

# Tattoo Pont Client

Web Technológiák II.  
beadandó feladat

**Készítette:** Martinák Mátyás - KLNSPG

Miskolc, 2024

# Tartalomjegyzék

<b>1. Bevezető</b>	<b>2</b>
<b>2. React Native struktúra és tartalmi felépítés</b>	<b>4</b>
2.1. Kezdőlap . . . . .	4
2.2. Validációs lap . . . . .	5
2.3. Felhasználási lap . . . . .	7
2.4. Ellenőrző gomb . . . . .	8
2.5. Érvényes kuponok kilistázása . . . . .	10
<b>3. Összefoglalás</b>	<b>12</b>

# 1. fejezet

## Bevezető

A mobilalkalmazások rohamos terjedése és az okostelefonok szinte állandó jelenléte az életünkben óriási lehetőségeket teremt a különféle üzleti modellek és szolgáltatások számára. Ebben a digitális korban, amikor a felhasználók egyre inkább kényelmi szempontok alapján döntenek, az ajándékutalványok és kuponok kezelésére szolgáló alkalmazások jelentősége kiemelkedően nőtt. Az ilyen típusú alkalmazások nem csupán a vásárlói élmény javítását szolgálják, hanem lehetővé teszik a vállalkozások számára is, hogy pontosabb képet kapjanak ügyfeleik viselkedéséről és preferenciáiról.

A „Tattoo Pont Client” alkalmazás fejlesztése során az Expo keretrendszer alkalmazásával egy olyan mobilalkalmazást hoztam létre, amely képes a felhasználók igényeit magas szinten kiszolgálni, miközben a fejlesztési folyamatot jelentősen leegyszerűsíti és felgyorsítja. Az Expo platform előnye, hogy egy közös kódbázison keresztül lehet natív alkalmazásokat fejleszteni iOS és Android platformokra egyaránt, ezáltal jelentősen csökkentve a fejlesztési és karbantartási költségeket.

Az alkalmazás ajándékkupon generáló részének fejlesztése során a C# nyelvet és a .NET keretrendszert választottam, ami lehetővé tette számomra, hogy biztonságos, skálázható és megbízható backend szolgáltatást hozzak létre. A C# programom az ajándékutalványok generálását és azok adatbázisban való tárolását végzi. A MongoDB-t választottam adatbázis-kezelő rendszernek, mivel annak dokumentum-orientált jellege kiválóan alkalmas a dinamikus és strukturálatlan adatok kezelésére, mint amilyenek az ajándékutalványok adatai is.

A generált ajándékutalványok QR kódokat tartalmaznak, amelyek egyedi azonosítót (`ObjectId`) hordoznak, így biztosítva az utalványok egyediségét és könnyen nyomon követhetőségét. Az `IsValid` és `CreationDate` mezők további funkciókat látnak el az utalványok kezelésében, például a validáció során az `IsValid` mező segítségével könnyedén megállapítható, hogy az utalvány

érvényes-e még, míg a `CreationDate` információkat szolgáltat az utalvány kibocsátásának idejéről, és ezt felhasználva azonosíthatja a felhasználó az ajándékkupon érvényességét is.

Az alkalmazás különlegessége abban rejlik, hogy a modern technológiák és a felhasználóbarát interfész ötvözésével egy olyan platformot hoz létre, amely nemcsak a végfelhasználók számára kínál kényelmet és egyszerűséget, hanem a vállalkozások számára is értékes adatokat szolgáltat az ügyfelek viselkedéséről. Az alkalmazás így nem csak egy technológiai megoldás, hanem egy üzleti eszköz is, amely hozzájárul a vásárlói lojalitás növeléséhez és az üzleti döntések megalapozásához.



1.1. ábra. Egy kigenerált ajándékkártya

## 2. fejezet

# React Native struktúra és tartalmi felépítés

A bevezető után, nézzük át a React Native mobilapplikáció struktúráját, mely az egyes kuponok érvényesítésért és felhasználásáért felel.

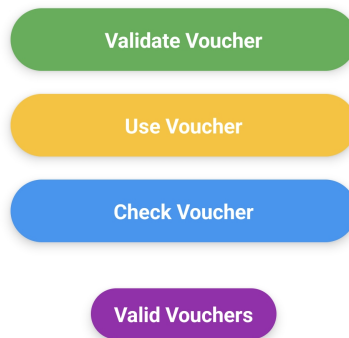
### 2.1. Kezdőlap

A *Tattoo Pont Client* alkalmazás kezdőlapja egy React Native alapú mobilalkalmazás felhasználói interfészét valósítja meg. Az alkalmazás a modern mobilfejlesztési keretrendszer előnyeit kihasználva biztosít cross-platform kompatibilitást és felhasználóbarát dizájnt.

A kezdőlap felépítése a következő elemekből áll:

- **Fő konténer:** Egy `View` komponens, amely az alkalmazás kezdőképnyőjének alapját képezi, és központosítja a felhasználói felület elemeit.
- **Címsor:** Egy `Text` komponens, amely kiemeli az alkalmazás nevét, segítve ezzel a felhasználók azonosítási folyamatát.
- **Gombok:** `TouchableOpacity` komponensek, amelyek a különböző funkciók elérését teszik lehetővé, mint például a kuponok érvényesítése, felhasználása, ellenőrzése és az érvényes kuponok listájának megtekintése.
- **Navigációs logika:** A `useNavigation` hook segítségével implementált logika, amely lehetővé teszi a felhasználók számára, hogy az alkalmazás különböző képernyőin között navigáljanak.

## Tattoo Pont Client



2.1. ábra. Kezdőlap

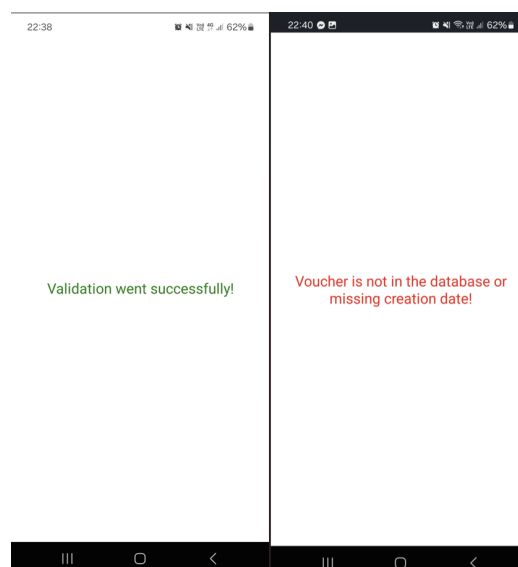
A kezdőlap vizuális stílusát a `StyleSheet.create` módszerrel definiált stílusok határozzák meg, amelyek az alkalmazás vizuális megjelenésének minden aspektusát lefedik, beleértve a színsémát, a gombok formáját, a szöveg méretét és súlyát, valamint az árnyékolást és kiemelést.

## 2.2. Validációs lap

A szoftver kezdőlapján található első gomb segítségével érhető el a „Validation” nevű osztály, amely az ajándékutalványok érvényesítésére szolgál. Ez a funkció kulcsfontosságú eleme az alkalmazásnak, lehetővé téve a felhasználók számára, hogy QR kódokat szkenneljenek és ellenőrizzék azokat egy központi adatbázissal való kommunikáció során.

A „Validation” osztály fő jellemzői és működése a következőképpen írható le:

- A kezdőlapon található első gombra kattintva a felhasználókat átirányítjuk a „Validation” osztályra.
- A vonalkód-szkennelési folyamatot az `expo-barcode-scanner` könyvtár `BarCodeScanner` komponense végzi, amely integrálva van az Expo keretrendszerbe. Ez a komponens hozzáférést biztosít a készülék kamerájához, lehetővé téve a QR kódok gyors és hatékony beolvasását.
- Miután a QR kód sikeresen beolvasásra került, a „Validation” osztály egy API-n keresztül kommunikál egy adatbázissal, hogy ellenőrizze, a szkennelt kupon szerepel-e az adatbázisban. Ez a lépés biztosítja, hogy csak érvényes és az adatbázisban regisztrált kuponok kerüljenek érvényesítésre.
- Amennyiben a kupon megtalálható az adatbázisban, a rendszer ellenőrzi annak érvényességét. Ha a kupon még nem került felhasználásra, az osztály végrehajtja az érvényesítési folyamatot. Ellenkező esetben a felhasználót tájékoztatja a kupon érvénytelenségéről vagy korábbi felhasználásáról.
- Az érvényesítési folyamat eredményéről a felhasználók visszajelzést kapnak egy felugró üzenet formájában, amely tájékoztatja őket a folyamat sikerességéről vagy esetleges hibáiról. Az üzenet színe (zöld sikeres érvényesítés, piros hiba esetén) vizuális visszajelzést is nyújt.



2.2. ábra. Sikeres és sikertelen érvényesítés

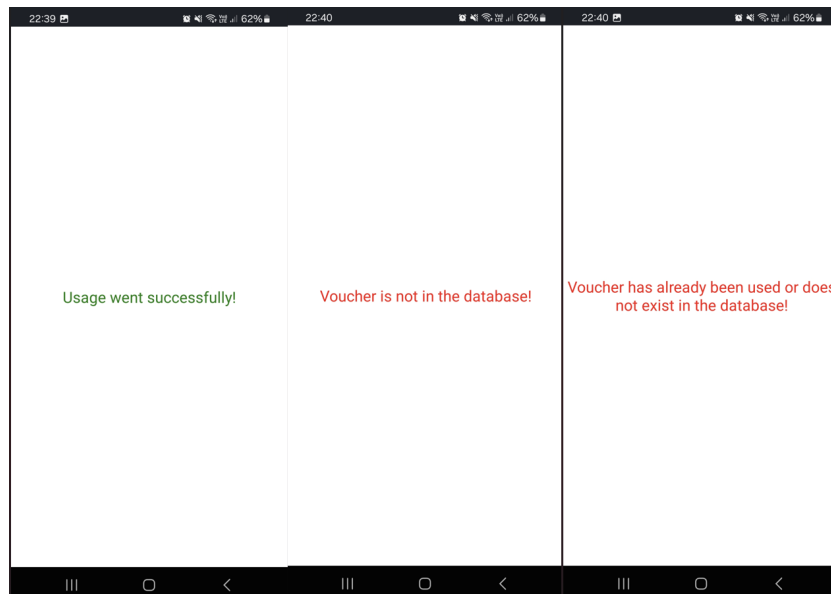
## 2.3. Felhasználási lap

A „Use” osztály egy kulcsfontosságú komponens a mobilalkalmazásban, amely a kuponok felhasználásának logikáját valósítja meg. Az osztály felelős a QR kódok beolvasásáért és azok ellenőrzéséért, hogy a kuponok érvényesek-e, és nem kerültek-e már felhasználásra.

A főbb jellemzők és működés a következők:

- **Kamera használati engedély:** A komponens a `useEffect` hook segítségével kéri és ellenőrzi a kamera használati engedélyét. Amennyiben az engedély megadásra kerül, lehetővé teszi a QR kódok beolvasását. Ellenkező esetben tájékoztatja a felhasználót az engedély hiányáról.
- **QR kód beolvasása:** A `BarCodeScanner` komponens felhasználásával az osztály folyamatosan várja a QR kódok beolvasását. Amikor egy QR kód beolvasásra kerül, a `handleBarCodeScanned` függvény aktiválódik.
- **Adatbázis ellenőrzés:** A beolvasott QR kód alapján az osztály egy API híváson keresztül ellenőrzi a kupon érvényességét egy adatbázisban. Amennyiben a kupon érvénytelen, már felhasznált, vagy az adatbázisban nem található, a rendszer hibaüzenetet jelenít meg.
- **Kupon felhasználása:** Ha a kupon érvényes és nem került még felhasználásra, az osztály elvégzi a szükséges lépéseket annak felhasználására. Ez magában foglalhatja a kupon adatbázisból történő törlését, ezzel jelezve, hogy a kupon már felhasznált.
- **Felhasználói visszajelzés:** A folyamat eredményétől függően a felhasználók visszajelzést kapnak egy szöveges üzenet formájában, amely tájékoztatja őket a művelet sikerességéről vagy esetleges hibáiról. Az üzenet színe (zöld sikeres felhasználás, piros hiba esetén) segít a felhasználóknak az eredmény azonnali értelmezésében.





2.3. ábra. Sikeres és sikertelen felhasználási kísérletek

Összefoglalva, a „Use” osztály egy esszenciális része az alkalmazásnak, amely lehetővé teszi a felhasználók számára, hogy hatékonyan használják fel a QR kód alapú kuponjaikat. Az osztály biztosítja a kuponok érvényességének ellenőrzését, a felhasználás folyamatának kezelését, valamint a felhasználók tájékoztatását az eredményről.

## 2.4. Ellenőrző gomb

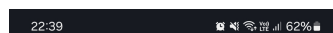
A „Check” osztály egy fontos komponens az alkalmazásban, amelynek feladata a QR kódok alapján történő kuponok ellenőrzése. Ez a folyamat magában foglalja a kupon adatbázisban való létezésének és az érvényességi időszakának ellenőrzését.

A főbb jellemzők és működés a következők:

- **Kamera használati engedély:** Az osztály a `useEffect` hook segítségével kéri a kamera használati engedélyét a felhasználótól. Amennyiben az engedélyt megadják, az alkalmazás képes lesz a QR kódok beolvasására.
- **QR kód beolvasása:** A `BarCodeScanner` komponens segítségével az osztály folyamatosan várja a QR kódok beolvasását. Beolvasás után a `handleBarCodeScanned` függvény kerül aktiválásra.

- **Adatbázis ellenőrzés:** A QR kód tartalmának felhasználásával az osztály egy API hívás segítségével lekéri a kupon adatait az adatbázisból. Ellenőrzi a kupon létezését, valamint annak létrehozási és lejárat dátumát.
- **Lejárat idő számítása:** Amennyiben a kupon létezik az adatbázisban, az osztály kiszámítja a lejárat dátumot a létrehozási idő alapján, majd összehasonlítja azt a jelenlegi dátummal, hogy megállapítsa a kupon érvényességét.
- **Felhasználói visszajelzés:** Az ellenőrzési folyamat eredményétől függően a felhasználók egy üzenetet kapnak, amely tájékoztatja őket a kupon érvényességéről vagy annak hiányáról. Az üzenet színe (zöld, ha a kupon érvényes; piros, ha a kupon érvénytelen vagy nem található az adatbázisban) az eredménytől függ.

A „Check” osztály tehát lehetővé teszi a felhasználók számára, hogy gyorsan és hatékonyan ellenőrizzék a kuponjaik érvényességét. Ez a folyamat növeli az alkalmazás használati értékét, biztosítva a felhasználók számára egy megbízható eszközt a kuponok kezelésére.



Checking went successfully!  
 Voucher expires in 366 day(s)  
 Expiry date: Mon Feb 24 2025 22:38:54  
 GMT+0100.



2.4. ábra. Ellenőrzési folyamat visszajelzése

## 2.5. Érvényes kuponok kilistázása

A "VoucherList" osztály egy fontos komponens az alkalmazásban, amely a felhasználóknak lehetővé teszi az érvényes kuponok aktuális számának megtekintését. Az osztály dinamikusan frissíti ezt az információt, biztosítva a legfrissebb adatok elérhetőségét.

A főbb jellemzők és működés a következők:

- **Állapotkezelés:** A komponens a `useState` hook segítségével kezeli az érvényes kuponok számát (`couponCount`) és a frissítés állapotát (`refreshing`).
- **Adatlekérés:** Az `useEffect` hook használatával, a komponens betöltésekor automatikusan meghívódik a `fetchCouponCount` függvény, amely lekéri az érvényes kuponok számát egy adatbázisból egy API hívás segítségével.
- **Frissítési logika:** A `ScrollView` komponens `RefreshControl` eleme lehetővé teszi a felhasználók számára, hogy lehúzással frissítsék az érvényes kuponok számát. Ez a folyamat újraindítja az `fetchCouponCount` függvényt.
- **Felhasználói felület:** A komponens egy gördíthető nézetet (`ScrollView`) használ, amely középen jeleníti meg az érvényes kuponok számát egy szöveges elem (`Text`) segítségével. A dizájn egyszerű, de informatív, biztosítva a felhasználók számára a könnyű érthetőséget és hozzáférést.



Number of valid vouchers: 3



2.5. ábra. Érvényes kuponok száma

Az osztály tehát egy interaktív és felhasználóbarát módot biztosít az érvényes kuponok nyomon követésére, lehetővé téve a felhasználók számára, hogy naprakészek maradjanak az aktuális kuponkínálattal kapcsolatban. A frissítési funkció növeli az alkalmazás használhatóságát, ösztönözve a felhasználókat arra, hogy aktívan használják az alkalmazást a kuponok kezelésére.

## 3. fejezet

# Összefoglalás

Az előző fejezetekben részletesen bemutatam a „Tattoo Pont Client” mobilalkalmazás tervezési és fejlesztési folyamatát, az alkalmazott technológiai stacket, valamint a különböző funkcionális modulokat. Az Expo keretrendszer felhasználása lehetővé tette számomra, hogy egyetlen kódbázissal fejlesszek alkalmazást mind iOS, mind Android platformokra, ami jelentős idő- és költségmegtakarítást eredményezett a fejlesztési ciklus során.

A backend szolgáltatások implementálásához a C# programozási nyelvet és a .NET keretrendszert választottam, amelyek skálázhatóságot, megbízhatóságot és magas szintű biztonságot kínálnak az adatkezelési folyamatok során. A MongoDB dokumentum-orientált adatbázis-kezelő rendszer használata optimális választásnak bizonyult a kuponokhoz kapcsolódó dinamikus adatstruktúrák tárolására és kezelésére.

A mobilalkalmazás kritikus funkcióit – mint például a kuponok érvényesítése (*Validation*), felhasználása (*Use*), ellenőrzése (*Check*) és az érvényes kuponok listázása (*VoucherList*) – részletesen elemeztem. Minden egyes funkció gondos tervezésen és implementáción esett át, hogy biztosítsa a felhasználói interakciók zökkenőmentességét és hatékonyságát. Kiemeltem továbbá az egyedi QR kódok generálásának és kezelésének jelentőségét, amely kulcsfontosságú szerepet játszik az ajándékutalványok egyediségének és nyomon követhetőségének biztosításában.

Az alkalmazás fejlesztése során folyamatosan törekedtem a felhasználói élmény optimalizálására és a GUI felület intuitív tervezésére. A modern UI/UX elvek alkalmazásával és a felhasználói visszajelzések figyelembe vételével egy olyan mobilalkalmazást hoztam létre, amely nem csak technológiai szempontból fejlett, hanem a végfelhasználók számára is vonzó és könnyen kezelhető.

Összességében a „Tattoo Pont Client” alkalmazás a modern webtechnológiák és a felhasználó-központú tervezés ötvözeteként jött létre, ami nem csak a felhasználói igényeket elégíti ki, hanem új lehetőségeket is nyit a vál-

lalkozások számára az ügyfélkapcsolatok menedzselésében és az ügyféladatok elemzésében. Az alkalmazás így hozzájárul a digitális kereskedelmi stratégiák fejlődéséhez, elősegítve a vállalkozások növekedését és versenyképességének erősítését a digitális korban.