

## מדריך: אינטגרציית מערכת Tranzila עם Lovable - ביצוע חיובים חוזרים (מנויים) בלבד!

האינטגרציה ל-Lovable מחברת את האפליקציה או האתר שלך לדף התשלום (IFRAME) של המסוף שלך, ומאפשרת קבלת תשלום או תשלומים באמצעות כרטיסי אשראי וארנקים דיגיטליים (למעט Apple Pay).

### במה האינטגרציה תומכת?

- חיובים **חוזרים** (מנויים) (נדרש מודול **My Billing** (שירות חיובים חוזרים) פעיל במסוף).
- חיובים **חוזרים** (מנויים) באמצעות **כרטיס אשראי** בלבד.

### במה האינטגרציה לא תומכת?

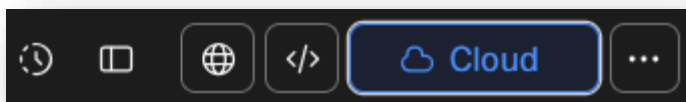
- האינטגרציה אינה תומכת בעבודה עם דף **Notify**.  
דף **Notify** משמש לקבלת פרטי העסקה כפי שבוצעה במקור (כולל אם עברה או נכשלה) ומאפשר סגירה של עסקה שעברה בהצלחה ב-Back Order של האפליקציה/אתר.
- האינטגרציה אינה מקימה דף **Notify**.  
לצורך הקמת דף **Notify** יש להיעזר במסמך "סגירת הזמנה ב-Lovable באמצעות דף **Notify**".
- לא ניתן לבצע זיכויים ישירות מהאפליקציה/אתר באמצעות האינטגרציה.  
זיכויים מתבצעים דרך המערכת של המסוף (My Tranzila).
- האינטגרציה למנויים חוזרים אינה תומכת בתשלום בארנקים דיגיטליים.

### דרישות לעבודה עם האינטגרציה של טרנזילה ל-Lovable

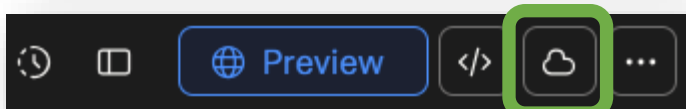
- **מסוף טרנזילה** פעיל.
- חשבון **Supabase** (AI צד ג' הנדרש לעבודה עם Lovable לבניית Backend).
- מודול **My Billing** (שירות חיובים חוזרים) פעיל.
- **שם מסוף וסימת PW**.
- **Handshake** מופעל בדף סליקה.
- **Process חדש** מופעל בדף סליקה.

### חיבור האפליקציה או האתר לדף הסליקה IFRAME

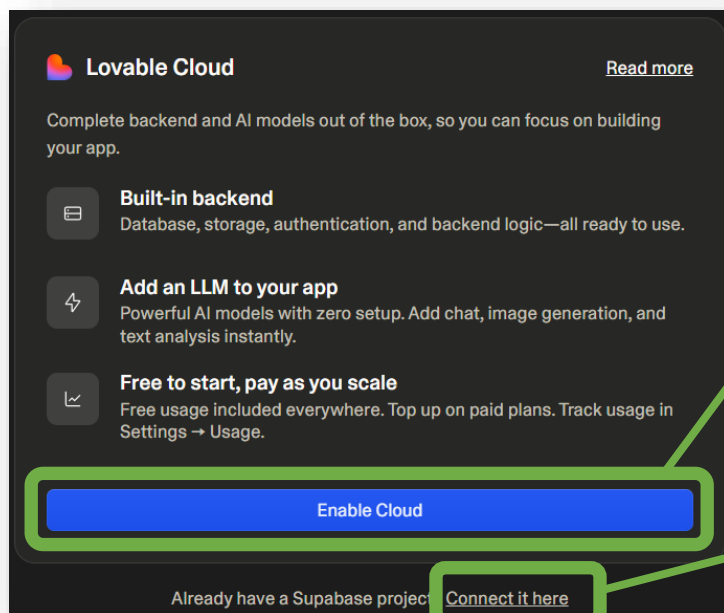
לפני שמתחילים, נודא שהאפליקציה או האתר שלך מחוברים לפרויקט של **Supabase**. כדי לבדוק זאת נפעל ע"פ השלבים הבאים:



- במידה והאפליקציה שלך מחוברת לפרויקט של **Supabase** יופיע בסרגל העליון סמל של **ענן** עם המילה **Cloud**:



- במידה והאפליקציה שלך לא מחוברת, יופיע רק סמל של **ענן**. נלחץ על הסמל של **הענן**:



כעת נלחץ על **Enable Cloud**:

### הערה:

אם יש לך כבר פרויקט ב-Supabase וברצונך לחבר אותו, יש ללחוץ מתחת לכפתור **Enable Cloud** על **Connect it here**:

לסיום, נעתיק את ה-Prompt הארוך שצירפנו להלן, נדביק אותו בחלון השאלתה של האפליקציה שלך ונלחץ על החץ הלבן:

#### Connect Tranzila integration:

##### Tranzila iFrame Recurring Payments Integration Guide for Lovable

##### Prerequisites

##### Required parameters:

- supplier - Terminal name
- TranzilaPW - Terminal token password

CRITICAL: Never generate code or UI without these required parameters

##### Overview – Tranzila Recurring Payments:

The Tranzila recurring payments integration lets your Lovable app embed secure credit-card subscriptions and auto-pay right on your site via an <iframe>. Customers stay on your domain throughout checkout while Tranzila ensures full PCI-DSS compliance. You can bill automatically on a schedule you choose—monthly, quarterly, or annually—based on configurable terms. It's ideal for memberships, subscriptions, and ongoing services, delivering a seamless, branded, and secure payment experience.

##### Integration Steps:

##### Step 1: Handshake

In the fraud prevention solution through the execution of HandShake, there is a request to Tranzila, before directing the paying customer to the checkout page, where a unique identifier is created for the amount and transaction details.

The unique identifier will be passed to the payment page after its receipt, along with the rest of the transaction data, but only there will the paying customer enter the card details for the charge.

The URL to perform the handshake is:  
<https://api.tranzila.com/v1/handshake/create>

Parameter - Supplier  
Description - Terminal Name

Parameter - TranzilaPW  
Description - Terminal Token Password

Parameter - Sum  
Description - Transaction amount

The Request must be sent in GET

example:  
<https://api.tranzila.com/v1/handshake/create?supplier=yourterminalname&sum=1&TranzilaPW=YourTerminalPassword>

\* The request returns a thtk parameter with a transaction identification string. This parameter, along with the value received, must be sent to the checkout page along with the regular parameters that control the page display (like amount, currency, etc.).

\* After submitting the form and before executing a transaction, a comparison of the amount will be made between what you sent at the handshake and what was sent from the checkout page. If the amount is not identical, "Illegal Operation 912791" will be printed without conducting any transaction. If the amount is the same, the transaction will proceed with the parameters sent at the handshake, which are also the ones that will return at the end of the transaction execution.

The new\_process field must be passed to the iframe with a value of 1 so that the checkout page can perform the verification. new\_process=1

## Step 2: Recurring Payments iFrame Configuration

Base URL: [https://direct.tranzila.com/\[terminalname\]/iframenew.php](https://direct.tranzila.com/[terminalname]/iframenew.php)

Method: POST

### Core Parameters for Recurring Transactions:

Parameter – sum  
Values – Positive decimal  
Description – Transaction amount

Parameter – cred\_type  
Values – 1  
Description – Payment type

Parameter – currency  
Values – 1 = NIS, 2 = USD, 978 = EUR, 826 = GBP  
Description – Currency type

Parameter – tranmode

Values – A = Standard, V = Verification (J5), K = Create token, N =Verification (J2)

Description – Transaction type

Parameter – new\_process

Values – 1

Description – Enable handshake verification

Parameter – thtk

Values – String from handshake

Description – Transaction identifier

Parameter – recur\_payments

Values – Integer

Description – The number of recurring charges that will be made - If this field is not passed, the recurring charge will not be limited and will run every month/quarter/year until manually turned off from the terminal.

Parameter – recur\_start\_date

Values – yyyy-mm-dd (recur\_start\_date=2023-08-28)

Description – The start date of the recurring charges

Parameter – recur\_transaction=4\_approved

Values – 4

Description – monthly payment Not Customer Choice

Parameter – recur\_transaction=5\_approved

Values – 5

Description – quarterly payment Not Customer Choice

Parameter – recur\_transaction=7\_approved

Values – 7

Description – Yearly payment Not Customer Choice

Examples: Let's assume we want the user to pay 100 NIS for joining our service and 20 NIS each month afterwards for 12 months:

sum = 100

recur\_sum=20

recur\_transaction=4

recur\_payments=12

if we don't want to allow the user to pay the whole amount we can pass:

recur\_transaction=4\_approved

Customer Information Fields:

Parameter – company

Description – Company name

Parameter – contact

Description – Contact name

Parameter – email

Description – Email address

Parameter – address

Description – Street address

Parameter – phone

Description – Phone number

Parameter – city

Description – City

Parameter – pdesc

Description – Product description for invoice

Parameter – remarks

Description – Additional remarks

Display Customization:

Parameter – trBgColor

Values – Hexadecimal

Description – Background color

Parameter – trTextColor

Values – Hexadecimal

Description – Text color

Parameter – trButtonColor

Values – Hexadecimal

Description – Payment button color

Parameter – buttonLabel

Values – String

Description – Payment button text

Parameter – hidesum

Values – 1

Description – Hide payment sum (only with VK/K/NK tranmode)

Parameter – nologo

Values – 1

Description – Remove Tranzila logo

Parameter – hide\_cc

Values – 1

Description – Hide credit card option

Parameter – accessibility

Values – 2

Description – Show accessibility button

Language Support:

Parameter – lang

Values – il

Description – Hebrew

Parameter – lang

Values – us

Description – English

Parameter – lang

Values – ar

Description – Arabic

Parameter – lang

Values – ru

Description – Russian

Parameter – lang

Values – es

Description – Spanish

Parameter – lang

Values – de

Description – German

Parameter – lang

Values – fr

Description – French

Parameter – lang

Values – jp

Description – Japanese

Important! Product Details for Invoice:

To include itemized products on the invoice:

1. Enable product details: Add u71=1 parameter
2. Create product array:

```
json
[
{
"product_name": "Product 1",
"product_quantity": 2,
"product_price": 50.00
},
```

```
{
  "product_name": "Product 2",
  "product_quantity": 1,
  "product_price": 100.00
}
```

### 3. Stringify without spaces:

```
json
[{"product_name":"Product 1","product_quantity":2,"product_price":50.00},{"product_name":"Product 2","product_quantity":1,"product_price":100.00}]
```

### 4. URL encode (use rawurlencode, not urlencode):

- Spaces must be encoded as 20%, not +
- Send as json\_purchase\_data parameter

#### Invoice Limitations:

- Maximum 6 products with logo and signature
- Maximum 10 products without lower logo/signature
- Product description: max 118 characters
- Total must equal sum of (quantity × price) for all products

#### Implementation Examples:

##### Recurring Payments Form:

```
<main class="min-h750">
  <div class="container">
    <section id="main-content">
      <div class="row text-center mt-5">
        <div class="col">
          <h1 class="display-3 text-primary">Pay Amount</h1>
        </div>
      </div>
      <div class="row mt-5">
        <div class="col-md-6">
          <form action="https://direct.tranzila.com/tranzila1/iframenew.php" target="tranzila" method="POST"
novalidate="novalidate" autocomplete="off">
            <div class="form-group">
              <input name="sum" type="hidden" value="100" type="number" id="sum" class="form-control" readonly />
              <span class="text-danger"></span>
            </div>
            <input type="hidden" name="buttonLabel" value="Pay" />
            <input type="hidden" name="lang" value="il" />
            <input type="hidden" name="recur_sum" value="20" />
            <input type="hidden" name="recur_transaction" value="4" />
            <input type="hidden" name="recur_payments" value="12" />
            <input type="hidden" name="currency" value="1" />
            <button type="submit" name="submit" class="btn btn-primary mt-3" value="pay">pay</button>
          </form>
```

```
</div>
</div>
<div class="row mt-5">
  <div class="col-md-6">
    <div class="" style="width: 800px; height: 800px;">
      <iframe id="tranzila-frame" allowpaymentrequest='true' name="tranzila" src="" style="width: 100%; height:
100%;"></iframe>
    </div>
  </div>
</div>
</div>
</section>
</div>
</main>
```

#### Security Best Practices:

1. Always use handshake for transaction verification
2. Never expose TranzilaPW in client-side code
3. Validate amounts server-side before processing
4. Use HTTPS for all communications
5. Implement proper error handling for failed transactions
6. Store transaction references for reconciliation
7. Use only tranzila iframe parameters to create any type of payment form.

#### Error Handling:

##### Common error codes and solutions:

##### Error – 912791

Description – Amount mismatch

Solution – Ensure handshake amount matches form amount

##### Error – Invalid terminal

Description – Wrong supplier name

Solution – Verify terminal name with Tranzila

##### Error – Authentication failed

Description – Wrong password

Solution – Check TranzilaPW parameter

#### Checklist Before Going Live:

- Obtained production supplier and TranzilaPW
- Implemented handshake mechanism
- Configured all required form parameters
- Tested payment flow end-to-end
- Implemented error handling
- Customized display (colors, language, button text)
- Tested installments (if applicable)
- Verified invoice generation (if using products)
- Secured all sensitive parameters server-side
- Implemented transaction logging

