# Tartalomjegyzék

[Tartalomjegyzék 1](#_Toc190513538)

[Bevezetés 2](#_Toc190513539)

[Témaválasztás 2](#_Toc190513540)

[Tervezés, drótvázak 4](#_Toc190513541)

[Felhasználói dokumentáció 5](#_Toc190513542)

[A program célja és lényegesebb funkciói 5](#_Toc190513543)

[Futtatási környezet 5](#_Toc190513544)

[A program fő funkcióinak leírása 6](#_Toc190513545)

[Fejlesztői dokumentáció 8](#_Toc190513546)

[Fejlesztői környezet 8](#_Toc190513547)

[Program struktúra 8](#_Toc190513548)

[Adatbázis 8](#_Toc190513549)

[Leírás 8](#_Toc190513550)

[Funkciók 9](#_Toc190513551)

[Szerkezet 9](#_Toc190513552)

[Egyed kapcsolat diagram 10](#_Toc190513553)

[Kapcsolati ábra 10](#_Toc190513554)

[Táblák szerkezete a tábla szintű megszorításokkal 11](#_Toc190513555)

[Fogalmak 11](#_Toc190513556)

[Hasznos lekérdezések 11](#_Toc190513557)

[Beépítendő megszorítások 11](#_Toc190513558)

[Megvalósítás 11](#_Toc190513559)

[Scriptek 11](#_Toc190513560)

[Teszt adatok 11](#_Toc190513561)

[Továbbfejlesztési ötletek 11](#_Toc190513562)

[Backend 11](#_Toc190513563)

[Modellek és Controllerek 11](#_Toc190513564)

[API végpontok 11](#_Toc190513565)

[Frontend 12](#_Toc190513566)

[Reszponzivitás 12](#_Toc190513567)

[Tesztelés 13](#_Toc190513568)

[Fejlesztési lehetőségek 14](#_Toc190513569)

[Irodalomjegyzék 15](#_Toc190513570)

# Bevezetés

## Témaválasztás

A szakdolgozat témájának a magánklinika online foglalási rendszer fejlesztését választottuk, mert ez egy valós problémára ad digitális megoldást. Ez javíthatja a klinikák működését és a páciensek élményét.

Az egészségügyi szektor egyre inkább digitalizálódik, és egy hatékony, felhasználóbarát foglalási rendszer nagy előnyt jelenthet a magánklinikák számára. Az emberek hozzászoktak az online időpontfoglaláshoz, és elvárják, hogy gyorsan és kényelmesen intézhessék ügyeiket.

A projekt egy valós életben is használható rendszert mutat be, amelyet egy valódi klinika is bevezethetne.

A projekt során többféle technológiát kell alkalmazni, például:

* Backend fejlesztés (adatbázis-kezelés, API-k, autentikáció)
* Frontend fejlesztés (felhasználóbarát UI, reszponzivitás)
* Mobilbarát megoldások (mivel a páciensek sok esetben mobilról foglalnak)
* Tesztelési folyamatok (automatizált és manuális tesztelés a megbízhatóság érdekében)

Ezek hozzájárulnak ahhoz, hogy egy komplex, fejlesztői és tesztelői kihívást oldjunk meg.

# Tervezés, drótvázak

# Felhasználói dokumentáció

## A program célja és lényegesebb funkciói

A feladat célja egy orvos-páciens menedzsment rendszer létrehozása, amely lehetőséget biztosít a látogatóknak a regisztrációra, az orvosi vizsgálatok közötti keresésre, valamint a vizsgálatok időpontfoglalására és kezelésére.

A páciensek bejelentkezés után egyszerűen megtekinthetik, módosíthatják és lemondhatják a jövőbeni időpontjaikat, valamint kiválaszthatják a szabad időpontok közül azokat, amelyek az általuk kívánt eljárásokhoz elérhetőek, tehát a páciens egy szakon belül lévő vizsgálatra kiválaszt egy szabad időpontot, és látja a hozzá tartozó orvosokat. Az orvos kiválasztását követően le tudja foglalni a vizsgálatot.

Az orvosok saját profiljukon keresztül kezelhetik rendelési idejüket és nyomon követhetik a páciensek értékeléseit. Egy orvoshoz egy szak tartozhat. A páciensek kórtörténetét tudja módosítani (leírás betegségekről, kezelésekről, stb.).

Az adminisztrátorok, akik az orvosok között vannak (orvos tábla true / false értéke) hozzáféréssel rendelkeznek a felhasználói nem szenzitív adatokhoz és a foglalások kezeléséhez. Orvosokat regisztrálhat

## Futtatási környezet

Az ötletelés fázisában gyorsan felmerült, hogy időpontfoglaló applikáció készítése illeszkedik leginkább az elképzelésekhez, amelyet webalkalmazás formájában kívánunk véghez vinni. Ennek oka, hogy kiemelten fontosnak tartottuk a felhasználóbarátságot, és résztvevők sokrétűsége miatti általános megtekinthetőséget. Ezáltal telefonon, vagy nagy képernyő előtt számítógépen ugyanúgy megtekinthető a felület külön alkalmazás telepítése nélkül, amely bizonyos felhasználók esetén nehézséget okozna. Emellett fontosnak tartottuk a platform független működést, így a felhasználó számára is egyszerűbb egy weboldal megtekintése, mint egy alkalmazás futtatása (esetlegesen frissítést követően), emellett nem követel meg átlagon felüli sávszélességet. A megtekintésre a webböngészők friss verziói a legalkalmasabbak (Microsoft Edge, Google Chrome, Mozilla Firefox, Opera).

## A program fő funkcióinak leírása

A feladat célja egy orvos-páciens menedzsment rendszer létrehozása, amely lehetőséget biztosít a látogatóknak a regisztrációra, az orvosi vizsgálatok közötti keresésre, valamint a vizsgálatok időpontfoglalására és kezelésére.

A páciensek bejelentkezés után egyszerűen megtekinthetik, módosíthatják és lemondhatják a jövőbeni időpontjaikat, valamint kiválaszthatják a szabad időpontok közül azokat, amelyek az általuk kívánt eljárásokhoz elérhetőek, tehát a páciens egy szakon belül lévő vizsgálatra kiválaszt egy szabad időpontot, és látja a hozzá tartozó orvosokat. Az orvos kiválasztását követően le tudja foglalni a vizsgálatot.

Az orvosok saját profiljukon keresztül kezelhetik rendelési idejüket és nyomon követhetik a páciensek értékeléseit. Egy orvoshoz egy szak tartozhat. A páciensek kórtörténetét tudja módosítani (leírás betegségekről, kezelésekről, stb.).

Az adminisztrátorok, akik az orvosok között vannak (orvos tábla true / false értéke) hozzáféréssel rendelkeznek a felhasználói nem szenzitív adatokhoz és a foglalások kezeléséhez. Orvosokat regisztrálhat.

A program fő célja

A szakdolgozat keretében fejlesztett rendszer egy magánklinika online időpontfoglaló és menedzsment rendszerének létrehozása, amelynek célja, hogy egyszerűsítse és digitalizálja a klinika működésével kapcsolatos legfontosabb folyamatokat.

A rendszer fő céljai a következők:

1. Könnyű és gyors időpontfoglalás
   * A páciensek online, néhány kattintással foglalhatnak időpontot a megfelelő szakterület orvosához.
   * Az elérhető időpontokat valós időben frissíti a rendszer.
2. Orvosok és adminisztrátorok munkájának megkönnyítése
   * Az orvosok megadhatják rendelési idejüket, szabadságukat, és kezelhetik a foglalásokat.
   * Az adminisztrátorok szükség esetén módosíthatják az időpontokat és kezelhetik a felhasználókat.
3. Felhasználóbarát és mobilbarát felület
   * A rendszer reszponzív dizájnnal készül, így asztali gépen és mobiltelefonon is könnyen használható.
   * Letisztult és intuitív felület biztosítja a gördülékeny navigációt.
4. Értesítések és emlékeztetők
   * A páciensek automatikus értesítést kapnak foglalásukról és annak közelgő időpontjáról.
   * Az értesítések csökkentik az elfelejtett időpontok számát, optimalizálva az orvosok időbeosztását.
5. Biztonságos adatkezelés és jogosultságkezelés
   * A rendszer különböző szerepköröket használ (páciens, orvos, admin), hogy mindenki csak a számára releváns adatokhoz férjen hozzá.
   * Az érzékeny egészségügyi adatok védelméről modern titkosítási és biztonsági eljárások gondoskodnak.
6. Statisztikák és jelentések
   * Az adminisztrátorok és orvosok betekintést nyerhetnek a foglalási statisztikákba, például a leggyakoribb szolgáltatásokba vagy a kihasználtsági arányokba.
   * Az adatelemzés segíthet a klinika működésének optimalizálásában.

# Fejlesztői dokumentáció

## Fejlesztői környezet

A létrehozott magánklinika alkalmazás szerver-kliens alapon működik, segítve az erőforrás optimálisabb használatát. Adatbázis szempontjából Microsoft SQL Servert alkalmaztunk. Szerver oldalon PHP programozási nyelvet használtunk, illetve RESTful mentén Laravel keretrendszert használtunk. Kliens oldali megvalósítás Javascript segítségével történt. A fejlesztést Visual Studio Code-ban végeztünk, verziókövetésre GitHub-ot használtunk.

## Program struktúra

## Adatbázis

### Leírás

A Magánklinika adatbázis a klinika betegfoglalásait, orvosi szakrendeléseit és felhasználóit kezeli.

Az adatbázisban kezeljük a felhasználók adatait, akik lehetnek páciensek, orvosok vagy adminisztrátorok. A felhasználók szerepkör alapján vannak besorolva, amely meghatározza a jogosultságaikat a rendszerben. Minden orvos rendelkezik egy adott szakosodással, amelyet a specializations tábla tartalmaz.

A páciensek foglalhatnak időpontokat az orvosokhoz a doctor\_appointments táblán keresztül. Egy foglalás tartalmazza az orvos azonosítóját, a beteg azonosítóját, a foglalás kezdeti időpontját, a kiválasztott kezelést és annak állapotát. Az időpontokat az orvosok szabad kapacitásai alapján lehet lefoglalni.

A rendszerben a kezelések és azok időtartama is nyilván van tartva a treatments táblában. Minden kezeléshez tartozik egy adott szakosodás, amely biztosítja, hogy csak megfelelő szaktudással rendelkező orvosok végezhetik el az adott beavatkozásokat.

A navigations és navigation\_roles táblák kezelik a rendszer menüpontjait és azt, hogy egy adott szerepkör milyen felületeket érhet el. Az adminisztrátorok jogosultak kezelni a felhasználókat, orvosokat, kezeléseket és a páciensek foglalásait.

A rendszer lehetőséget biztosít az orvosok értékelésére is, amely segíti a pácienseket a megfelelő szakember kiválasztásában. Az adatbázis biztosítja a betegadatok biztonságos tárolását és az orvosi időpontok hatékony kezelését.

### Funkciók

- Felhasználók (páciensek, orvosok, adminisztrátorok) felvétele, módosítása, törlése

- Páciensek időpontfoglalása, módosítása, törlése

- Orvosi szakrendelések kezelése

- Kezelések nyilvántartása, módosítása, törlése

- Orvosok szakosodásának kezelése

- Navigációs menüpontok és jogosultságok kezelése

- Adminisztrátori műveletek: felhasználók, kezelések és foglalások menedzselése

- Orvosi értékelések kezelése és megjelenítése

- Statisztikák és riportok generálása

### Szerkezet

jelölés:   
Tábla magyar neve (TÉNYLEGES TÁBLA NEVE)  
(kulcs, leíró, *kapcsolathordozó*)  
Rövid leírás

Felhasználói szerepkörök (ROLES)

{role\_id, name}  
A rendszerben minden felhasználó egy meghatározott szerepkörbe tartozik, amely meghatározza a jogosultságait. Például az adminisztrátorok kezelhetik a felhasználókat, míg az orvosok csak a saját időpontjaikat láthatják és módosíthatják. A szerepkörök előre definiáltak, és minden egyes felhasználónak pontosan egy szerepkör van hozzárendelve.

Felhasználók (USERS)  
{id, name, phone\_number, email, email\_verified\_at, password, *role\_id*, remember\_token}

Ez a tábla tárolja a rendszer összes regisztrált felhasználóját, függetlenül attól, hogy páciensekről, orvosokról vagy adminisztrátorokról van szó. Minden felhasználónak meg kell adnia a személyes adatait, beleértve az elérhetőségeit és egy biztonságos jelszót. A rendszer biztosítja, hogy egy felhasználónak mindig legyen érvényes szerepköre.

Navigációs menüpontok (NAVIGATIONS)  
{navigation\_id, name, URL, component\_name, parent}

A rendszer különböző funkcióit menüpontokon keresztül érhetik el a felhasználók. Minden menüpontnak van egy neve és egy URL-címe, amely meghatározza, hogy melyik oldalra vezet. Egyes menüpontok önállóan léteznek, míg mások egy főmenü alá tartoznak.

Menüpont-hozzáférési jogosultságok (NAVIGATION\_ROLES)

{navigationRole\_id, ranking, *role\_id*, *navigation\_id*}

Nem minden menüpont érhető el minden felhasználói szerepkör számára. Ebben a táblában van meghatározva, hogy mely szerepkörök milyen menüpontokat láthatnak és használhatnak. Így például a páciensek nem férhetnek hozzá az adminisztrációs funkciókhoz, míg az adminisztrátorok teljes hozzáféréssel rendelkeznek.

Páciensek (PATIENTS)  
{user\_id, taj\_number, birth\_date, address}  
Ez a tábla tartalmazza a klinikára regisztrált páciensek adatait. Minden páciens rendelkezik egy egyedi azonosítóval, amelyet a rendszerben való azonosításra használunk. Az egészségügyi ellátás biztosítása érdekében itt tároljuk a társadalombiztosítási számukat, a születési dátumukat és lakcímüket is.

Orvosi szakterületek (SPECIALIZATIONS)  
{specialization\_id, specialization\_name}  
A klinikán dolgozó orvosok különböző szakterületekre specializálódnak, például belgyógyászat, sebészet vagy ortopédia. Ebben a táblában vannak felsorolva ezek a szakterületek.

Orvosok (DOCTORS)  
{user\_id, *specialization\_id*}  
Az orvosok adatai ebben a táblában vannak tárolva. Minden orvos egyedi azonosítóval rendelkezik, amely a rendszer többi táblájában is használható azonosításra. Az orvosok csak olyan kezeléseket végezhetnek, amelyek a saját szakterületükhöz tartoznak.

Kezelések (TREATMENTS)  
{treatment\_id, *specialization\_id*, treatment\_name, treatment\_length, price}  
A klinika által nyújtott különböző kezelések listája ebben a táblában található. Minden kezeléshez meghatározott időtartam és ár tartozik, valamint egy szakirány, amely biztosítja, hogy csak a megfelelő szaktudással rendelkező orvosok végezhessék el az adott beavatkozást.

Orvosi időpontfoglalások (DOCTOR\_APPOINTMENTS)  
{*doctor\_id*, start\_time, *user\_id*, *treatment\_id*, status, rating}

A páciensek a rendszerben foglalhatnak időpontokat az orvosokhoz. Minden foglalás tartalmazza az érintett orvos és páciens azonosítóját, valamint az időpont kezdetét. A páciens azonosítója akkor kerül a táblába, ha a foglalás megtörténik, ezzel a státusza „v” (szabad). Az időpont státusza jelezheti, hogy az aktív, törölt vagy lezárt. A rendszer lehetőséget biztosít a kezelések értékelésére is, így a páciensek visszajelzést adhatnak az ellátás minőségéről.

### Egyed kapcsolat diagram

### Kapcsolati ábra

### Táblák szerkezete a tábla szintű megszorításokkal

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **SZEREP (ROLE)** | | | | |
| Mező magyarul | Mező valós | Adattípus | Szerep | Korlátozás |
| jogosultság azonosító | role\_id | bigint(20) | K |  |
| név | name | varchar(255) | kötelező, egyedi |  |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **FELHASZNÁLÓ (USER)** | | | | |
| Mező magyarul | Mező valós | Adattípus | Szerep | Korlátozás |
| azonosító | id | bigint(20) | K |  |
| név | name | varchar(255) | kötelező |  |
| telefonszám | phone\_number | varchar(255) | kötelező, egyedi |  |
| email | email | varchar(255) | kötelező, egyedi |  |
| jelszó | password | varchar(255) | kötelező |  |
| jogosultság azonosító | role\_id | bigint(20) | KK(Role) | role\_id > 0 & role\_id < role tála méret |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **SPECIALIZÁCIÓ (SPECIALIZATION)** | | | | |
| Mező magyarul | Mező valós | Adattípus | Szerep | Korlátozás |
| specializáció azonosító | specialization\_id | bigint(20) | K |  |
| specializáció név | specializaton\_name | varchar(255) | kötelező, egyedi |  |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **ORVOS (DOCTOR)** | | | | |
| Mező magyarul | Mező valós | Adattípus | Szerep | Korlátozás |
| felhasználó azonosító | user\_id | bigint(20) | K, KK (USER) |  |
| specializáció azonosító | specialization\_id | bigint(20) | KK (SPECIALIZATION) |  |

További megszorítások:

* Egy orvos csak egy szakkal rendelkezhez

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **PÁCIENS (PATIENT)** | | | | |
| Mező magyarul | Mező valós | Adattípus | Szerep | Korlátozás |
| felhasználó azonosító | user\_id | bigint(20) | K, KK (USER) |  |
| tajszám | taj\_number | int(11) | kötelező, egyedi | taj\_number > 0 |
| születési idő | birth\_date | date | kötelező | birth\_date > mai dátum |
| cím | address | varchar(255) | kötelező |  |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **KEZELÉS (TREATMENT)** | | | | |
| Mező magyarul | Mező valós | Adattípus | Szerep | Korlátozás |
| kezelés azonosító | treatment\_id | bigint(20) | K |  |
| specializáció azonosító | specialization\_id | bigint(20) | KK (SPECIALIZATION) |  |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **ORVOSI IDŐPONTFOGLALÁSOK (DOCTOR\_APPOINTMENT)** | | | | |
| Mező magyarul | Mező valós | Adattípus | Szerep | Korlátozás |
| orvos azonosító | doctor\_id | bigint(20) | K, KK része (DOCTOR) |  |
| kezdés ideje | start\_time | datetime | K | kezdés idő > mai nap |
| páciens azonosító | patient\_id | bigint(20) | KK (PATIENT) |  |
| kezelés azonosító | treatment\_id | bigint(20) | KK (TREATMENT) |  |
| státusz | status | enum | default vacant |  |
| értékelés | rating | int(11) |  | rating > 0 & rating < 6 |

További megszorítások:

* ugyanarra a kezelésre adott páciensnek egy 'booked'(foglalt) státuszú kezelése legyen
* ha a kezelés státusza nem "d", akkor nem lehetséges értékelni
* ugyanazon kezelésre adott páciens csak egy értékelést tudjon adni

### Fogalmak

### Hasznos lekérdezések

1. Adott orvosnak hány elvégzett kezelése volt az előző hónapban?

DECLARE @doctor\_id INT = 1;

SELECT COUNT(\*) AS completed\_treatments

FROM doctor\_appointments

WHERE doctor\_id = @doctor\_id

AND status = 'done'

AND start\_time >= DATEADD(MONTH, -1, DATEADD(DAY, 1 - DAY(GETDATE()), GETDATE()))

AND start\_time < DATEADD(DAY, 1 - DAY(GETDATE()), GETDATE());

1. Adott Páciensnek hány kezelése volt az előző hónapban?

DECLARE @taj\_number BIGINT = 123456789;

SELECT COUNT(\*) AS patient\_treatments

FROM doctor\_appointments

WHERE taj\_number = @taj\_number

AND status = 'done'

AND start\_time >= DATEADD(MONTH, -1, DATEADD(DAY, 1 - DAY(GETDATE()), GETDATE()))

AND start\_time < DATEADD(DAY, 1 - DAY(GETDATE()), GETDATE());

1. Hány Páciens regisztrált akinek volt elvégzett kezelése?

SELECT COUNT(DISTINCT p.taj\_number) AS registered\_patients

FROM patients p

JOIN doctor\_appointments da ON p.taj\_number = da.taj\_number

WHERE da.status = 'done';

1. Adott orvosnak hány törölt időpontja volt az adott hónapban/évben?
2. Melyik három orvos kapta a legtöbb értékelést tavaly?

SELECT TOP 3 d.doctor\_id, u.name, COUNT(\*) AS total\_ratings

FROM doctor\_appointments da

JOIN doctors d ON da.doctor\_id = d.doctor\_id

JOIN users u ON d.user\_id = u.id

WHERE YEAR(start\_time) = YEAR(DATEADD(YEAR, -1, GETDATE()))

AND da.rating IS NOT NULL

GROUP BY d.doctor\_id, u.name

ORDER BY total\_ratings DESC;

1. Melyik három orvos kapta a legrosszabb értékelést tavaly?

SELECT TOP 3 d.doctor\_id, u.name, AVG(da.rating) AS avg\_rating

FROM doctor\_appointments da

JOIN doctors d ON da.doctor\_id = d.doctor\_id

JOIN users u ON d.user\_id = u.id

WHERE YEAR(start\_time) = YEAR(DATEADD(YEAR, -1, GETDATE()))

AND da.rating IS NOT NULL

GROUP BY d.doctor\_id, u.name

ORDER BY avg\_rating ASC;

1. Mely userek nem lépett be a rendszerbe több mint 3 éve?

SELECT TOP 3 d.doctor\_id, u.name, AVG(da.rating) AS avg\_rating

FROM doctor\_appointments da

JOIN doctors d ON da.doctor\_id = d.doctor\_id

JOIN users u ON d.user\_id = u.id

WHERE YEAR(start\_time) = YEAR(DATEADD(YEAR, -1, GETDATE()))

AND da.rating IS NOT NULL

GROUP BY d.doctor\_id, u.name

ORDER BY avg\_rating ASC;

1. Mely szakon történt a legtöbb elvégzett vizsgálat/kezelés?

SELECT TOP 1 s.specialisation\_name, COUNT(\*) AS total\_treatments

FROM doctor\_appointments da

JOIN treatments t ON da.treatment\_id = t.treatment\_id

JOIN specialisations s ON t.specialisation\_id = s.specialisation\_id

WHERE da.status = 'done'

GROUP BY s.specialisation\_name

ORDER BY total\_treatments DESC;

1. Mely páciensek mondták le az időpontjaikat legalább 10 alkalommal tavaly?

SELECT p.taj\_number, u.name, COUNT(\*) AS cancelled\_count

FROM doctor\_appointments da

JOIN patients p ON da.taj\_number = p.taj\_number

JOIN users u ON p.user\_id = u.id

WHERE da.status = 'cancelled\_by\_patient'

AND YEAR(start\_time) = YEAR(DATEADD(YEAR, -1, GETDATE()))

GROUP BY p.taj\_number, u.name

HAVING COUNT(\*) >= 10;

1. Mely Vizsgálatok hozták a legtöbb pénzt adott évben vagy hónapban?

DECLARE @revenue\_year INT = 2024;

DECLARE @revenue\_month INT = NULL; -- Ha NULL, akkor az egész évre vonatkozik

SELECT TOP 5 t.treatment\_name, SUM(t.price) AS total\_revenue

FROM doctor\_appointments da

JOIN treatments t ON da.treatment\_id = t.treatment\_id

WHERE da.status = 'done'

AND YEAR(da.start\_time) = @revenue\_year

AND (@revenue\_month IS NULL OR MONTH(da.start\_time) = @revenue\_month)

GROUP BY t.treatment\_name

ORDER BY total\_revenue DESC;

### Beépítendő megszorítások

### Megvalósítás

### Scriptek

### Teszt adatok

### Továbbfejlesztési ötletek

## Backend

### Modellek és Controllerek

### API végpontok

## Frontend

## Reszponzivitás

.

# Tesztelés

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Az összes teszteset** | **Tesztelési egység** | | | |
|  | | | |
| Teszteset / Bemeneti paraméterek | Várt eredmény | Kapott eredmény | Státusz |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |

# Fejlesztési lehetőségek

# Irodalomjegyzék