Používateľ stlačí tlačidlo REQUEST CLASS3. Zobrazí sa zoznam dostupných tried4. Používateľ stlačí na vybranú triedu5. Zobrazia sa informácie o vybranej triede6. Používateľ stlačí tlačidlo ASK TEACHER TO JOIN THIS CLASS

Test 2: Potvrdenie žiadosti o registráciu na lekcii (positive)	
Vstupné podmienky:	Používateľ, ktorý má rolu učiteľa je prihlásený v aplikácii, aplikácia má pripojenie na internet
Výstupné podmienky:	Študentovi ale aj učiteľovi sa zobrazí potvrdená lekcii vo vlastnej sekcii My Classes
Postup:	<ol style="list-style-type: none">1. Používateľ stlačí tlačidlo HAMBURGER IKONA a následne sa otvorí menu na ľavej strane2. Používateľ stlačí tlačidlo LIST OF REQUESTS3. Zobrazí sa zoznam čakajúcich žiadostí4. Používateľ stlačí na vybranú žiadosť5. Zobrazia sa informácie o vybranej žiadosti6. Používateľ stlačí tlačidlo ADD STUDENT TO CLASS

Test 3: Prihlásenie do aplikácie (positive)	
Vstupné podmienky:	Používateľ otvorí aplikáciu a nie je prihlásený, aplikácia má pripojenie na internet
Výstupné podmienky:	Používateľ sa prihlási do aplikácie
Postup:	<ol style="list-style-type: none"> 1. Používateľ otvorí aplikáciu 2. Zobrazí sa za pár sekúnd Splashscreen 3. Zobrazí sa Login obrazovka 4. Používateľ vloží správne prihlasovacie údaje a stlačí tlačidlo LOGIN 5. Databáza preverí správnosť údajov a sa zobrazí hlavná stránka aplikácie

Test 4: Prihlásenie do aplikácie (negative test)	
Vstupné podmienky:	Používateľ otvorí aplikáciu a nie je prihlásený, aplikácia má pripojenie na internet
Výstupné podmienky:	Používateľ sa neprihlási do aplikácie
Postup:	<ol style="list-style-type: none"> 1. Používateľ otvorí aplikáciu 2. Zobrazí sa za pár sekúnd Splashscreen 3. Zobrazí sa Login obrazovka 4. Používateľ vloží nesprávne prihlasovacie údaje a stlačí tlačidlo LOGIN 5. Databáza preverí správnosť údajov a vypíše správu "The username or password is incorrect"

Test 5: Pripojenie ku video hovoru (negative test)	
Vstupné podmienky:	Používateľ, ktorý má rolu študenta je prihlásený v aplikácii, aplikácia má pripojenie na internet
Výstupné podmienky:	Študent sa nepripojí ku video hovoru
Postup:	<ol style="list-style-type: none"> 1. Používateľ stlačí tlačidlo HAMBURGER IKONA a následne sa otvorí menu na ľavej strane 2. Používateľ stlačí tlačidlo MY CLASSES 3. Zobrazí sa zoznam tried ku ktorým je prihlásený 4. Používateľ stlačí na vybranú triedu 5. Zobrazia sa informácie o vybranej triede 6. Používateľ stlačí na lekciu, ktorá ešte nezačala 7. Zobrazia sa informácie o vybranej triede 8. Používateľ stlačí tlačidlo JOIN VIDEO CALL 9. Video hovor sa nespusti a sa vypíše správa "The lesson hasn't started yet"

Test 6: Zmena údajov (negative test)	
Vstupné podmienky:	Používateľ, ktorý má rolu učiteľa je prihlásený v aplikácii, aplikácia má pripojenie na internet
Výstupné podmienky:	Údaje sa neaktualizujú
Postup:	<ol style="list-style-type: none"> 1. Používateľ stlačí tlačidlo HAMBURGER IKONA a následne sa otvory menu na ľavej strany 2. Používateľ stlačí tlačidlo SETTINGS 3. Zobrazí sa obrazovka s textovými poliami 4. Používateľ vyplní všetky textové polia, okrem pole Email 5. Používateľ stlačí tlačidlo CONFIRM CAHNGES 6. Údaje sa neaktualizujú a sa vypíše správa "Please fill in all the fields"

Test 7: Registrácia účtu (negative test)	
Vstupné podmienky:	Používateľ otvorí aplikáciu a nie je prihlásený, aplikácia má pripojenie na internet
Výstupné podmienky:	Používateľ nevytvorí nový účet
Postup:	<ol style="list-style-type: none"> 1. Používateľ otvorí aplikáciu 2. Zobrazí sa za pár sekúnd Splashscreen 3. Zobrazí sa Login obrazovka 4. Používateľ stlačí tlačidlo SIGN UP 5. Zobrazí sa obrazovka s textovým polom pre Username a Password 6. Používateľ vyplní pole Username s používateľským meno, ktoré už používa iný používateľ 7. Databáza preverí použité používateľské mena a vypíše správu "This username has already been used"

Backend akceptačné testy

Test 1: Registrácia používateľa a prihlásenie (positive)	
Vstupné podmienky:	Používateľ nie je prihlásený ani zaregistrovaný.
Výstupné podmienky:	Vytvorí sa nový používateľ na backende a frontend dostane potvrdenie o autorizácii
Postup:	<ol style="list-style-type: none">1.) Aplikácia pošle POST na /register s JSON dátami pre vytvorenie používateľa2.) Backend odpovie 200 OK – používateľ zaregistrovaný3.) Aplikácia pošle GET na /auth s Basic Auth hlavičkou s menom a heslom používateľa4.) Backend odpovie 200 OK – dané prihlasovacie údaje sú správne a používateľ je autorizovaný
Výsledok: PASS/FAIL	

Test 2: Vytvorenie novej triedy ako učiteľ (positive)	
Vstupné podmienky:	Používateľ je prihlásený ako učiteľ (požiadavky majú Basic Auth header).
Výstupné podmienky:	Vytvorí sa nová trieda, objaví sa v zozname tried daného učiteľa.
Postup:	<ol style="list-style-type: none">1.) Aplikácia pošle POST na /classes s JSON dátami pre vytvorenie novej triedy2.) Backend odpovie 200 OK – trieda bola vytvorená3.) Aplikácia pošle GET na /classes.4.) Backend odpovie 200 OK so zoznamom tried vrátane novovytvorenej triedy.
Výsledok: PASS/FAIL	

Test 3: Akceptovanie požiadavky študenta o prijatie do triedy (positive)	
Vstupné podmienky:	Používateľ je prihlásený ako učiteľ (požiadavky majú Basic Auth header).
Výstupné podmienky:	Požiadavka o prijatie do triedy bude vymazaná, študent bude pridaný do danej triedy.
Postup:	<ol style="list-style-type: none"> 1.) Aplikácia pošle POST na /requests/{requestId}, kde parameter requestId je ID danej požiadavky 2.) Backend odpovie 200 OK – požiadavka bola akceptovaná a študent pridaný do triedy, požiadavka o prijatie bude vymazaná 3.) Aplikácia pošle GET na /classes/{id}/students, kde parameter id je ID triedy, na ktorú bola požiadavka o prijatie adresovaná. 4.) Backend odpovie 200 OK so zoznamom študentov danej triedy, obsahujúcim študenta, od ktorého bola odoslaná požiadavka na prijatie do triedy.
Výsledok: PASS/FAIL	

Test 4: Registrácia používateľa s už existujúcim menom (negative)	
Vstupné podmienky:	Používateľ nie je prihlásený – požiadavky nemajú dáta o autorizácii, v databáze už existuje používateľ s menom „janko“.
Výstupné podmienky:	Backend nedovolí registráciu používateľa s rovnaký menom, aké má už existujúci používateľ
Postup:	<ol style="list-style-type: none"> 1.) Aplikácia pošle POST na /register s JSON dátami pre vytvorenie používateľa, kde meno používateľa je „janko“. 2.) Backend odpovie 400 BAD REQUEST – nemožné zaregistrovať používateľa 3.) Aplikácia pošle GET na /auth s Basic Auth hlavičkou s menom a heslom používateľa 4.) Backend odpovie 401 UNAUTHORIZED – dané prihlasovacie údaje nie sú správne alebo používateľ neexistuje
Výsledok: PASS/FAIL	

Test 5: Aktualizácia dát lekcie (negative)	
Vstupné podmienky:	Používateľ je prihlásený ako učiteľ – požiadavky majú dáta o autorizácii.
Výstupné podmienky:	Backend nedovolí upraviť lekcii s dátumom ukončenia skorším, než je dátum začatia lekcie
Postup:	<ol style="list-style-type: none"> 1.) Aplikácia pošle PUT na /lessons/{id}, kde parameter id je ID existujúcej lekcie, pričom požiadavka obsahuje JSON s atribútmi „start“ a „end“, čo sú dátumy začatia a ukončenia lekcie. Dátum ukončenia je skorší než dátum začatia. 2.) Backend odpovie 400 BAD REQUEST – neplatné aktualizčné dáta 3.) Aplikácia pošle GET na /lessons/{id}, kde parameter id je ID danej lekcie. 4.) Backend odpovie 200 OK, s nezmenenými dátami lekcie (obsahujúc správne dátumy ešte pred pokusom o aktualizáciu).
Výsledok: PASS/FAIL	