

# Követelmény specifikáció – Minta E-learning rendszer

*Programozási technológia 2.*

## Bevezetés

A követelmény specifikáció bemutatja az E-learning rendszerrel szemben támasztott felhasználói követelményeket valamint a rendszer tulajdonságait.

## Scope

Az E-learning rendszer egy távoktatást megvalósító online rendszer, amely lehetőséget biztosít a tanulni vágyóknak folyamatosan frissített, korszerű tananyagokból összeállított kurzusok teljesítésére. A kurzusok valamilyen rendszerességgel közzétett videó vagy szöveges tananyagokból és a számonkérést segítő különféle tesztek, vizsgák állnak, amely anyagokat a tanulók munkáját segítő és értékelő felhasználók (tanárok) állítják össze.

A tananyagok díjának számlázását a hallgatók számára egy külső rendszer kezeli.

## Követelmények áttekintése

Ez a fejezet a rendszer funkcióinak és tulajdonságainak rövid áttekintését ismerteti.

## Interfészek

### Payment service

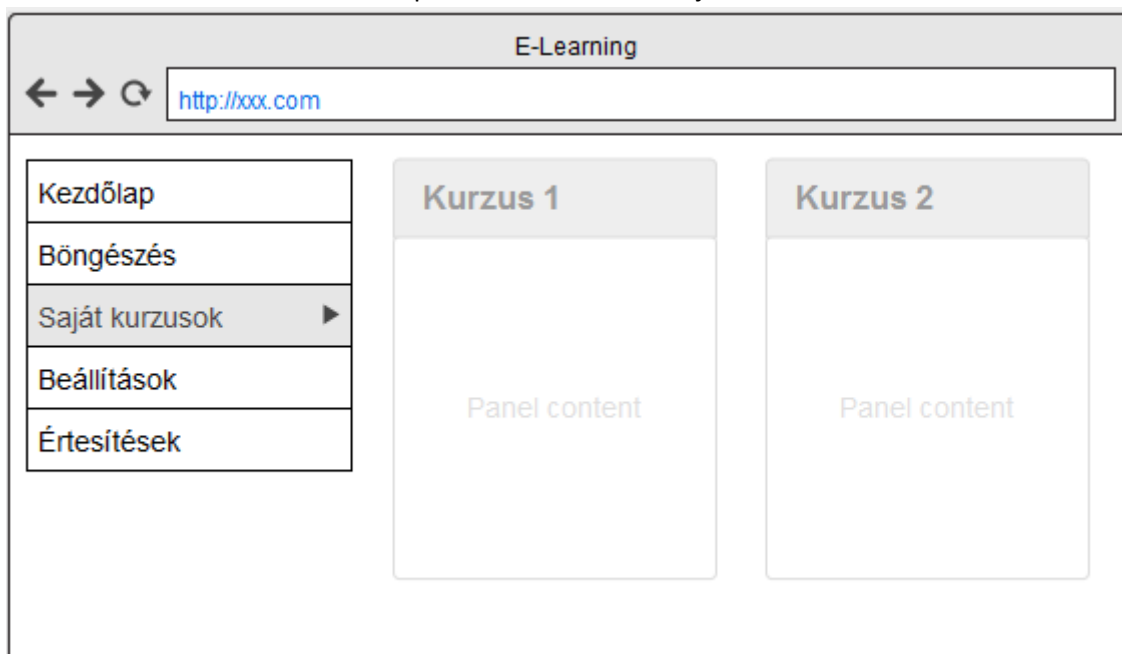
A hallgatók által befizetett összegeket és a számlák kiállítását egy külső számlázó rendszer kezeli. A számlázó rendszert egy REST interfészen keresztül érhetjük el, amely lehetőséget biztosít:

- Befizetések beküldésére
- Számlák kiállítására
- Régebbi számlák lekérése
- Befizetés történet lekérdezésére hallgatónként

### Felhasználói felület

A felhasználói felület elérése böngészőből történik és egy egy-oldalas alkalmazás formájában kerül megvalósításra. A különböző megjelenített információkhoz (kurzusok, vizsgák, tananyagok, tesztek) egyedi linket kell biztosítani.

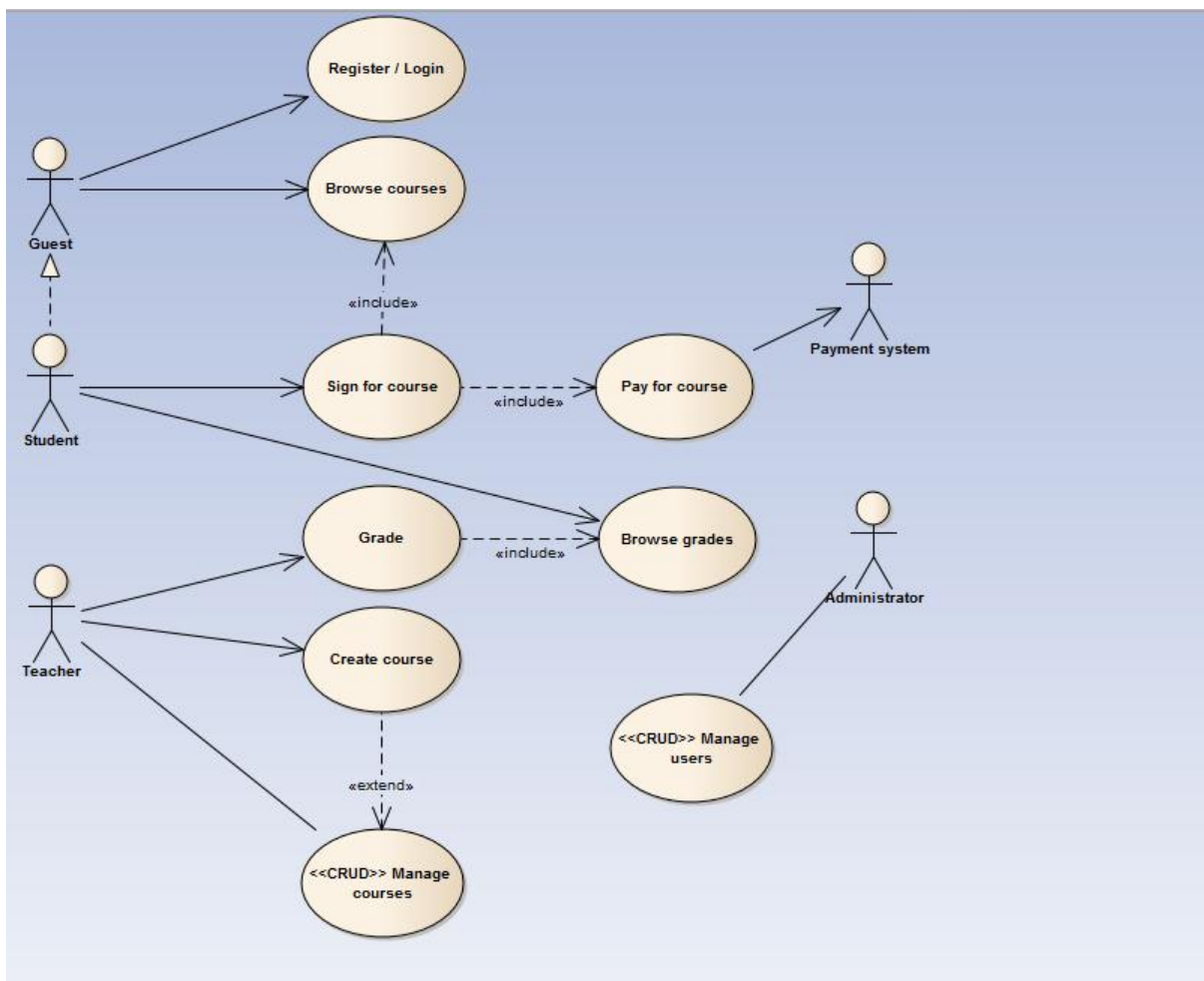
A felhasználói felület általános felpépítést az 1. ábra mutatja.



## Az alkalmazás funkciói

- **Use Case-1: regisztráció** Vendég, (be nem jelentkezett felhasználó) hallgatóként regisztrálhat az oldalra adatai megadásával.
- **Use Case 2: bejelentkezés** Vendég, (be nem jelentkezett felhasználó) bejelentkezhet a rendszerbe felhasználóneve és jelszava megadásával.
- **Use Case 2: kurzusok böngészése** a felhasználók az elérhető kurzusokat listázhatják, a listát különböző feltételek megadásával szűkíthetik.
- **Use Case 3: kurzusra jelentkezés** a bejelentkezett hallgatók kurzusokra jelentkezhetnek.
- **Use Case 4: kurzus fizetése** a jelentkezés után hallgatóknak fizetniük kell a kurzus díját a kurzus megkezdéséhez.
- .....

A funkciók közötti kapcsolatokat a 2. ábra mutatja.



## Követelmények leírása

### Interfészek

#### Payment service

A Payment service REST interfészen keresztül elérhető funkciói. A service használatakor szükséges adatokat a felhasználók adják meg a különböző use case-ek szerint. Pl.: Use Case 4: kurzis fizetése. A megadott adatokat előzetes feldolgozás után a Payment service REST interfészen keresztül kapja meg. Ez a fejezet a használt REST interfész dokumentációját tartalmazza.

<b>Név</b>	Befizetés beküldése
<b>Url</b>	/payment
<b>Method</b>	POST
<b>Url parameters</b>	-
<b>Data parameters</b>	<pre>{   u : {     user_id : [string],     ammount : [integer],     ....   } }</pre>
<b>Sikeres válasz</b>	<b>Code:</b> 200 <b>Content:</b> { payment_id : [integer] }
<b>Hiba válasz</b>	<b>Code:</b> 401 UNAUTHORIZED <b>Content:</b> { error : "Log in" } OR <b>Code:</b> 422 Unprocessable Entry <b>Content:</b> { error : "Email invalid" } OR <b>Code:</b> 404 Not Found <b>Content:</b> { error : "User not found" }

<b>Név</b>	Befizetés történet lekérdezésére hallgatónkénkt
<b>Url</b>	/payment/history/:user_id
<b>Method</b>	GET
<b>Url parameters</b>	<b>Required:</b> user_id =[integer] example: user_id =12
<b>Data parameters</b>	
<b>Sikeres válasz</b>	<b>Code:</b> 200 <b>Content:</b> { ... }
<b>Hiba válasz</b>	<b>Code:</b> 401 UNAUTHORIZED <b>Content:</b> { error : "Log in" } OR <b>Code:</b> 404 Not Found <b>Content:</b> { error : "User not found" }

STB.....

#### Felhasználói felület

[A felhasználói felület leírása képernyőképekkel. A képernyőképek részeinek a definiálásval. Részletesen. Minden egyes funkcióhoz]

## A rendszer funkciói

### Use Case-1: regisztráció

Vendég, (be nem jelentkezett felhasználó) hallgatóként regisztrálhat az oldalra adatai megadásával.

<b>Elsődleges Actor</b>	Vendég
<b>Előfeltétel</b>	A felhasználó nincs bejelentkezve
<b>Sikertelen eredmény</b>	A rögzített felhasználók nem változnak, a felmerült probléma leírása megjelenik a felhasználónak.
<b>Sikeres eredmény</b>	Új felhasználó rögzítésre kerül a felhasználók között: nevének, email címének, felhasználónevének, jelszavának és számlázási adatainak rögzítésével.
<b>Kiváltja</b>	A felhasználó a regisztrációs oldalra lépett.
<b>Események</b>	<ol style="list-style-type: none"><li>1. A felhasználó megadta a regisztrációhoz szükséges adatokat: név, email cím, felhasználóné, jelszó (2x) és számlázási adatok</li><li>2. A felhasználó a regisztráció elküldése gombra kattint</li><li>3. A rendszer ellenőrzi az email cím helyességét.</li><li>4. A rendszer ellenőrzi, hogy a megadott két jelszó egyezik-e.</li><li>5. A rendszer a jelszóból hash-t állít elő.</li><li>6. A rendszer rögzíti a regisztráció időpontját.</li><li>7. A rendszer rögzíti a regisztrációkor használt IP címet.</li><li>8. A rendszer adatbázisba menti a felhasználó adatait: név, email cím, felhasználóné, jelszó, számlázási adatok, regisztráció dátuma, regisztrációs IP.</li><li>9. A rendszer megerősítő emailt küld a felhasználó által megadott email címre.</li><li>10. .) A Use case sikeres eredménnyel zárul</li></ol>
<b>Kiegészítés</b>	<p>3a.) A megadott email cím formátuma hibás 3a2.) A rendszer jelzi a hibát „Nem megfelelő email cím formátum” 3a3.) A Use case sikertelen eredménnyel zárul</p> <p>4a.) A megadott jelszavak nem egyeznek 4a2.) A rendszer jelzi a hibát „A megadott jelszavak nem egyeznek” 4a3.) A Use case sikertelen eredménnyel zárul</p>

### Use Case 3: kurzusra jelentkezés

a bejelentkezett hallgatók kurzusokra jelentkezhetnek.

<b>Elsődleges Actor</b>	Hallgató
<b>Előfeltétel</b>	A felhasználó bejelentkezett, státusza hallgató.
<b>Sikertelen eredmény</b>	A kurzus adatai és feljelentkezett hallgatói változatlanok maradnak a felmerült probléma leírása megjelenik a felhasználónak.
<b>Sikeres eredmény</b>	A felhasználó bekerül a kurzusra jelentkező hallgatók közé.
<b>Kiváltja</b>	A felhasználó a <b>Use Case 2: kurzusok böngészése</b> felületen valamely kurzushoz tartozó jelentkezés opciót választja.
<b>Események</b>	<ol style="list-style-type: none"><li>1. A rendszer megejelíti a kurzus részletes adatlapját valamint a fizetési lehetőségeket. (Bankkártya/PayPal)</li><li>2. A rendszer megejelíti a kurzus részletes adatlapját.</li><li>3. A Use case a <b>Use Case 4: kurzus fizetése</b> Use Case-el folytatódik.</li><li>4. A rendszer rögzíti a kurzus hallgatói között a felhasználót a jelentkezés időpontjával.</li><li>5. A rendszer rögzíti, hogy az adott felhasználó hanyadszor jelentkezik az adott kurzusra.</li><li>6. A rendszer emailben elküldi a kurzushoz tartozó egyéb tudnivalókat a felhasználó számára.</li><li>7. A Use case sikeres eredménnyel zárul</li></ol>
<b>Kiegészítés</b>	3a.) A fizetés sikertelen volt 3a2.) A rendszer jelzi a hibát „Sikertelen fizetés: ”+hiba 3a3.) A Use case sikertelen eredménnyel zárul

## Use Case 4: kurzus fizetése

a bejelentkezett hallgatók kurzusokra jelentkezhetnek.

<b>Elsődleges Actor</b>	Hallgató, Payment service
<b>Előfeltétel</b>	A felhasználó bejelentkezett, státusza hallgató és a <b>Use Case 3: kurzusra jelentkezés</b> megfelelő pontján áll.
<b>Sikertelen eredmény</b>	A fizetési adatok nem változnak, a felmerült probléma leírása megjelenik a felhasználónak.
<b>Sikeres eredmény</b>	A fizetési adatokat rögzíti a számlázási rendszer, elékszül a számla.
<b>Kiváltja</b>	<b>Use Case 3: kurzusra jelentkezés.</b>
<b>Események</b>	<ol style="list-style-type: none"> <li>1. A rendszer meghatározza a hallgató számára jóváírható kedvezményt <b>Rule 1 hallgatói kedvezmény meghatározása</b> – ban leírt számítási mód alapján.</li> <li>2. A rendszer a kiszámított kedvezménynek megfelelően módosítja a kurzus árát.</li> <li>3. A rendszer megejelíti a fizetési lehetőségeket és a fizetendő összeget. (Bankkártya/PayPal)</li> <li>4. A felhasználó kiválasztja a kívánt fizetési lehetőséget.</li> <li>5. A rendszer feldolgozza a fizetési rendszertől kapott adatokat.</li> <li>6. A rendszer a számlázási rendszer (Payment service) felé továbbítja a fizetés adatait.</li> <li>7. A rendszer lekéri a számlázási rendszer által előállított számlát.</li> <li>8. A rendszer emailben elküldi a számlát a felhasználó részére.</li> <li>9. A Use case sikeres eredménnyel zárul</li> </ol>
<b>Kiegészítés</b>	<p>4a.) A választott fizetési lehetőség: Paypal  4a2.) A felhasználó a paypal oldalán keresztül elvégzi a fizetést  4a3.) A Use case a 5. pontban folytatódik</p> <p>4b.) A választott fizetési lehetőség: Bankkártya  4b2.) A felhasználó a XXXX bank oldalán keresztül elvégzi a fizetést  4b3.) A Use case a 5. pontban folytatódik</p> <p>5a.) A fizetés sikertelen volt  5a2.) A rendszer jelzi a hibát „Sikertelen fizetés: ”+hiba  5a3.) A Use case sikertelen eredménnyel zárul</p>

## Rule 1 hallgatói kedvezmény meghatározása

[Annak leírása, hogy a hallgató milyen felételek mellett milyen kedvezményekben részesülhet]