

Deposit Fixes Summary

Проблема 1: Admin Finance Wallet - TypeError ☒ FIXED

Помилка

```
TypeError: l.map is not a function
    at getValue (wallet-37e972aa295db8d4.js:1:1570)
```

Причина

В колонці "Custom Addresses" код намагався викликати `.map()` на non-array значенні. Поле `customAddressWalletsPairFields` могло бути JSON string замість array.

Файл

`src/pages/admin/finance/wallet/index.tsx`

Вирішення

Додано JSON parsing та array validation в обох колонках:

1. Колонка "Has Custom Address" (lines 53-72)

```
getValue: (item) => {
  let addresses = item.user?.customAddressWalletsPairFields;

  // Parse JSON string if needed
  if (typeof addresses === 'string') {
    try {
      addresses = JSON.parse(addresses);
    } catch (e) {
      addresses = [];
    }
  }

  return Array.isArray(addresses) && addresses.length > 0;
}
```

2. Колонка "Custom Addresses" (lines 74-99)

```
getValue: (item) => {
  let addresses = item.user?.customAddressWalletsPairFields;
```

```
// Parse JSON string if needed
if (typeof addresses === 'string') {
  try {
    addresses = JSON.parse(addresses);
  } catch (e) {
    console.error('Failed to parse customAddressWalletsPairFields:', e);
    addresses = [];
  }
}

// Validate array
if (!Array.isArray(addresses) || addresses.length === 0) {
  return "-";
}

return addresses
  .map((a: { currency: string; network: string; address: string }) =>
    `${a.currency}/${a.network}: ${a.address}`
  )
  .join(", ");
}
```

Проблема 2: User застряє на "Processing payment..." ☒ FIXED

Ситуація

1. Admin approve deposit в `/admin/finance/transaction` ☒
2. Transaction status змінився на COMPLETED ☒
3. User balance має оновитись ☒
4. Але user page показує "Processing payment..." і не переходить далі ☒

Причина

WebSocket підписка працювала ТІЛЬКИ коли user manually надсилав transaction hash (`transactionSent === true`). Коли admin approve deposit напряму, user не отримував WebSocket updates.

Файл

`src/pages/user/wallet/deposit/index.tsx`

Вирішення

1. Розширено WebSocket підписку (lines 120-170)

Було:

```

useEffect(() => {
  if (wsManager && step === 4 && selectedWalletType.value === "SPOT" &&
transactionSent) {
    // Тільки якщо user надіслав transaction
    wsManager.send({
      action: "SUBSCRIBE",
      payload: { trx: transactionHash }
    });
  }
}, [wsManager, step, transactionSent, transactionHash]);

```

Стало:

```

useEffect(() => {
  if (wsManager && step === 4 && selectedWalletType.value === "SPOT") {
    if (transactionSent && transactionHash) {
      // User надіслав transaction - підписуємось на trx
      console.log('[Deposit] Subscribing to transaction verification', {
transactionHash });
      wsManager.send({
        action: "SUBSCRIBE",
        payload: { trx: transactionHash }
      });
    } else {
      // Чекаємо admin approval - підписуємось на userId + currency + network
      console.log('[Deposit] Subscribing to pending deposits (waiting for admin
approval)', {
        userId: profile?.id,
        currency: selectedCurrency,
        network: selectedDepositMethod
      });
      wsManager.send({
        action: "SUBSCRIBE",
        payload: {
          userId: profile?.id,
          currency: selectedCurrency,
          network: selectedDepositMethod,
        },
      });
    }
  }

  return () => {
    // Cleanup - unsubscribe відповідно до типу підписки
    if (transactionHash) {
      wsManager.send({
        action: "UNSUBSCRIBE",
        payload: { trx: transactionHash },
      });
    } else {
      wsManager.send({

```

```

        action: "UNSUBSCRIBE",
        payload: {
            userId: profile?.id,
            currency: selectedCurrency,
            network: selectedDepositMethod,
        },
    });
}
};
}
}, [wsManager, step, transactionSent, transactionHash, profile?.id,
selectedCurrency, selectedDepositMethod]);

```

2. Додано детальний logging в WebSocket handler (lines 89-129)

```

manager.on("message", (message: any) => {
    console.log('[Deposit] WebSocket message received:', message);

    if (!(message && message.data) || message.stream !== "verification") {
        console.log('[Deposit] Ignoring message - wrong format or stream:', {
            hasData: !(message && message.data),
            stream: message?.stream
        });
        return;
    }

    console.log('[Deposit] Processing verification message:', {
        status: message.data.status,
        message: message.data.message
    });

    switch (message.data.status) {
        case 200:
        case 201:
            console.log('[Deposit] ☒ Deposit verified successfully!');
            toast.success(message.data.message);
            setDeposit(message.data);
            setLoading(false);
            setStep(5);
            break;
        case 400:
        case 401:
        case 403:
        case 404:
        case 500:
            console.log('[Deposit] ☐ Deposit verification failed:',
message.data.status);
            setLoading(false);
            toast.error(message.data.message);
            break;
    }
}

```

```

    default:
      console.log('[Deposit] Unknown status code:', message.data.status);
      break;
    }
  });
});

```

3. Додано polling fallback mechanism (lines 171-240)

Якщо WebSocket не працює, система автоматично перевіряє статус кожні 10 секунд:

```

useEffect(() => {
  if (step === 4 && loading && selectedWalletType.value === "SPOT" &&
  depositAddress?.address) {
    console.log('[Deposit] Starting polling fallback for deposit status');

    const checkDepositStatus = async () => {
      try {
        console.log('[Deposit] Polling: Checking deposit status...');
        const { data, error } = await $fetch({
          url: "/api/finance/deposit/status",
          method: "GET",
          silent: true,
          params: {
            userId: profile?.id,
            currency: selectedCurrency,
            address: depositAddress.address,
          },
        });

        if (!error && data) {
          console.log('[Deposit] Polling: Status response:', data);

          if (data.status === 'COMPLETED') {
            console.log('[Deposit] ✅ Polling detected completed deposit!');
            toast.success('Deposit confirmed successfully');
            setDeposit(data);
            setLoading(false);
            setStep(5);
          } else {
            console.log('[Deposit] Polling: Deposit still pending');
          }
        }
      } catch (err) {
        console.error('[Deposit] Polling error:', err);
      }
    };

    // Check immediately
    checkDepositStatus();
  }
});

```

```
// Then check every 10 seconds
const interval = setInterval(checkDepositStatus, 10000);

return () => {
  console.log('[Deposit] Stopping polling fallback');
  clearInterval(interval);
};
}
}, [step, loading, selectedWalletType.value, depositAddress?.address,
profile?.id, selectedCurrency]);
```

4. Додано \$fetch import (line 20)

```
import $fetch from "@/utils/api";
```

Тестування

Fix 1: Admin Wallet Page

1. Відкрити `/admin/finance/wallet`
2. Перевірити що сторінка відкривається без помилок
3. Перевірити що колонка "Custom Addresses" відображається правильно
4. Перевірити що колонка "Has Custom Address" працює коректно

Fix 2: User Deposit Flow

Сценарій A: User надсилає transaction hash

1. User відкриває `/user/wallet/deposit`
2. Вибирає SPOT wallet → Currency → Network
3. Копіює deposit address
4. Надсилає crypto
5. Вводить transaction hash
6. Натискає "Deposit"
7. ☒ WebSocket підписується на `trx: transactionHash`
8. Backend verify → WebSocket notification → Step 5

Сценарій B: Admin approves deposit

1. User відкриває `/user/wallet/deposit`
2. Вибирає SPOT wallet → Currency → Network
3. Бачить deposit address (step 4, loading = true)
4. Admin відкриває `/admin/finance/transaction`
5. Admin approve deposit
6. ☒ WebSocket підписується на `userId + currency + network`

7. Backend approve → WebSocket notification → Step 5
8. ☒ Fallback polling перевіряє кожні 10 сек

Сценарій C: WebSocket не працює

1. Те саме що Сценарій B
2. Але WebSocket connection failed
3. ☒ Polling fallback автоматично перевіряє статус кожні 10 секунд
4. Коли deposit COMPLETED → автоматично переходить на Step 5

Debug Console Logs

WebSocket Setup

```
[Deposit] Setting up WebSocket for SPOT deposit monitoring {  
  currency: "BTC",  
  network: "BTC",  
  address: "bc1q..."  
}
```

WebSocket Subscription (3 transaction hash)

```
[Deposit] Subscribing to transaction verification {  
  transactionHash: "0x123..."  
}
```

WebSocket Subscription (без transaction hash - admin approval)

```
[Deposit] Subscribing to pending deposits (waiting for admin approval) {  
  userId: 123,  
  currency: "BTC",  
  network: "BTC"  
}
```

WebSocket Message Received

```
[Deposit] WebSocket message received: {  
  stream: "verification",  
  data: {  
    status: 200,  
    message: "Deposit confirmed"  
  }  
}
```

```
[Deposit] Processing verification message: {
  status: 200,
  message: "Deposit confirmed"
}
[Deposit] ☒ Deposit verified successfully!
```

Polling Fallback

```
[Deposit] Starting polling fallback for deposit status
[Deposit] Polling: Checking deposit status...
[Deposit] Polling: Status response: { status: "PENDING" }
[Deposit] Polling: Deposit still pending
... (10 seconds later) ...
[Deposit] Polling: Checking deposit status...
[Deposit] Polling: Status response: { status: "COMPLETED" }
[Deposit] ☒ Polling detected completed deposit!
[Deposit] Stopping polling fallback
```

Backend Requirements

Для повної роботи цих fixes, backend має підтримувати:

1. WebSocket Endpoint: `/api/finance/deposit/spot`

- Має приймати SUBSCRIBE з `{ trx: string } АБО { userId, currency, network }`
- Має відправляти messages з `{ stream: "verification", data: { status: 200|400|..., message: string } }`

2. REST API Endpoint: `/api/finance/deposit/status` (для polling)

- Method: GET
- Params: `userId, currency, address`
- Response: `{ status: "PENDING" | "COMPLETED" | "FAILED" }`

Summary

☒ **Fix 1:** Admin wallet page тепер safely handles JSON string і array для custom addresses

☒ **Fix 2:** User deposit page тепер отримує updates через:

- WebSocket (з або без transaction hash)
- Polling fallback (кожні 10 сек)
- Детальний logging для debug

Total Files Changed: 2

- `src/pages/admin/finance/wallet/index.tsx`

- `src/pages/user/wallet/deposit/index.tsx`