

UI Dokumentáció

Az UI project 5 file-ből épül fel, ezek:

1. Program.cs

A main függvénye itt található a programnak és a belépés / regisztráció is itt van kezelve.

2. menuDisplays.cs

A menük statikus részeinek kiírása itt található.

3. helperFunctions.cs

Segéd függvényeket tárol, főként kiírással vagy beolvasással kapcsolatosak.

4. displayConfig.cs

Egy konfig file ami statikus összegeket tárol

5. userSession.cs

A programban a felhasználó összes akcióját itt kezeli a program belépés után.

Ezekben a fileokban találhatóak a program futásához szükséges függvények, ezek leírásai:

1. Program.cs:

```
public static async Task Main(string[] args)
```

A program fő ciklusát tartalmazza, ez hívja meg a függvényeket a program inicializálásához és indítja el a user input kezelését.

```
private static async Task<User?> handleRegistration(UserRepository users)
```

A programban ez a függvény kezeli a felhasználói regisztrációt, bekéri az adatokat, elmenti őket és esetlegesen hibát is kezel.

```
private static async Task<User?> handleLogin(UserRepository users)
```

A programban ez a függvény kezeli a felhasználók belépését, bekéri az adatokat, ellenőrzi őket és visszaad egy user objektumot, ha sikeres a belépés.

2. menuDisplays.cs:

```
public static void displayLoginMenu()
```

Ez a függvény jeleníti meg a belépési menüt, statikus stringeket ír csak ki.

```
public static void displayMenu(User user)
```

Ez a függvény dönti el, hogy a felhasználónak az admin vagy a felhasználói menüt kell-e megjeleníteni, majd meghívja a hozzá tartozó függvényt.

```
private static void displayGuestMenu()
```

Ez a függvény jeleníti meg a felhasználói menüt, statikus stringeket ír csak ki.

```
private static void displayAdminMenu()
```

Ez a függvény jeleníti meg az admin menüt, statikus stringeket ír csak ki.

```
public static void displayDividerLine(int length, char character)
```

Ez a függvény kirajzol egy megadott hosszúságú sort, megadott karakterből.

```
public static void displayDynamicDividerLine(string input, char character)
```

Ez a függvény kirajzol egy bemeneti string hosszúságú sort, megadott karakterből.

3. helperFunctions.cs:

`public static int readMenuChoice()`

Ez a függvény feldolgozza a felhasználó menü választását.

`public static string Truncate(string value, int maxLength)`

Ez a függvény levágja egy bemeneti string végét maxLength hosszra.

`public static void DisplayError(string message)`

Ez a függvény kiír egy formázott bemeneti üzenetet.

`public static void DisplaySuccess(string message)`

Ez a függvény kiír egy formázott bemeneti üzenetet.

`public static bool DisplayInvalidChoice()`

Ez a függvény kiír egy formázott üzenetet.

4. displayConfig.cs

Ez a file konstansokat tárol, amik használva vannak több helyen a programban.

5. userSession.cs

```
public static async Task handleUserSession(BookingService  
bookingService, User user)
```

Ez a függvény a fő felhasználói ciklust tartalmazza.

```
private static async Task<bool> handleAdminMenuChoice(int choice,  
BookingService bookingService, User user)
```

Ez a függvény az admin menü választást kezeli és meghívja a megfelelő függvényeket.

```
private static async Task<bool> handleGuestMenuChoice(int choice,  
BookingService bookingService, User user)
```

Ez a függvény az felhasználó menü választást kezeli és meghívja a megfelelő függvényeket.

```
private static bool TryParseDate(string? input, string prompt, out DateTime  
result)
```

Ez a függvény felhasználó által beírt dátumot parseol.

```
private static bool TryParseInt(string? input, string fieldName, out int result,  
int minValue = int.MinValue)
```

Ez a függvény felhasználó által beírt egész számot parseol.

```
private static bool TryParseFloat(string? input, string fieldName, out float  
result)
```

Ez a függvény felhasználó által beírt tizedes számot parseol.

```
private static bool ValidateDateRange(DateTime from, DateTime to, string context = "")
```

Ez a függvény felhasználó által megadott időintervallumot ellenőriz le.

```
private static void DisplayEnumOptions<T>(string prompt)
```

Ez a függvény egy enum változót írja ki a felhasználónak.

```
private static bool TryParseEnum<T>(string? input, string fieldName, out T result)
```

Ez a függvény felhasználó által beírt enum-ot parseol.

```
private static async Task<bool?> handleCreateBooking(BookingService bookingService, User user)
```

Ez a függvény a felhasználót végig vezeti egy foglalás létrehozásán.

```
private static async Task<bool?> displayUserBookings(BookingService bookingService, User user)
```

Ez a függvény kiírja a felhasználó összes foglalását.

```
private static async Task<bool?> handleModifyBooking(BookingService bookingService, User user)
```

Ez a függvény végig vezeti a felhasználót egy foglalás módosításán.

```
private static async Task<bool?> handleDeleteBooking(BookingService bookingService, User user)
```

Ez a függvény egy foglalást töröl a rendszerből, admin felhasználónak enged bármilyen foglalást törölni, sima felhasználónak pedig csak a saját foglalásait engedi törölni.

```
private static async Task<bool?> displayPlaces(BookingService  
bookingService)
```

Ez a függvény kiírja az összes helyet a felhasználónak.

```
private static async Task<bool?>  
handleAdminManageBooking(BookingService bookingService, User  
admin)
```

Ez a függvény az admin felhasználónak kezeli a foglalások kezelését, módosító vagy törlő függvényeket hív meg.

```
private static async Task<bool?> handleAddPlace()
```

Ez a függvény végig vezeti az admin felhasználót egy hely létrehozásán.

```
private static async Task<bool?> handleModifyPlace()
```

Ez a függvény végig vezeti az admin felhasználót egy hely módosításán.

```
private static async Task<bool?> handleAddSeasonalRule()
```

Ez a függvény végig vezeti az admin felhasználót egy szezonális szabály létrehozásán.

```
private static async Task<bool?> handleModifySeasonalRule()
```

Ez a függvény végig vezeti az admin felhasználót egy szezonális szabály módosításán.

```
private static List<hazifeladat.DAL1.Models.Enums.PlaceTypes>?  
ParsePlaceTypesInput(string? input)
```

Ez a függvény feldolgozza, hogy a helyi szabályok milyen típusú helyekre vonatkozzanak.

```
private static async Task<bool?> displaySeasonalRules()
```

Ez a függvény kiírja az összes szezonális szabályt.

```
private static async Task<bool?> displayAllBookings(BookingService  
bookingService)
```

Ez a függvény kiírja az összes foglalást.