

Audit des bugs d'affichage de devises - Call a Star

Date de l'audit : 26 décembre 2025

Fichiers identifiés avec bugs de devises

1. app/dashboard/creator/payouts/page.tsx ✖

- Ligne 453: `{settings.minimum.toFixed(2)} €` - Symbole € codé en dur
- Ligne 537: `{{(item.amount / 100).toFixed(2)} €` - Symbole € codé en dur
- Devises dynamiques partiellement implémentées (EUR vs stripeCurrency)
- **Bug critique:** Mélange de EUR (base) et stripeCurrency

2. app/dashboard/creator/payouts/settings/page.tsx ✖

- Ligne 78: `'Le montant minimum doit être au moins 10 €'` - Montant en dur avec €
- Ligne 83: `'Le montant minimum ne peut pas dépasser 1 000 000 €'` - Montant en dur avec €
- Ligne 197: `Montant minimum (€)` - Label avec € codé en dur
- Ligne 209: `(entre 10 € et 1 000 000 €)` - Montants en dur avec €

3. app/dashboard/creator/payouts/request/page.tsx ✖

- Ligne 114: `{{settings?.minimum || 10}} €` - Symbole € codé en dur
- Ligne 238: `({settings?.minimum || 10} €)` - Symbole € codé en dur
- Ligne 260: `Montant (€)` - Label avec € codé en dur
- Ligne 272:
 - Minimum: `{settings?.minimum || 10} €` • Maximum: `{balance?.available.toFixed(2) || 0} €` - Symboles € codés en dur

4. app/dashboard/admin/page.tsx ✖

- Ligne 252: `{totalRevenue.toFixed(2)} €` - Symbole € codé en dur
- Ligne 264: `{totalCommissions.toFixed(2)} €` - Symbole € codé en dur
- Ligne 378: `name="Revenus (€)"` - Label avec € codé en dur
- Ligne 488: `Montant (€)` - Label avec € codé en dur
- Ligne 536: `{Number(p.amount).toFixed(2)} €` - Symbole € codé en dur

5. app/dashboard/user/page.tsx ✖

- Ligne 342: `{Number(request.proposedPrice).toFixed(2)} €` - Symbole € codé en dur

6. app/api/payments/webhook/route.ts ✖

- Ligne 178: `{{(payout.amount / 100).toFixed(2)} €` - Symbole € codé en dur
- Ligne 204: `{{(payout.amount / 100).toFixed(2)} €` - Symbole € codé en dur (email HTML)
- Ligne 271: `{{(payout.amount / 100).toFixed(2)} €` - Symbole € codé en dur
- Ligne 296: `{{(payout.amount / 100).toFixed(2)} €` - Symbole € codé en dur (email HTML)

7. app/dashboard/creator/page.tsx ⚠

- Ligne 43: `const [creatorCurrency, setCreatorCurrency] = useState<string>('EUR');` - Défaut EUR codé en dur

8. app/dashboard/admin/settings/page.tsx ⚠

- Ligne 44: `currency: 'EUR'` - Défaut EUR codé en dur
- Ligne 49: `const currencyOptions = ['EUR', 'USD', 'GBP', 'CHF'];` - Liste statique acceptable

Priorités de correction

Priorité 1 - CRITIQUE (Affichage utilisateur final)

1. **app/dashboard/user/page.tsx** - L'utilisateur doit voir la devise exacte avec laquelle il sera débité
2. **app/dashboard/creator/payouts/settings/page.tsx** - Montant minimum de retrait
3. **app/dashboard/creator/payouts/request/page.tsx** - Formulaire de demande de retrait

Priorité 2 - HAUTE (Interface créateur)

1. **app/dashboard/creator/payouts/page.tsx** - Tableau de bord des paiements
2. **app/dashboard/creator/page.tsx** - Page principale créateur

Priorité 3 - MOYENNE (Backend & Admin)

1. **app/api/payments/webhook/route.ts** - Notifications email
2. **app/dashboard/admin/page.tsx** - Dashboard admin

Solution proposée

Créer un composant CurrencyDisplay





```
// components/ui/currency-display.tsx
export function CurrencyDisplay({ amount, currency }: { amount: number; currency: string }) {
  return <span>{amount.toFixed(2)} {currency}</span>
}
```

Créer une fonction getCurrencySymbol

```
// lib/currency.ts
export function getCurrencySymbol(currency: string): string {
  const symbols: Record<string, string> = {
    EUR: '€',
    USD: '$',
    GBP: '£',
    CHF: 'CHF'
  }
  return symbols[currency] || currency
}
```

Règles strictes à respecter

1. ☒ La devise affichée = la devise du créateur

2.  La devise débitée = la devise affichée
3.  Aucun symbole de devise codé en dur
4.  Toujours utiliser `stripeCurrency` (devise réelle Stripe Connect)
5.  Si estimation affichée, la marquer clairement comme telle