

Foglalások funkció működés

PHP forrásfájlok

Foglalások létrehozása

```
<?php
// Betöltöm az adatbázis-kapcsolatot biztosító fájlt.
require_once "kapcsolat.php";

// Beállítom, hogy JSON formátumban küldöm vissza a választ.
header('Content-Type: application/json');

// Beolvasom a JSON bemenetet, amit a kliens küldött.
$input = json_decode(file_get_contents("php://input"), true);

// Ellenőrzöm, hogy megkaptam-e a szükséges mezőket.
if (!isset($input['felhasznalo_id'], $input['idopont_id'])) {
    http_response_code(400);
    echo json_encode(['error' => 'Hiányzó mezők']);
    exit;
}

try {
    // Elindítom az adatbázis tranzakciót, hogy egyszerre tudjak több műveletet.
    biztonságosan végrehajtani
    $pdo->beginTransaction();

    // Létrehozok egy új foglalást az adatbázisban.
    $stmt = $pdo->prepare("INSERT INTO foglalasok (felhasznalok_id, idopontok_id,
    állapot, megjegyzes)
                                VALUES (:fid, :iid, 'lefoglalt', :msg)");
    $stmt->execute([
        ':fid' => $input['felhasznalo_id'],          // A bejelentkezett felhasználó
ID-je
        ':iid' => $input['idopont_id'],              // A kiválasztott időpont ID-je
        ':msg' => $input['megjegyzes'] ?? null       // Opcionális megjegyzés
    ]);

    // A kiválasztott időpontot már nem lehet újra lefoglalni, ezért frissítem az
    állapotát.
    $pdo->prepare("UPDATE idopontok SET foglalhato = 0 WHERE id = :iid")
        ->execute([':iid' => $input['idopont_id']]);

    // Ha minden sikerült, elkötelezem a tranzakciót
    $pdo->commit();

    // Sikeres foglalás visszajelzése JSON-ben.
```

Szakmai Gyakorlat I. – Frontend – Backend integrációs szakasz

Orosz Kristóf – EYZWG9

```
        echo json_encode(['status' => 'Sikeres foglalás']);
    } catch (PDOException $e) {
        // Ha bármilyen hiba történik, visszavonom a tranzakciót
        $pdo->rollBack();
        http_response_code(500);
        echo json_encode(['error' => $e->getMessage()]);
    }
```

Foglalások módosítása

```
<?php
// Betöltöm az adatbázis kapcsolatot.
require_once "kapcsolat.php";

// Beállítom, hogy JSON formátumban válaszoljon a szerver.
header('Content-Type: application/json');

// Ellenőrzöm, hogy érkezett-e érvényes ID a lekérdezéshez.
if (!isset($_GET['id']) || !ctype_digit($_GET['id'])) {
    http_response_code(400);
    echo json_encode(['error' => 'Hiányzó vagy hibás foglalás ID']);
    exit;
}

// Beolvasom a JSON-bemenetet és tömbbé alakítom.
$input = json_decode(file_get_contents('php://input'), true);

// Előkészítem a módosítandó mezőket és paramétereket.
$fields = [];
$params = ['id' => $_GET['id']];

// Ha az "allapot" mező szerepel a kérésben, hozzáadom a frissítéshez.
if (isset($input['allapot'])) {
    $fields[] = 'allapot = :a';
    $params['a'] = $input['allapot'];
}

// Ha a "megjegyzes" mező is szerepel, azt is hozzáadom.
if (isset($input['megjegyzes'])) {
    $fields[] = 'megjegyzes = :m';
    $params['m'] = $input['megjegyzes'];
}

// Ha egyik módosítható mező sincs megadva, hibaüzenetet küldök.
if (empty($fields)) {
    http_response_code(400);
    echo json_encode(['error' => 'Nincs módosítandó mező']);
    exit;
}
```

```
}

// Összeállítom a dinamikus SQL lekérdezést.
$sql = "UPDATE foglalasok SET " . implode(', ', $fields) . " WHERE id = :id";

try {
    // Elküldöm az SQL frissítést az adatbázisnak.
    $stmt = $pdo->prepare($sql);
    $stmt->execute($params);

    // Sikeres válasz JSON-ben.
    echo json_encode(['status' => 'Foglalás módosítva']);
} catch (PDOException $e) {
    // Ha valami hiba történik, visszajelzek róla
    http_response_code(500);
    echo json_encode(['error' => $e->getMessage()]);
}
```

Foglalások törlése

```
<?php
// Először betöltöm az adatbáziskapcsolatot.
require_once "kapcsolat.php";

// Beállítom, hogy JSON válaszokat küldjek vissza.
header('Content-Type: application/json');

// Ellenőrzöm, hogy kaptam-e érvényes foglalás ID-t a GET paraméterek között.
if (!isset($_GET['id']) || !ctype_digit($_GET['id'])) {
    http_response_code(400); // Hibás kérés
    echo json_encode(['error' => 'Érvénytelen foglalás ID']);
    exit;
}

// Megpróbálom törölni a megadott ID-jú foglalást az adatbázisból.
try {
    $stmt = $pdo->prepare("DELETE FROM foglalasok WHERE id = :id");
    $stmt->execute([':id' => $_GET['id']]);

    // Ha sikerült, visszajelzem JSON-ben.
    echo json_encode(['status' => 'Foglalás törölve']);
} catch (PDOException $e) {
    // Ha hiba történik az adatbázisművelet közben, hibakódot küldök vissza.
    http_response_code(500);
    echo json_encode(['error' => $e->getMessage()]);
}
```

HTML forrásfájlok

Foglalások létrehozása

```
<!DOCTYPE html>
<html lang="hu">
<head>
  <!-- **Dokumentum típusa és karakterkészlet beállítása** -->
  <meta charset="UTF-8" />
  <!-- **Reszponzív nézet beállítása** -->
  <meta name="viewport" content="width=device-width, initial-scale=1.0" />
  <!-- **Navigációs menü stíluslap betöltése** -->
  <link rel="stylesheet" href="menu.css" />
  <!-- **Oldal címe** -->
  <title>Foglalás létrehozása</title>
  <style>
    /* **Oldal alapstílusai** */
    body {
      font-family: Arial, sans-serif;
      background: #f5f5f5;
      margin: 0;
      padding: 0;
    }
    /* **Űrlap konténer középre igazítása és stílus** */
    .container {
      max-width: 500px;
      margin: 40px auto;
      background: white;
      padding: 30px;
      border-radius: 8px;
      box-shadow: 0 2px 8px rgba(0,0,0,0.1);
    }
    /* **Címsor stílusa** */
    h2 {
      text-align: center;
      color: black;
    }
    /* **Űrlapmezők címkéinek stílusa** */
    label {
      display: block;
      margin-top: 15px;
      font-weight: bold;
    }
    /* **Beviteli mezők és szövegterület** */
    input,
    textarea {
      width: 100%;
```

```
padding: 10px;
margin-top: 5px;
border: 1px solid #ccc;
border-radius: 4px;
}
/* **Szövegterület méretezése** */
textarea {
  resize: vertical;
  min-height: 80px;
}
/* **Küldőgomb stílusa** */
button {
  margin-top: 20px;
  width: 100%;
  padding: 12px;
  background-color: #28a745;
  color: white;
  border: none;
  border-radius: 5px;
  cursor: pointer;
}
/* **Gomb hover állapota** */
button:hover {
  background-color: #218838;
}
/* **Figyelmeztető szöveg stílusa** */
.color {
  color: red;
}
/* **Rejtett elem osztály** */
.hidden {
  display: none !important;
}
/* **Reszponzív stílusok (768px alatt)** */
@media (max-width: 768px) {
  .container {
    width: 90%;
    padding: 20px;
    margin: 30px auto;
  }
  h2 {
    font-size: 24px;
  }
  label,
  input,
  textarea,
  button {
    font-size: 15px;
  }
}
```

```

    }
    button {
        padding: 10px;
    }
}
/* **Mobil nézet (480px alatt)** */
@media (max-width: 480px) {
    h2 {
        font-size: 20px;
    }
    label,
    input,
    textarea,
    button {
        font-size: 14px;
    }
    button {
        padding: 10px;
    }
}
</style>
</head>
<body>
    <!-- **Fejléc és navigáció** -->
    <header>
        <a href="index.html" class="logo">Időpontfoglaló <span id="felhasznalo-
nev"></span></a>
        <nav>
            <a href="szolgaltatok.html">Szolgáltatók</a>
            <a href="idopontok.html">Időpontok</a>
            <a href="ertekelesek.html">Értékelések</a>
            <a href="bejelentkezes.html" id="nav-login"><b>Bejelentkezés</b></a>
            <a href="fiokom.html" id="nav-account" class="hidden">Fiókom</a>
            <a href="#" id="nav-logout" onclick="kijelentkezes()" class="hidden"><span
class="color">Kijelentkezés</span></a>
        </nav>
    </header>

    <!-- **Űrlap előkészítő szkript: felhasználó ellenőrzése és navigáció** -->
    <script>
        const user = JSON.parse(sessionStorage.getItem('felhasznalo'));
        if (!user || !user.id) {
            alert("Csak bejelentkezett felhasználó tud foglalni.");
            window.location.href = "bejelentkezes.html";
        }
        if (user.nev) {
            document.getElementById('felhasznalo-nev').textContent = ` - ${user.nev}`;
        }
    </script>

```

```

document.getElementById('nav-login').classList.add('hidden');
document.getElementById('nav-account').classList.remove('hidden');
document.getElementById('nav-logout').classList.remove('hidden');
</script>

<!-- **Foglalás megerősítése űrlap** -->
<h2>Foglalás megerősítése</h2>
<div class="container">
  <form id="foglalas-form">
    <!-- **Dátum mező (csak olvasható)** -->
    <label>Dátum:</label>
    <input type="text" id="datum" readonly />

    <!-- **Idő mező (csak olvasható)** -->
    <label>Idő:</label>
    <input type="text" id="ido" readonly />

    <!-- **Megjegyzés mező (opcionális)** -->
    <label for="megjegyzes">Megjegyzés (opcionális):</label>
    <textarea id="megjegyzes" placeholder="Pl. Kérem, érkezzen pontosan..."></textarea>

    <!-- **Beküldés gomb** -->
    <button type="submit">Foglalás megerősítése</button>
  </form>
</div>

<script>
  // **URL paraméterek olvasása**
  const params = new URLSearchParams(window.location.search);
  const idopontId = params.get('id');
  if (!idopontId) {
    alert("Hiányzó időpont azonosító.");
    window.location.href = "idopontok.html";
  }

  // **Időpont adatok lekérése és beállítása**
  fetch("idopontok_listazasa.php")
    .then(res => res.json())
    .then(data => {
      if (data.status !== 'siker') throw new Error("Időpontok betöltése sikertelen.");
      const rec = data.data.find(r => String(r.id) === idopontId);
      if (!rec) {
        alert("Nincs ilyen időpont.");
        window.location.href = "idopontok.html";
      }
      document.getElementById('datum').value = rec.datum;
    })

```

```
        document.getElementById('ido').value = rec.ido;
    })
    .catch(err => {
        console.error(err);
        alert(err.message);
        window.location.href = "idopontok.html";
    });

// **Űrlap beküldése AJAX-szal**
document.getElementById('foglalas-form').addEventListener('submit',
function(e) {
    e.preventDefault();
    const megj = document.getElementById('megjegyzes').value.trim();

    fetch("Foglalasok_letrehozasa.php", {
        method: "POST",
        headers: { "Content-Type": "application/json" },
        body: JSON.stringify({
            felhasznalo_id: user.id,
            idopont_id: parseInt(idopontId),
            megjegyzes: megj || null
        })
    })
    .then(res => res.json())
    .then(data => {
        if (data.status) {
            alert("Foglalás sikeresen létrehozva.");
            window.location.href = "idopontok.html";
        } else {
            alert("Hiba: " + (data.error || "Ismeretlen hiba"));
        }
    })
    .catch(err => {
        console.error(err);
        alert("Hiba a foglalás során.");
    });
});

// **Kijelentkezés funkció**
function kijelentkezés() {
    sessionStorage.removeItem('felhasznalo');
    window.location.href = "index.html";
}
</script>
</body>
</html>
```


Foglalások módosítása

```
<!DOCTYPE html>
<html lang="hu">

<head>
  <!-- Dokumentum típusa és karakterkészlet beállítása -->
  <meta charset="UTF-8" />
  <!-- Reszponzív nézet beállítása -->
  <meta name="viewport" content="width=device-width, initial-scale=1.0" />
  <!-- Navigációs menü stíluslap betöltése -->
  <link rel="stylesheet" href="menu.css" />
  <!-- Oldal címe -->
  <title>Foglalás módosítása</title>
  <style>
    /* Alapoldal stílusok */
    body {
      font-family: Arial, sans-serif;
      background-color: #f9f9f9;
      margin: 0;
      padding: 0;
    }
    /* Láthatatlan osztály */
    .hidden {
      display: none !important;
    }
    /* Űrlap tároló stílusai */
    .container {
      max-width: 500px;
      margin: 40px auto;
      background: white;
      padding: 30px;
      border-radius: 8px;
      box-shadow: 0 2px 8px rgba(0,0,0,0.1);
    }
    /* Címsor stílusa */
    h2 {
      text-align: center;
      color: black;
      margin-bottom: 20px;
    }
    /* Űrlapmező címkék */
    label {
      display: block;
```

```
    margin-top: 15px;
    font-weight: bold;
}
/* Beviteli mezők és szövegterület stílusa */
input,
textarea {
    width: 100%;
    padding: 10px;
    margin-top: 5px;
    border: 1px solid #ccc;
    border-radius: 4px;
    box-sizing: border-box;
}
/* Gomb stílusok */
button {
    margin-top: 20px;
    width: 100%;
    padding: 12px;
    background-color: #007BFF;
    color: white;
    border: none;
    border-radius: 5px;
    cursor: pointer;
}
/* Gomb hover állapota */
button:hover {
    background-color: #0056b3;
}
/* Reszponzív (768px alatt) */
@media (max-width: 768px) {
    .container {
        width: 90%;
        padding: 20px;
        margin: 30px auto;
    }
    h2 {
        font-size: 24px;
    }
    label,
    input,
    textarea,
    button {
        font-size: 15px;
    }
    button {
        padding: 10px;
    }
}
```

```

/* Mobil nézet (480px alatt) */
@media (max-width: 480px) {
  h2 {
    font-size: 20px;
  }
  label,
  input,
  textarea,
  button {
    font-size: 14px;
  }
  button {
    padding: 10px;
  }
}
</style>
</head>

<body>
  <!-- Fejléc és navigáció -->
  <header>
    <a href="index.html" class="logo">
      Időpontfoglaló <span id="felhasznalo-nev"></span>
    </a>
    <nav>
      <a href="szolgaltatok.html">Szolgáltatók</a>
      <a href="idopontok.html">Időpontok</a>
      <a href="ertekelesek.html">Értékelések</a>
      <a href="bejelentkezes.html" id="nav-login"><b>Bejelentkezés</b></a>
      <a href="fiokom.html" id="nav-account" class="hidden">Fiókom</a>
      <a href="#" id="nav-logout" class="hidden" onclick="kijelentkezes()">
        <span style="color:red">Kijelentkezés</span>
      </a>
    </nav>
  </header>

  <!-- Oldalcím -->
  <h2>Foglalás módosítása</h2>

  <!-- Űrlap konténer -->
  <div class="container">
    <form id="modositas-form">
      <!-- Állapot mező -->
      <label for="allapot">Állapot:</label>
      <input type="text" id="allapot" name="allapot" required />

      <!-- Megjegyzés mező -->
      <label for="megjegyzes">Megjegyzés:</label>

```

```

        <textarea id="megjegyzes" name="megjegyzes" rows="4"></textarea>

        <!-- Mentés gomb -->
        <button type="submit">Mentés</button>
    </form>
</div>

<script>
    // Felhasználó ellenőrzése
    const user = JSON.parse(sessionStorage.getItem('felhasznalo'));
    if (!user || user.szerepkor !== 'felhasznalo') {
        alert("Csak bejelentkezett felhasználó módosíthat foglalást.");
        window.location.href = "bejelentkezes.html";
    }
    // Felhasználónév beállítása
    document.getElementById('felhasznalo-nev').textContent = ` - ${user.nev}`;
    // Navigáció állapotok beállítása
    document.getElementById('nav-login').classList.add('hidden');
    document.getElementById('nav-account').classList.remove('hidden');
    document.getElementById('nav-logout').classList.remove('hidden');

    // URL paraméterből ID kinyerése
    const params = new URLSearchParams(window.location.search);
    const id = params.get('id');
    if (!id) {
        alert("Hiányzó foglalás ID.");
        window.location.href = "fiokom.html";
    }

    // Foglalás adatainak betöltése
    fetch(`Foglalasaim_listazasa.php?felhasznalo_id=${user.id}`)
        .then(res => res.json())
        .then(json => {
            if (json.error) throw new Error(json.error);
            const fog = json.data.find(f => f.id == id);
            if (!fog) {
                alert("Nincs ilyen foglalás.");
                window.location.href = "fiokom.html";
            }
            document.getElementById('allapot').value = fog.allapot;
            document.getElementById('megjegyzes').value = fog.megjegyzes || '';
        })
        .catch(err => {
            console.error(err);
            alert("Hiba a foglalás betöltésekor: " + err.message);
            window.location.href = "fiokom.html";
        });

```

```
// Form beküldése PUT módszer szimulációval
document.getElementById('modositas-form').addEventListener('submit', e => {
  e.preventDefault();
  const payload = {
    _method: 'PUT',
    allapot: document.getElementById('allapot').value,
    megjegyzes: document.getElementById('megjegyzes').value
  };
  // Továbbítás POST-tal
  fetch(`Foglalasok_modositasa.php?id=${id}`, {
    method: 'POST',
    headers: { 'Content-Type': 'application/json' },
    body: JSON.stringify(payload)
  })
  .then(res => res.json())
  .then(res => {
    if (res.status) {
      alert("Foglalás sikeresen módosítva.");
      window.location.href = "fiokom.html";
    } else {
      alert("Hiba: " + (res.error || 'Ismeretlen hiba'));
    }
  })
  .catch(err => {
    console.error(err);
    alert("Hiba a módosítás során.");
  });
});

// Kijelentkezés függvény
function kijelentkezés() {
  sessionStorage.removeItem('felhasznalo');
  window.location.href = "index.html";
}
</script>
</body>

</html>
```

Foglalások törlése

```
<!DOCTYPE html>
<html lang="hu">
<head>
  <!-- Dokumentum típusa és karakterkészlet beállítása -->
  <meta charset="UTF-8"/>
  <!-- Reszponzív nézet beállítása -->
  <meta name="viewport" content="width=device-width, initial-scale=1.0"/>
```

```
<!-- Navigációs menü stíluslap betöltése -->
<link rel="stylesheet" href="menu.css">
<!-- Oldal címe -->
<title>Foglalás törlése</title>
<style>
  /* Alapoldal stílusok */
  body {
    font-family: Arial, sans-serif;
    background-color: #f9f9f9;
    margin: 0; padding: 0;
  }
  /* Űrlap tároló középre igazítása */
  .container {
    max-width: 500px;
    margin: 60px auto;
    background: white;
    padding: 30px;
    border-radius: 8px;
    box-shadow: 0 2px 8px rgba(0,0,0,0.1);
    text-align: center;
  }
  /* Címsor stílusa */
  h2 {
    text-align: center;
    color: #dc3545;
    margin-bottom: 20px;
  }
  /* Információs szöveg */
  p {
    margin-bottom: 30px;
    font-size: 16px;
  }
  /* Gombok alapstílusai */
  button {
    padding: 10px 20px;
    font-size: 16px;
    border: none;
    border-radius: 5px;
    cursor: pointer;
  }
  /* Törlés gomb */
  .delete-btn {
    background-color: blue;
    color: white;
    margin-right: 10px;
  }
  .delete-btn:hover {
    background-color: blue;
```

```
}
/* Mégse gomb */
.cancel-btn {
  background-color: #6c757d;
  color: white;
}
.cancel-btn:hover {
  background-color: #5a6268;
}
/* Mobil nézet (768px alatt) */
@media (max-width: 768px) {
  .container {
    width: 90%;
    padding: 20px;
    margin: 40px auto;
  }
  h2 { font-size: 24px; }
  p { font-size: 15px; }
  button {
    width: 100%;
    margin: 10px 0;
    font-size: 15px;
  }
}
/* Mobil nézet (480px alatt) */
@media (max-width: 480px) {
  h2 { font-size: 20px; }
  p { font-size: 14px; }
  button {
    font-size: 14px;
    padding: 10px;
  }
}
</style>
</head>
<body>
  <!-- Fejléc és navigáció -->
  <header>
    <a href="index.html" class="logo">
      Időpontfoglaló <span id="felhasznalo-nev"></span>
    </a>
  <nav>
    <a href="szolgaltatok.html">Szolgáltatók</a>
    <a href="idopontok.html">Időpontok</a>
    <a href="ertekelesek.html">Értékelések</a>
    <a href="fiokom.html" id="nav-account">Fiókom</a>
    <a href="#" onclick="kijelentkezés()">
      <span style="color:red">Kijelentkezés</span>
    </a>
  </nav>
</body>
```

```

    </a>
  </nav>
</header>

<!-- Oldalcím -->
<h2>Foglalás törlése</h2>

<!-- Törlési megerősítő konténer -->
<div class="container">
  <!-- Információ az aktuális foglalásról -->
  <p id="info">Betöltés...</p>
  <!-- Véglegesítés gomb -->
  <button class="delete-btn" id="confirm-delete">Törlés véglegesítése</button>
  <!-- Mégse gomb -->
  <button class="cancel-btn" id="cancel">Mégse</button>
</div>

<script>
  // Jogosultság ellenőrzése: csak felhasználó
  const user = JSON.parse(sessionStorage.getItem('felhasznalo'));
  if (!user || user.szerepkor !== 'felhasznalo') {
    alert("Csak bejelentkezett felhasználó törölhet foglalást.");
    location.href = "bejelentkezes.html";
  }
  // Felhasználónév megjelenítése a fejlécben
  document.getElementById('felhasznalo-nev').textContent = ` - ${user.nev}`;

  // URL paraméterként érkező foglalás ID
  const params = new URLSearchParams(window.location.search);
  const id = params.get('id');
  if (!id) {
    alert("Hiányzó foglalás ID.");
    location.href = "fiokom.html";
  }

  // Foglalás adatainak lekérése és info mező frissítése
  fetch(`Foglalasaim_listazasa.php?felhasznalo_id=${user.id}`)
    .then(response => response.json())
    .then(data => {
      const fog = data.data.find(f => f.id == id);
      if (!fog) throw new Error("Érvénytelen foglalás ID.");
      document.getElementById('info').textContent =
        `Biztosan törölni szeretné a következő foglalást? Dátum: ${fog.datum},
        Idő: ${fog.ido}`;
    })
    .catch(error => {
      alert(error.message);
      location.href = "fiokom.html";
    });

```



```
});

// Törlés gomb eseménykezelő
document.getElementById('confirm-delete').addEventListener('click', () => {
    fetch(`Foglalasok_torlese.php?id=${id}`)
        .then(res => res.json())
        .then(result => {
            if (result.status) {
                alert("Foglalás törölve.");
                location.href = "fiokom.html";
            } else {
                alert("Hiba: " + (result.error || "Ismeretlen hiba"));
            }
        })
        .catch(() => {
            alert("Hiba a törlés során.");
        });
});

// Mégse gomb visszairányítás
document.getElementById('cancel').addEventListener('click', () => {
    location.href = "fiokom.html";
});

// Kijelentkezés funkció
function kijelentkezés() {
    sessionStorage.removeItem('felhasznalo');
    location.href = "index.html";
}
</script>
</body>
</html>
```

HTML kimenetek

← → ↺

localhost/idopontfoglalo/Foglasok_letrehozasa.html?id=133

☆

🔖

📄

🌐

⋮

Időpontfoglaló - Orosz Kristóf

SzolgáltatókIdőpontokÉrtékelésekFiómkomKijelentkezés

Foglalás megerősítése

Dátum:

2025-06-19

Idő:

20:14:00

Megjegyzés (opcionális):

P1. Kérem, érkezzen pontosan...

Foglalás megerősítése

← → ↺

localhost/idopontfoglalo/Foglasok_modositasa.html?id=33

☆

🔖

📄

🌐

⋮

Időpontfoglaló - Orosz Kristóf

SzolgáltatókIdőpontokÉrtékelésekFiómkomKijelentkezés

Foglalás módosítása

Állapot:

lefoglalt

Megjegyzés:

Mentés

Szakmai Gyakorlat I. – Frontend – Backend integrációs szakasz

Orosz Kristóf – EYZWG9

